Project Initiation and Maintenance
Integration Specification

InEight®
PLATFORM

INEIGHT

Changelog

This changelog contains only significant or other notable changes to the document revision. Editorial or minor changes that do not affect the context of the document are not included in the changelog.

Rev	Date	Description
1.0	10-JUNE-2019	Initial Release
		Removed ProjectCodes.
		Added details for ProjectEquipments_Get and ProjectEmployees_Get Updated multiple field descriptions in ProjectEmployee for clarity.
		Updated descriptions for start and end dates on ProjectEquipments_Get
2.0	21-OCT-2019	Published update
		Added new fields for Project Equipment (import and get). Updated start and end date fields for Project Equipment (Get) to reflect their true names.
		Updated the Project import integration to include the addition of optional contract related information.
		Added note about deprecation of the Project Codes.
		Added note about equipment and project currencies, and updated field description for EquipmentNumber indicate need for SSID instead of DisplayId to ProjectEquipment.
		Removed Project Equipment (Get) from the document, refer to APIM for this information.
3.0	14-APR-2020	Published revision.
4.0	24-AUG-2020	Updated UoMName description in Project Employees (Post). Field accepts either Name or SourceSystemId of a valid Unit of Measure record. Added ProjectAttributes array and associated fields in Projects. Added section Project Attributes. Published revision (Release 20.7).
5.0	09-OCT-2020	Added BillingClass field in Project Employees (Post), Project Employees (Get), Project Equipment Import (Post), and Project Equipments (Get). Added Project Equipments (Get), which was removed in a previous revision. Deprecated ActivityType field in Project Equipments Import, a legacy field that was not used. Added information about a deleting master data Employee record in Project Employee (Post) verification, added IMPORTANT. Added information about a deleting master data Equipment record in Project Equipment (Post) verification, added IMPORTANT. Published revision (Release 20.9).
6.0	02-MAR-2021	Added integrations Project Settings (Post) and Project Settings (Get). Published revision (Release 21.1).
7.0	30-APR-2021	Updated AttributeValue field description in Projects. Added entry for Project Attribute in Error Messages. Added integrations for Project Employee Billing Classes (Import and Get) Published revision (Release 21.3).
8.0	09-JUL-2021	Added integrations Component Records (Request) and Workplan Component Records (Request). Published revision (Release 21.5).

9.0	17-SEP-2021	Added Forecast and CertificateOf related date fields to Projects. Published revision (Release 21.7).
10.0	25-OCT-2021	In Projects, added Markets and CompanyLegalName fields, and added NOTE in descriptions for Latitude and Longitude fields. Added GlobalOptions array and associated fields to Project Settings (Post) and Project Settings (Get). Published revision (Release 21.10).
11.0	02-FEB- 2022	Updated description of AttributeValue to include custom lists in Projects. Added fields in Projects for revised forecast/contract dates and duration. Published revision (Release 21.12).
12.0	11-MAY-2022	Added integrations for ProjectVendors_Import and ProjectVendors_Get. Published revision (Release 22.2).
13.0	29-JUL-2022	Added integrations for Project Currency. Added BusinessCase field to Projects. Published revision (Release 22.6).
14.0	04-NOV-2022	Added integrations for Rate Codes, Commodities, and Disciplines. Published revision (Release 22.10).
15.0	23-JAN-2023	Updated to add missing ProjectEmployeeBillingClassSourceSystemId field in Project Employees (Post) and Project Employees (Get). Updated to add missing ProjectEquipmentBillingClassSourceSystemId field in Project Equipments (Post) and Project Equipments (Get). Published revision (Release 22.12).
16.0	02-JUN-2023	Added Phase field to Projects. Added integrations for Project Values. Published revision (Release 23.4).
17.0	05-FEB-2024	Added OwnerLegalName and TaxId fields to Projects. Published revision (Release 23.12).
18.0	10 APR-2024	Removed UnionSourceSystemID field in Project Employees (Post) and Project Employees (Get). Published revision (Release 24.3).
19.0	20-JUN-2024	Updated URL, renamed IsActive to IsAvailable, added ID, and deleted Code in Project Values integrations. (updated) Added additional PhaseName and ProjectStatusName fields to PhaseName array in Projects. Published revision (Release 24.5).
20.0	20-AUG-2024	Updated to remove IsActive field and added IsSystem and IsAvailable fields in Commodities and Disciplines. Published revision (Release 24.7).
21.0	04-DEC-2024	Added integrations for Project Allowances, Project Attendance Types, and Project Employee Premiums. Published revision (Release 24.11).
22.0	07-FEB-2025	Renamed integrations Upsert ProjectEquipmentType to Upsert ProjectEquipmentRateCodes and Upsert ProjectCraft to Upsert ProjectCraftRateCodes. Deleted IsActive field and added IsAvailable field in Rate Codes, Project Craft and, Equipment Rate Codes.
23.0	29-MAY-2025	Updated Error Message sections.

Contents

Overview

Project initiation and maintenance integrations are used throughout the lifecycle of a project to create and maintain data that can be used by multiple products in the InEight cloud platform.

Integrations in this Document

Integration	Description
Projects	This integration allows new projects to be created in the InEight cloud platform. In earlier releases, maintenance of projects for status changes, dates and other project data was done manually through the InEight cloud platform UI. **NOTE:** This method is used as a workaround because the Project Status Name cannot be changed through the APIM from new to any other value.
Project Craft	This integration allows InEight customers to make specific rates by craft available for use in a project.
Project Employees (Post)	This integration allows InEight customers to provide a specific list of employees that will be available for use as a resource on a specific project. This list can be managed via the integration as employees come and go from the project, and details such as estimated personal pay rates change. Every employee in the Project Employees integration must already exist in InEight as part of the master list of all employees for the customer (refer to Employees Integration Specification).
Project Employees (Get)	This integration provides the full list of employees assigned to a specified project, and all details about the assignment.
Project Equipment Types	This integration allows InEight customers to maintain specific rates for equipment types on a specified project.
Project Equipment (Post)	This integration allows InEight customers to provide a list of specific equipment that is usable in a project. Records sent in this integration must match the master equipment records for the customer (see Equipment Integration Specification).
Cost Centers	This integration allows a third-party financial system to maintain a list of Cost Centers in InEight.
Project Codes	This integration has been deprecated due to limitations in its intended use. Customers that want to obtain a full list of project records available in the InEight cloud platform can use Projects_Get instead.
Project Settings (Post)	This integration allows a customer to import data from an external system for global options, fuel types, and custom lists in the InEight cloud platform.
Project Settings (Get)	This integration allows a customer to export data for global options, fuel types, and custom lists from the InEight cloud platform to an external system.
Employee Billing Classes	These integrations allow a customer to import and export information for their master list of project employee billing classes between their internal systems and the InEight cloud platform. The integrations allows customers to create and maintain their master list of project employee billing classes, which are used for various business functions in multiple InEight cloud applications. New employee billing classes and updates to billing classes performed using the import API are automatically published (instead of being staged) and are immediately available for use in InEight cloud applications.

Integration	Description
Equipment Billing Classes	These integrations allow a customer to import and export information for their master list of project equipment billing classes between their internal systems and the InEight cloud platform. The integrations allows customers to create and maintain their master list of project equipment billing classes, which are used for various business functions in multiple InEight cloud applications. New equipment billing classes and updates to billing classes performed using the import API are automatically published (instead of being staged) and are immediately available for use in InEight cloud applications.
Project Currency	These integrations allow customers to import and export project currencies between their internal systems and the InEight cloud platform. The integrations allow customers to create and maintain project currencies, which are used for various business functions in multiple InEight cloud applications.
Rate Codes	This integration allows InEight customers to provide a specific list of rate codes that are available for use as a resource on a specific project. This list can be managed via the integration as rate codes are added and removed from the project, and details such as estimated personal pay rate changes.
Disciplines	These integrations allow customers to import and export their master list of disciplines that can be used for various business functions in multiple InEight cloud applications.
Commodities	These integrations allow customers to import and export their master list of commodities that can be used for various business functions in multiple InEight cloud applications.
Project Values	These integrations allow customers to import and export information regarding their list of project values between their internal systems and the InEight cloud platform.
Project Payroll Indicators	These integrations allow a customer to read and export information regarding their project payroll indicator data for allowances, attendance types, and employee premiums from the InEight cloud platform into their internal systems.

Prerequisites

The following table lists prerequisites to using the integrations in this document that are required due to the presence of supporting data or system configurations. All items in the table represent required fields in the integrations. There are many other fields in the integrations that reference data from different areas of the InEight system but are optional and will not prevent the addition or update of a record. These optional fields are called out in the individual integration field descriptions.

Prerequisite	Description	Required by
OrganizationSourceSystemId	All projects must be assigned to an organization entity in InEight. The OrganizationSourceSystemId is created by the Organizations integration.	Projects
CostCenterType	All cost centers must be assigned to a cost center type. Cost center types are manually added into the InEight system.	Cost Centers

NOTE:	To add a cost center type, from the main menu go to Master data libraries > Cost centers, and then select the **Cost Center Types** tab.

Projects

Direction	To the InEight cloud platform from an external system.
Frequency	Integration occurs each time a project is created in the customers external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – On initial customer implementation, potentially hundreds of records. Entire entity – As projects are added, potentially several per month.
APIM Name	Upsert Project
Application Starting Version	18.2

Direction	From the InEight cloud platform to an external system.
Frequency	Determined by external system.
Trigger Methods	Triggered manually by user or is triggered through automated process.
Average Payload Size	Entire entity – Hundreds or thousands of records. Entity delta – Tens or hundreds of records.
APIM Name	List Projects
Application Starting Version	18.2

NOTE:	The fields, field descriptions, and JSON for the Projects_Get and Project_Import integrations are identical, however, the purpose of these integrations and how they are used with the InEight cloud platform differ. Refer to the InEight APIM for details.

Project Attributes

If a request to add or modify a project via Project_Import includes an element within the ProjectAttributes array, the AttributeValue provided must match a valid attribute where value for the attribute equals ProjectID.

The system then determines if the value provided for the attribute matches an existing valid Project SourceSystemId, DisplayId, or Name (whichever are valid fields in the entity). If a match is found and the ProjectAttribute value does not match the project record being imported, updated, or modified, then the system sets the value for the project characteristic with the value provided in the request.

Fields

Depth	Name	Type	Precision	Parent	Req.
1	ProjectDisplayId[1]	String	200		Yes
1	SourceSystemId	String	200		No
1	SourceSystemName	String	50		No
1	ProjectName	String	300		Yes
1	OrganizationSourceSystemId	String	300		Yes
1	BaseCurrency	String	50		Yes
1	TimeZone	String	100		Yes
1	ProjectStatusName	Sting	50		Yes
1	StartDate[2]	String	25		No
1	EndDate[2]	String	25		No
1	City	String	100		No
1	RegionISOCode	String	10		No
1	CountryISOCode	String	10		No
1	ZipCode	String	10		No
1	Address1	String	250		No
1	Address2	String	250		No
1	Latitude	Number	Double		No
1	Longitude	Number	Double		No
1	Notes	String	No Limit		No
1	IsActive	Boolean	N/A		No
1	OriginalContractAmount	Number	Decimal (16,5)		No
1	ContractNumber	String	50		No
1	DateProjectStarted	String	Date		No
1	ContractDate	String	Date		No
1	OriginalContractCompletionDate	String	Date		No
1	OriginalDuration	Number	Int64		No
1	ForecastStartDate	String	Date		No
1	ForecastCompletionDate	String	Date		No
1	ForecastDuration	String	Date		No

Depth	Name	Type	Precision	Parent	Req.
1	ForecastExtensionsReductions	Number	Int64		No
1	ForecastRevisedDuration	Number	Int64		No
1	ForecastRevisedCompletionDate	String	Date		No
1	ContractExtensionsReductions	Number	Int64		No
1	ContractRevisedDuration	Number	Int64		No
1	ContractRevisedCompletionDate	String	Date		No
1	CertificateOfSubstantialCompletionExpectedDate	String	Date		No
1	CertificateOfSubstantialCompletionAwardedDate	String	Date		No
1	CertificateOfFinalCompletionExpectedDate	String	Date		No
1	CertificateOfFinalCompletionAwardedDate	String	Date		No
1	CompanyLegalName	String	255		No
	OwnerLegalName	String	255		No
	TaxId	String	255		No
1	ProjectAttributes	Array	N/A		No
2	AttributeName	String	50	ProjectAttributes	No
2	AttributeSourceSystemId	String	50	ProjectAttributes	Yes
2	AttributeValue	String	300	ProjectAttributes	Yes
1	ProjectPhases	Array	N/A		No
2	PhaseName	String	50	ProjectPhases	Yes
2	ProjectStatusName	String	50	ProjectPhases	Yes
2	PhaseName	String	50	ProjectPhases	Yes
2	ProjectStatusName	String	50	ProjectPhases	Yes
1	Markets	Array	N/A		No
1	BusinessCase	String	100		No

1 - Natural Key field.

2 - The data format for Date/Time fields is YYYY-MM-DDTHH:MM:SS+hhmm, where hhmm is the time zone offset. If the time is already converted to UTC, then the offset will be +0000

Field Descriptions

Name	Description	Example
ProjectDisplayId	Unique identifier that is publicly recognizable. This field is shown in normal use of InEight applications. The value sent in this field can match SourceSystemId if there is no need to use a public versus private unique identifier. When manually editing master data records through the UI or via import, users must understand that DisplayId must be unique and does not perform the same function as SourceSystemId.	10603
SourceSystemId	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not display in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users. When manually editing master data records through the UI or via import, users must understand that if the entity contains records received from integrations, the SourceSystemId is required and must either match a known external system identifier or the record must never be one that is managed by an external system. Failure to follow this rule could result in scenarios where an external system either fails to update an existing record or attempts to create a duplicate record because a matching value could not be found. **IMPORTANT:** When manually entering data through the UI or using an ERP system, SourceSystemId must be provided and cannot be left blank.	10603
SourceSystemName	Name of the external system that is using the integration. Work with InEight to provision a unique value for this field.	D365
ProjectName	Text field that provides a short descriptive name of the project.	LV I-15 Midtown Overpass
OrganizationSourceSystemId	SourceSystemId of the organization to which the project is assigned. Currently, there no integration available for organizations. Provide a SourceSystemId for all organizations created in the InEight cloud platform either through the UI or provide to InEight for an initial implementation seeding.	Region 15
BaseCurrency	Default currency of the project. The value provided in this field must match a currency code in the InEight cloud platform.	USD
TimeZone	Time zone of the project in UTC format.	UTC-06:00
ProjectStatusName	Status of the project. Acceptable values are New, Active, and Closed.	New
StartDate	Starting date of the project in UTC format.	2020-10-31T00:00:00+0000
EndDate	Expected end date of the project in UTC format.	2017-12-31T00:00:00+0000

Name	Description	Example
City	Name of the city where the project is located.	Las Vegas
RegionISOCode	ISO code (3166-2) for the region where the project is located.	US-NV
CountryISOCode	ISO code (Alpha-2) for the country where the project is located.	US
ZipCode	Zip code or postal code of the project location.	89106
Address1	Primary street address of the project location.	1111 Las Vegas Blvd
Address2	Additional street address information for where the project is located.	Suite 151
Latitude	Geographic Information System (GIS) latitude of the project location. Allowed values are maximum of 90 and minimum of -90. **NOTE:** If a value is not provided, zip code/country, then city/state/country is used to calculate the value.	36.244072
Longitude	GIS longitude of the project location. Allowed values are maximum of 180 and minimum of -180. **NOTE:** If a value is not provided, zip code/country, then city/state/country is used to calculate the value.	-115.347356
Notes	Free-form text field for notes about the project.	See Sample JSON.
IsActive	Sending a value of false in this field causes the record to be soft deleted from the InEight cloud platform. If a value is not provided, the default value true is used.	True
OriginalContractAmount	Dollar value of the original contract.	1234567.89
ContractNumber	Identifier of the primary contract for the project.	
DateProjectStarted	Actual start date of the project.	05-21-2018
ContractDate	Effective date of the primary contract.	10-15-2017
OriginalContractCompletionDate	Date on which the primary contract was completed.	08-22-2017
OriginalDuration	Planned duration of the project in days.	758
ForecastStartDate	Forecasted start date of the project in UTC format.	05-21-2018
ForecastCompletionDate	Forecasted completion date of the project in UTC format.	10-15-2018
ForecastDuration	Forecasted duration of the project in whole number.	24
ForecastExtensionsReductions	Forecasted extension of the project duration in whole number. **NOTE:** Read-only UI field in Project details and cannot be updated via Project_Import.	21
ForecastRevisedDuration	Forecasted revised duration of the project in whole number. **NOTE:** Read-only UI field in Project details and cannot be updated via Project_Import.	77
ForecastRevisedCompletionDate	Forecasted revised completion date of the project in UTC format. **NOTE:** Read-only UI field in Project details and cannot be updated via Project_Import.	09-12-2021

Name	Description	Example
ContractExtensionsReductions	Contract extension of the project duration in whole number. **NOTE:** Read-only UI field in Project details and cannot be updated via Project_Import.	44
ContractRevisedDuration	Contract revised duration of the project in whole number. **NOTE:** Read-only UI field in Project details and cannot be updated via Project_Import.	67
ContractRevisedCompletionDate	Contract revised completion date of the project in UTC format. **NOTE:** Read-only UI field in Project details and cannot be updated via Project_Import.	09-12-2021
CertificateOfSubstantialCompletion ExpectedDate	Expected date of receiving the certificate of substantial completion.	05-21-2018
CertificateOfSubstantialCompletion AwardedDate	Expected date of being awarded the certificate of substantial completion.	10-15-2017
CertificateOfFinalCompletionExpect edDate	Expected date of receiving the certificate of final completion.	08-22-2017
CertificateOfFinalCompletionAward edDate	Expected date of being awarded the certificate of final completion.	08-24-2021
CompanyLegalName	Legal name of the company or contractor.	ABC Contractors Inc.
OwnerLegalName	Name that appears on the formation document of a corporation.	
TaxId	Tax identification number used to identify a business entity.	
ProjectAttributes	Use this array to assign values for known project attributes to the Project record. For each attribute, AttributeName or AttributeSourceSystemId must be provided for InEight to find a matching attribute defined in the system. See Project Attributes for details.	
AttributeName	Provide a valid name of a project attribute defined in the system. If a value is provided for this field, do not include AttributeSourceSystemId.	TestAttribute
AttributeSourceSystemId	Provide a valid SourceSystemId of a project attribute defined in the system. If a value is provided for this field, do not include AttributeName.	TestAttributeSSID

Name	Description	Example
AttributeValue	Use to send multiple organizations/sub organization values for an organization or project. Values are comma-delimited with a maximum of 10 values allowed. Duplicate values and empty spaces are ignored. Provide the value for the attribute specified by either AttributeName or AttributeSourceSystemId. The entry provided in this field must match the expected data type of the project attribute. If the expected attribute value is text, number, or date, then the field should contain a string, number, or date, respectively. If the data type of the attribute is specified as data, then the field value must match a valid option for the project attributes data source. For example, if the data source of the project attribute is specified as Organization, then the value must match a valid OrganizationName or SourceSystemId. AttributeValue can also define custom list type project attributes.	"Test" or 12345 or 06-03-2020 or "OrgSSID" (SSID of an organization) or "Custom List"
ProjectPhases	Use this array to define the project phase. **NOTE:** Both Pre-execution and Execution phases must be included in the array.	
PhaseName	Used to set the phase or the current timeline of the project. Acceptable values are Pre-execution and Execution.	Pre-execution
ProjectStatusName	Status of the project. If phase is Pre-execution, acceptable values are Not Started and New. If phase is Execution, acceptable values are Not Started, New, Active, and Closed. **NOTE:** The status Not Started is not available in the UI but acceptable via API.	Not Started
PhaseName	Used to set the phase or the current timeline of the project. Acceptable values are Pre-execution and Execution.	Execution
ProjectStatusName	Status of the project. If phase is Pre-execution, acceptable values are Not Started and New. If phase is Execution, acceptable values are Not Started, New, Active, and Closed.	Active
Markets	Shows the markets that are applicable to the project.	APAC, EMEA, US
BusinessCase	Optional field assigned to a project to be used by the capital product's workflow when a project is funded.	

Error Messages

The following error messages are generated by the InEight cloud platform and products for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity logic errors are performed internally in the InEight cloud platform and products to look for specific business rule or data integrity issues record-by-record. Failures with entity logic validations only cause the individual record to cease processing and are written to internal logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. Possible [Required Field] options: ProjectDisplayId ProjectName OrganizationSourceSystemId BaseCurrency TimeZone ProjectStatusName	400	The request is invalid. The [Required Field] is required.
API Validation	Provided Data Exceeds String Length. Possible [Field Names]: ProjectDisplayId SourceSystemId SourceSystemName ProjectName OrganizationSourceSystemId BaseCurrency TimeZone ProjectStatusName City ReqionISOCode CountryISOCode ZipCode Address1 Address2	400	The request is invalid. The field [Field Name] must be a string with a maximum length of [Field Length].
API Validation	Latitude value is invalid.	400	The field Latitude must be between -90 and 90.
API Validation	Longitude value is invalid.	400	The field Longitude must be between -180 and 180.
Entity Logic Validation	Payload has more than one record with same SourceSystemId/DisplayId.	200	Duplicate SourceSystemId/DisplayId {Comma separated list} found. Aborting the process.
Entity Logic Validation	Payload has conflict on SourceSystemId or DisplayId with Existing Data	200	Ignoring Project '{ProjectDisplayId}' as it is having duplicate SourceSystemId or DisplayId
Entity Logic Validation	Payload data has Invalid Organization	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid OrganizationId
Entity Logic Validation	Payload data has Invalid CountryISOCode	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid CountryISOCode '{CountryISOCode}'
Entity Logic Validation	Payload data has Invalid RegionISOCode	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid RegionISOCode '{RegionISOCode}'
Entity Logic Validation	Payload data has Invalid Currency	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid CurrencyCode'{CurrencyCode}'
Entity Logic Validation	Payload data has Invalid TimeZone	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid TimeZone '{TimeZoneName}'
Entity Logic Validation	Payload data has Invalid ProjectStatus	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid ProjectStatusName '{ProjectStatusName}'

API/Entity Logic	Condition	Code	Message
Entity Logic Validation	Payload data has ProjectAttributes with blank values.	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid AttributeName/AttributeSourceSystemId
Entity Logic Validation	Payload data has ProjectAttributes with incorrect attribute Name or SourceSystemId.	200	Ignoring Project '{ProjectDisplayId}' because Invalid AttributeName/AttributeSourceSystemId '{ invalidNameOrSSID }'
Entity Logic Validation	Payload data has ProjectAttributes with incorrect format for date type attribute.	200	Ignoring Project '{ProjectDisplayId}' because The AttributeValue associated with the ProjectAttribute '{AttributeName}' is not in a valid date format.
Entity Logic Validation	Payload data has ProjectAttributes with incorrect format for number type attribute.	200	Ignoring Project '{ProjectDisplayId}' because The AttributeValue associated with the ProjectAttribute '{AttributeName}' does not match a valid number format.
Entity Logic Validation	Payload data has ProjectAttributes with value outside of the supported range.	200	Ignoring Project '{ProjectDisplayId}' because The AttributeValue associated with the ProjectAttribute '{AttributeName}' must be between '-99999999999.99999' and '99999999999.99999'.
Entity Logic Validation	Payload data has ProjectAttributes with value larger than supported length of the attribute.	200	Ignoring Project '{ProjectDisplayId}' because The Value associated with the ProjectAttribute '{AttributeName}' exceeds the maximum character limit of 300.
Entity Logic Validation	Payload data has ProjectAttribute with the same ProjectSourceSystemId used.	200	Ignoring Project '{ProjectDisplayId}' because 'The AttributeValue associated with the ProjectAttribute '{AttributeName}' should not have Self Project'
Entity Logic Valication	Payload data has less than or greater than two ProjectPhases specified. It should only be two phases "Pre-execution" and "Execution".	200	Ignoring Project '{ProjectDisplayId}' because ProjectPhases payload is invalid
Entity Logic Valication	Payload data has a new Project that currently doesn't exist in the system, and it's ProjectPhases object doesn't have at-least one phase with "New" status.	200	Ignoring Project '{inputProject.ProjectDisplay}' because we cannot create a project without 'New' ProjectStatusName
Entity Logic Valication	Payload data has invalid Project Phase name.	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid PhaseName '{PhaseName}' in ProjectPhases
Entity Logic Valication	Payload data has invalid Project Status name.	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid ProjectStatusName '{ProjectStatusName}' in ProjectPhases
Entity Logic Valication	Payload data has invalid Project Status set for the Phase.	200	Ignoring Project '{project.ProjectDisplay}' as Pre-Execution phase cannot be set for active or closed ProjectStatus.
Entity Logic Valication	Payload data has duplicate phases in the ProjectPhases object.	200	Ignoring Project '{ProjectDisplayId}' as duplicate PhaseName exists in ProjectPhases object

API/Entity Logic	Condition	Code	Message
Entity Logic Valication	Payload data has duplicate statuses in ProjectPhases object.	200	Ignoring Project '{ProjectDisplayId}' as duplicate ProjectStatusName exists in ProjectPhases object
Entity Logic Valication	Payload data doesn't have ProjectPhase objects with as Not started and another as Not Started	200	Ignoring Project '{project.ProjectDisplay}' as 'ProjectPhase must have one 'Not started' status.'
Entity Logic Valication	Payload data has a project to update Project status from "Execution" to "Pre-execution" that's not allowed.	200	Ignoring Project '{ProjectDisplayId}' for update because 'Project Phase cannot be updated to pre-execution because project is in execution phase'
Entity Logic Valication	Payload data has a project to update ProjectStatusName for a "Pre-execution" project without ProjectPhases populated.	200	Ignoring Project '{ProjectDisplayId}' for update because 'ProjectstatusName cannot be modified for the pre-execution project.'
Entity Logic Valication	Payload data has a project that updates a Execution project to "Pre-execution" without setting "ProjectPhases" object.	200	Ignoring Project '{ProjectDisplayId}' for update because 'ProjectstatusName cannot be Not Started.'
Entity Logic Valication	Payload data has incorrect Market.	200	Ignoring Project '{ProjectDisplayId}' as it is having invalid market '{Market}'
Entity Logic Validation	Payload data has StartDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because StartDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has EndDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because EndDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has StartDate greater than EndDate	200	Ignoring Project '{ProjectDisplayId}' because End date must be greater than the start date
Entity Logic Validatoin	Payload data has OriginalDuration as zero.	200	Ignoring Project '{ProjectDisplayId}' as it is having Original Duration is Zero
Entity Logic Validatoin	Payload data has OriginalDuration as a negative value or larger than 99999.	200	Ignoring Project '{ProjectDisplayId}' because the value for OriginalDuration must be between 1 and 99999.
Entity Logic Validatoin	Payload data has DateProjectStarted as NULL but OriginalDuration or OriginalContractCompletionDate has value.	200	Ignoring Project '{ProjectDisplayId}' as DateProjectStarted is required
Entity Logic Validatoin	Payload data has invalid DateProjectStarted	200	Ignoring Project '{ProjectDisplayId}' because DateProjectStarted should be after 01/01/1900
Entity Logic Validatoin	Payload data has invalid OriginalContractCompletionDate	200	Ignoring Project '{ProjectDisplayId}' because OriginalContractCompletionDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has DateProjectStarted greater than OriginalContractCompletionDate.	200	Ignoring Project '{ProjectDisplayId}' because DateProjectStarted is greater than OriginalContractCompletionDate
Entity Logic Validatoin	Payload data has DateProjectStarted, OriginalContractCompletionDate and OriginalDuration fields populated but the calculation OriginalContractCompletionDate – DateProjectStarted = OriginalDuration is not satisfied.	200	Ignoring Project '{ProjectDisplayId}' because it is having invalid OriginalContractCompletionDate

API/Entity Logic	Condition	Code	Message
Entity Logic Validatoin	OriginalDuration calculated from OriginalContractCompletionDate – DateProjectStarted = OriginalDuration is not valid.	200	Ignoring Project '{ProjectDisplayId}' because the value for OriginalDuration must be between 1 and 99999.
Entity Logic Validatoin	ForecastDuration is 0.	200	Ignoring Project '{ProjectDisplayId}' as it is having Forecast Duration is Zero
Entity Logic Validatoin	ForecastDuration is invalid	200	Ignoring Project '{ProjectDisplayId}' because the value for ForecastDuration must be between 1 and 99999.
Entity Logic Validatoin	Payload data has ForecastStartDate as NULL but ForecastDuration or ForecastCompletionDate has value.	200	Ignoring Project '{ProjectDisplayId}' as ForecastStartDate is required
Entity Logic Validatoin	Payload data has invalid ForecastStartDate	200	Ignoring Project '{ProjectDisplayId}' because ForecastStartDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has invalid ForecastCompletionDate	200	Ignoring Project '{ProjectDisplayId}' because ForecastCompletionDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has ForecastStartDate greater than ForecastCompletionDate	200	Ignoring Project '{ProjectDisplayId}' because ForecastStartDate is greater than ForecastCompletionDate
Entity Logic Validatoin	Payload data has ForecastStartDate, ForecastCompletionDate and ForecastDuration fields populated but the calculation ForecastCompletionDate – ForecastStartDate = ForecastDuration is not satisfied.	200	Ignoring Project '{ProjectDisplayId}' because it is having invalid ForecastCompletionDate
Entity Logic Validatoin	ForecastDuration calculated from ForecastCompletionDate – ForecastStartDate = ForecastDuration is not valid.	200	Ignoring Project '{ProjectDisplayId}' because the value for ForecastDuration must be between 1 and 99999.
Entity Logic Validatoin	Payload data has ContractDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because ContractDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has ForecastRevisedCompletionDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because ForecastRevisedCompletionDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has ContractRevisedCompletionDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because ContractRevisedCompletionDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has CertificateOfSubstantialCompletionExpectedDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because CertificateOfSubstantialCompletionExpectedDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has CertificateOfSubstantialCompletionAwardedDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because CertificateOfSubstantialCompletionAwardedDate should be after 01/01/1900

API/Entity Logic	Condition	Code	Message
Entity Logic Validatoin	Payload data has CertificateOfFinalCompletionExpectedDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because CertificateOfFinalCompletionExpectedDate should be after 01/01/1900
Entity Logic Validatoin	Payload data has CertificateOfFinalCompletionAwardedDate earlier than 01/01/1900	200	Ignoring Project '{ProjectDisplayId}' because CertificateOfFinalCompletionAwardedDate should be after 01/01/1900
Entity Logic Validation	Payload has inactive records	200	Ignoring inactive Project {ProjectDisplayId} for adding
Entity Logic Validation	Payload has updates to a closed project other than changing the status.	200	Ignoring Project '{ProjectDisplayId}' because we cannot update any details for the closed project except 'ProjectStatusName'
Entity Logic Validation	Payload has updates to Project currency when there are resources assigned to the project.	200	Ignoring Project '{ProjectDisplayId}' because Currency cannot be modified once the project has resources assigned or project status is in new/active/closed state.
Entity Logic Validation	Inactivating an existing project	200	Ignoring Project {ProjectDisplayId} as delete is not allowed.
Entity Logic Validation	Project update transaction was not successful in the backend.	200	Ignoring invalid Project '{ ProjectDisplayId}' for update
Entity Logic Validation	Payload data has Invalid Organization	200	Ignoring Project '{ProjectDisplayId}' for update as it is having invalid OrganizationSourceSystemId {OrganizationSourceSystemId}
Entity Logic Validation	Payload data has Invalid ProjectStatus	200	Ignoring Project '{ProjectDisplayId}' for update as it is having invalid ProjectStatusName '{ProjectStatusName}'

Sample JSON

```
[
{
"ProjectDisplayId": "TEST_PROJ_001",
"SourceSystemId": "TEST_PROJ_001",
"SourceSystemName": "TEST_PROJ_001",
"ProjectName": "TEST_PROJ_001",
"OrganizationSourceSystemId": "ChildOrg",
"BaseCurrency": "USD",
"TimeZone": "(UTC+05:30) Chennai, Kolkata, Mumbai, New Delhi",
"ProjectStatusName": "New",
"StartDate": "2019-11-03T00:00:00+00:00",
"EndDate": "2022-12-31T00:00:00+00:00",
"City": 'Omaha',
"RegionISOCode": 'US-NE',
"CountryISOCode": 'US',
"ZipCode": 68136,
"Address1": '2356 View Drive',
"Address2": 'Main St',
"Latitude": 41.2565,
"Longitude": -95.9345,
```

```
"Notes": "This is a new free-form text note you have for a project.",
"IsActive": true,
"OriginalContractAmount": 100,
"ContractNumber": "1237905-12-88"
"DateProjectStarted": "2020-03-01T00:00:00+00:00",
"ContractDate": "2020-03-31T00:00:00+00:00",
"OriginalContractCompletionDate": "",
"OriginalDuration": 120,
"ForecastStartDate": "05-21-2018",
"ForecastCompletionDate": "10-15-2018",
"ForecastDuration": 46,
"ForecastExtensionsReductions": 21,
"ForecastRevisedDuration": 77,
"ForecastRevisedCompletionDate": "09-12-2021",
"ContractExtensionsReductions": 44,
"ContractRevisedDuration": 67,
"ContractRevisedCompletionDate": "09-12-2021",
"CertificateOfSubstantialCompletionExpectedDate": "05-21-2018",
"CertificateOfSubstantialCompletionAwardedDate": "10-15-2017",
"CertificateOfFinalCompletionExpectedDate": "08-22-2017",
"CertificateOfFinalCompletionAwardedDate": "08-24-2021",
"CompanyLegalName": " ABC Contractors Inc.",
"OwnerLegalName ": "",
"TaxId": "",
"ProjectAttributes": [
    {
    "AttributeName": "TEST",
    "AttributeSourceSystemId": "TEST",
    "AttributeValue": "TEST1,TEST2,TEST3, ,TEST4, ,, ,TEST5"
    }
    ]
"ProjectPhases": [
    {
    "PhaseName": "Pre-execution",
    "ProjectStatusName": "New",
    "PhaseName": "Execution",
    "ProjectStatusName": "Not Started",
    }
    ]
}
"Markets": "APAC,EMEA,US",
"BusinessCase": ""
]
```

Verification

Projects are accessible from the main navigation menu. To see all projects, click the **Main menu** icon, and then select **All projects & organizations**.

Users are allowed to use Project_Import and Projects_Get APIs to pass Custom List type project attributes. Users can also assign projects with Custom List type project attribute values.

Project Craft

Within a project, employees can use a different hourly rate than the one assigned to their master data resource record. To override the master data rate using one based on a craft (instead of a manually entered value), craft records must first be assigned to the project using this integration to make them available as an option in the assignment UI.

Direction	To the InEight cloud platform.
Frequency	Determined by external system. Suggested implementation to trigger an update whenever an available rate is changed for a project in the ERP or Pay system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Tens or hundreds of records depending on the complexity of pay rates and union or vendor agreements. Entity delta – Less than 100 records.
APIM Name	Upsert ProjectCraftRateCodes
Application Starting Version	18.3

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectId[2]	String	200		Yes
1	CraftId[2]	String	50		Yes
1	StraightTimeRate	Decimal	16,5		No
1	OverTimeFactor	Decimal	16,5		No
1	OverTimeRate	Decimal	16,5		No
1	DoubleTimeFactor	Decimal	16,5		No
1	DoubleTimeRate	Decimal	16,5		No
1	IsAvailable	Boolean	N/A		No
1	UseBaseWageFactors	Boolean	N/A		No

1 - For numeric data types, precision is given as total digits allowed in the field and the number of those digits that exist to the right of the decimal. For example, 16,5 represents a total of 16-digits allowed in the field with 5 of those digits existing as decimal places and 11 digits on the left of the decimal. The decimal is not counted as a digit.

2 - Natural Key field.

INEIGHT

Field Descriptions

Name	Description	Example
ProjectId	ID of the project in which the craft should be made available.	103444
CraftId	ID of the craft to make available to the project.	PA00
StraightTimeRate	Hourly rate that should be given to the craft in the project. This field is left blank when using the value in the master data record for the craft.	15.00
OverTimeFactor	Multiplier against straight time used to calculate overtime. If specifying a value for the record, use this field or OverTimeRate, but not both. This field is left blank when using the value in the master data record for the craft.	1.5
OverTimeRate	A specific rate to apply to overtime hours. If specifying a value for the record, use this field or OverTimeFactor, but not both. This field is left blank when using the value in the master data record for the craft.	20.00
DoubleTimeFactor	Multiplier against straight time used to calculate Double Time. If specifying a value for the record, use this field or DoubleTimeRate, but not both. This field is left blank when using the value in the master data record for the craft.	2.0
DoubleTimeRate	A specific rate to apply to double time hours. If specifying a value for the record, use this field or DoubleTimeFactor, but not both. This field is left blank when using the value in the master data record for the craft.	25.00
IsAvailable	Determines the status of the craft rate code and whether it can be assigned to the project.	true
UseBaseWageFactors	Determines if factors or rates are used when calculating the actual rates against hours worked. This field is left blank when using the value in the master data record for the craft.	false

Project Craft

Error Messages

The following error messages are generated by the InEight cloud platform and products for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity logic errors are performed internally in the InEight cloud platform and products to look for specific business rule or data integrity issues record-by-record. Failures with entity logic validations only cause the individual record to cease processing and are written to internal logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. Possible [Required Field] options: ProjectId CraftId	400	The request is invalid. The [Required Field] is required.
Entity Logic Validation	Payload has invalid Project.	200	Ignoring ProjectCraft having CraftId {CraftId} as it is having invalid ProjectId {ProjectDisplayId}
Entity Logic Validation	The CraftId in the payload doesn't match with any Craft in the system by CraftId or SourceSystemId.	200	Ignoring ProjectCraft as it is having invalid CraftId {CraftId}
Entity Logic Validation	StraightTimeRate is null and OverTimeFactor / DoubleTimeFactor / OverTimeRate / DoubleTimeRate are not null.	200	Ignoring ProjectCraft having CraftId {CraftId} because ST rate is not passed
Entity Logic Validation	Any of the ProjectCraft's TimeFactor or TimeRate is a negative value.	200	Ignoring ProjectCraft having '{CraftId}' because Rates/Factors should not accept -ve value

Sample JSON

```
[
    {
```
Creates a new record with specific project values.

```
    "ProjectId": "103361",
    "CraftId": "PA00",
    "StraightTimeRate": 8.00,
    "OverTimeFactor": 0.0,
    "OverTimeRate": 10.50,
    "DoubleTimeFactor": 0.0,
    "DoubleTimeRate": 13.00,
    "IsAvailable": true,
    "UseBaseWageFactors": false
    },{
```

Creates new records using values inherited from the Craft master data.

```
    "ProjectId": "103361",
    "CraftId": "LA1L",
    },{
    "ProjectId": "103361",
    "CraftId": "LA1M",
    }
]
```

Verification

From the main menu, select the project that records were added to, and then select **Operational rate codes**.

New or updated records should show in the UI.

	Rate cod...	Vendor	Union	Override...	ST rate	Use base...	OT factor	DT factor	OT rate	DT rate
☐	LA1L			☐	$10.00000	☐	1.50000	2.00000	$15.00000	$20.00000
☐	LA1M			☐	$12.00000	☑	1.50000	2.00000	$18.00000	$24.00000
☐	PA00			☑	$8.00000	☐	1.31250	1.62500	$10.50000	$13.00000

Project Employees (Post)

Direction	To the InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Hundreds or thousands of records. Entity delta – Tens or hundreds of records.
APIM Name	Upsert ProjectEmployees
Application Starting Version	18.2

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectId[2]	String	200		Yes
1	EmployeeId[2]	String	50		Yes
1	StraightTimeRate	Number	16,5		No
1	OverTimeFactor	Number	16,5		No
1	OverTimeRate	Number	16,5		No
1	DoubleTimeFactor	Number	16,5		No
1	DoubleTimeRate	Number	16,5		No
1	SourceSystemName	String	50		No
1	UseBaseWageFactors	Boolean	N/A		No
1	StartDate[3]	String	25		No
1	EndDate[3]	String	25		No
1	SourceSystemId	String	50		No
1	UoMName	String	255		No
1	ProjectCraftSourceSystemId	String	50		No
1	ProjectRateCodeSourceSystemId	String	50		No
1	IsActive	Boolean	N/A		No
1	BillingClass	String	255		No
1	CustomField1	String	255		No
1	ProjectEmployeeBillingClassSourceSystemId	String	100		No

1 - For numeric data types, precision is given as total digits allowed in the field and the number of those digits that exist to the right of the decimal. For example, 16,5 represents a total of 16-digits allowed in the field with 5 of those digits existing as decimal places and 11 digits on the left of the decimal. The decimal is not counted as a digit.

2 - Natural Key field.
3 - The data format for Date/Time fields is YYYY-MM-DDTHH:MM:SS+hhmm, where hhmm is the time zone offset. If the time is already converted to UTC, then the offset will be +0000

Field Descriptions

Name	Description	Example
ProjectId	SourceSystemId of the project that the employees are assigned to. In the InEight UI, this field is labeled External Project Id.	106774
EmployeeId	SourceSystemId of the employee who is being assigned to the project.	67346774
StraightTimeRate	Base rate to apply to the employee for calculating estimated cost of work on the specified project. Providing a value in this field will set a specific rate to use for the employee only on the assigned project. If this field is left blank, the system will use the equivalent value in the master data record for the employee.	11.75
OverTimeFactor	Adjustment percentage factor applied to the StraightTimeRate to determine the rate for overtime. Use a factor or use a set rate, but not both. If using factors instead of rates, set UseBaseWageFactors to true. Providing a value in this field will set a specific factor to use for the employee only on the assigned project. If this field is left blank, the system will use the equivalent value in the master data record for the employee.	1.5
OverTimeRate	Specific rate to apply to the employee for overtime. Use a factor or use a set rate, but not both. If using rates instead of factors, set UseBaseWageFactors to false. Providing a value in this field will set a specific rate to use for the employee only on the assigned project. If this field is left blank, the system will use the equivalent value in the master data record for the employee.	17.50
DoubleTimeFactor	Adjustment percentage factor applied to the StraightTimeRate to determine the rate for double time. Use a factor or use a set rate, but not both. If using factors instead of rates, set UseBaseWageFactors to true. Providing a value in this field will set a specific factor to use for the employee only on the assigned project. If this field is left blank, the system will use the equivalent value in the master data record for the employee.	2.0
DoubleTimeRate	The specific rate to apply to the employee for Double Time. Use a factor or use a set rate, but not both. If using rates instead of factors, set 'UseBaseWageFactors' to false. Providing a value in this field will set a specific rate to use for the employee only on the assigned project. If this field is left blank, the system will use the equivalent value in the master data record for the employee.	23.50

Name	Description	Example
SourceSystemName	Name of the system of record that is assigning employees to projects.	D365
UseBaseWageFactors	Determines if the system should use wage adjustment factors instead of specific rates for overtime and double time. This field should be left blank when using the values in the master data record for the employee.	false
StartDate	Effective first date that the employee should be considered active on the project and available for selection in application functions. This date cannot precede the start date of the master data record for the employee.	2019-01-01
EndDate	Last effective date that the employee should be considered active on the project and available for selection in application functions. This date cannot exceed the end date of the master data record for the employee.	2021-12-31
SourceSystemId	Do not use this field. A unique record is created internally by InEight for the association of an employee and a project through their respective SourceSystemId fields.	
UoMName	Unit of measure used in conjunction with the rate applied specifically to the employee for the assigned project. If a value is not provided for this field, the system will use the unit of measure from the Employee master record or the craft associated to the employee. The value provided in this field must match Name or SourceSystemId of an existing Unit of Measure record.	HR
ProjectCraftSourceSystemId	Used to set a specific craft for the employee while they are assigned to the project. For example, an employee that typically works a trade can be assigned as a supervisor on a temporary assignment to the project. If a specific craft is sent in this field with no other specific rate information provided, the rate values for the employee is determined from the craft value in this field. The value in this field must match a record that was previously provided in the ProjectCraft integration.	PCPTR
ProjectRateCodeSourceSystemId	Unique identifier of the ProjectRateCode entity associated to the employee.	946e61cb-6815-45d1-90da-d72a0e001ee0
IsActive	This field is not recommended for use. Sending a value of false in this field will cause the record to be soft deleted from InEight and unrecoverable without assistance from InEight Professional Services.	true
BillingClass	Project level free-form text value entered manually via the UI or using the integration.	Pile-driver foreman
CustomField1	Project level free-form text value entered manually via the UI or using the integration.	Custom value 1

Name	Description	Example
ProjectEmployeeBillingClassSourceSystemId	Unique ID of the system of record that is assigning the project employee billing class.	3a37eee1-1bb5-4f88-bfb8-e6e309499309

Error Messages

The following error messages are generated by the InEight cloud platform and products for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity logic errors are performed internally in the InEight cloud platform and products to look for specific business rule or data integrity issues record-by-record. Failures with entity logic validations only cause the individual record to cease processing and are written to internal logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. Possible [Field Name] options: ProjectId EmployeeId	400	The request is invalid. The [Field Name] field is required.
API Validation	Provided data exceeds the string size. Possible [Field Name] options: ProjectId EmployeeId SourceSystemName SourceSystemId UnionSourceSystemId UoMName ProjectCraftSourceSystemId	400	The request is invalid. The field [Field Name] must be a string with a maximum length of [Field Length].
Entity Logic Validation	Payload has invalid Project.	200	Ignoring ProjectEmployee having EmployeeId '{EmployeeId}' as it is having invalid ProjectId '{ProjectId}'
Entity Logic Validation	Payload has invalid Employee.	200	Ignoring ProjectEmployee due to Invalid Employee {EmployeeId}
Entity Logic Validation	Payload has a ProjectEmployee with EmployeeId NULL.	200	Ignoring ProjectEmployee as it is having invalid EmployeeId {EmployeeId}
Entity Logic Validation	Project is closed.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Project is closed
Entity Logic Validation	Employee EndDate is in the past.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because employee with End date in past cannot be assigned to a project.

API/Entity Logic	Condition	Code	Message
Entity Logic Validation	ProjectCraftSourceSystemId is either unavailable or invalid or NULL.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} as it is having invalid/unavailable ProjectCraftSourceSystemId {ProjectCraftSourceSystemId}
Entity Logic Validation	ProjectRateCodeSourceSystemId is either unavailable or invalid.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} as it is having invalid/unavailable ProjectRateCodeSourceSystemId {ProjectRateCodeSourceSystemId}
Entity Logic Validation	Invalid or blank UOMName.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} as it is having invalid UOMName/UOMSourcesystemId {UOMId}
Entity Logic Validation	ProjectEmployee currency does not match with ProjectCraft or ProjectRateCode or Employee currency.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} as it is having invalid CurrencyId {CurrencyId}
Entity Logic Validation	ProjectEmployee currency does not match with one of the available currencies for Project.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Employee currency did not match with project currencies
Entity Logic Validation	ProjectEmployee has invalid BillingClass	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} as it is having invalid/unavailable ProjectEmployeeBillingClassId {BillingClass}
Entity Logic Validation	ProjectEmployee's BillingClass belongs to a different Project	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because ProjectEmployeeBillingClassId {BillingClass} is belongs to different project
Entity Logic Validation	Employee and BillingClass vendors do not match	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Employee vendor and Billing class vendor are not matching
Entity Logic Validation	Project StartDate is greater than Employee EndDate or Employee StartDate is greater than Project EndDate	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Master data employee start date and end date is not overlapping between Project start date and end dates.
Entity Logic Validation	ProjectEmployee StartDate greater than EndDate	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Project employee end date is less than the Project employee start date.
Entity Logic Validation	Employee StartDate greater than ProjectEmployee StartDate or Employee EndDate less than ProjectEmployee EndDate	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Project employee start and end date should be within Master Employee start and end date.
Entity Logic Validation	Employee already exists in the Project.	200	Ignoring ProjectEmployee as it is having duplicate EmployeeId {EmployeeSourceSystemId}
Entity Logic Validation	ProjectCraft is not associated with the Project.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because ProjectCraftSourcesystemId {ProjectCraftSourceSystemId} is not associated with the project

API/Entity Logic	Condition	Code	Message
Entity Logic Validation	Employee VendorId not matching with project craft VendorId.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Employee vendor and ProjectCraft vendor are not matching
Entity Logic Validation	ProjectRateCode is not associated with the Project.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because ProjectRateCodeSourcesystemId {ProjectRateCodeSourcesystemId} is not associated with the project
Entity Logic Validation	Employee VendorID not matching with project RateCode VendorID.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because Employee vendor and ProjectRateCode vendor are not matching
Entity Logic Validation	StraightTimeRate is null and OverTimeFactor / DoubleTimeFactor / OverTimeRate / DoubleTimeRate are not null.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} because ST rate is not passed
Entity Logic Validation	Any of the ProjectEmployee's TimeFactor or TimeRate is a negative value.	200	Ignoring ProjectEmployee having '{EmployeeId}' because Rates/Factors should not accept -ve value
Entity Logic Validation	Payload project employee UOM Id is not found.	200	Ignoring ProjectEmployee having EmployeeId {EmployeeId} as it is having invalid UOMName {UOMName}
Entity Logic Validation	Payload project employee is not active.	200	Ignoring inactive ProjectEmployee having EmployeeId {EmployeeId} for adding
Entity Logic Validation	Payload project employee is not active and the transaction could not succeed.	200	Ignoring ProjectEmployee having EmployeeId '{EmployeeId}' as delete is not allowed
Entity Logic Validation	ProjectEmployee update transaction could not succeed in the system.	200	Ignoring invalid ProjectEmployee having EmployeeId '{EmployeeId}' for update

Sample JSON

```
[
  {
  "ProjectId": "107654",
  "EmployeeId": "36546777",
  "StraightTimeRate": 12.0000,
  "OverTimeFactor": 0,
  "OverTimeRate": 18.0000,
  "DoubleTimeFactor": 0,
  "DoubleTimeRate": 24.0000,
  "SourceSystemName": "SAP",
  "UseBaseWageFactors": false,
  "StartDate": "2019-01-01",
  "EndDate": "2021-12-31",
  "SourceSystemId": "",
  "UoMName": "HR",
  "ProjectCraftSourcesystemId": "",
  "IsActive": true
  },{
  "ProjectId": "107654",
```

```
    "EmployeeId": "54646444",
    "StraightTimeRate": 10.0000,
    "OverTimeFactor": 1.5,
    "OverTimeRate": 0,
    "DoubleTimeFactor": 1.75,
    "DoubleTimeRate": 0,
    "SourceSystemName": "SAP",
    "UseBaseWageFactors": true,
    "StartDate": "2019-01-01",
    "EndDate": "2021-12-31",
    "SourceSystemId": "",
    "UoMName": "HR",
    "ProjectCraftSourcesystemId": "JCPTR",
    "IsActive": true,
    "BillingClass": "Crawler Tractor",
    "ProjectEmployeeBillingClassSourceSystemId": "3a37eee1-1bb5-4f88-bfb8-
e6e309499309"
    }
]
```

Verification

A master data Employee record can be deleted even if the employee is associated to a project. When deleting an employee via the UI, the system checks if the resource is active or inactive on a project before the record is deleted. If an employee is also a project employee for any project, based on end date of employee versus project end date versus current date, additional information is given to the user before the Employee record is deleted.

> **IMPORTANT:** When a master data Employee record is deleted via the integration, the system does not check if the employee is associated with a project before deleting the record.

Employees that have been assigned to a project are shown in the UI. Select the project to verify, then from the main menu go to **Assigned operational resources**.

Employees and equipment assigned to the project are shown on their respective tabs.

Project Employees (Get)

This API uses the standard InEight asynchronous GET pattern (refer to *"Getting Started with InEight Integrations – Developer Guide"* for details) and returns 500 records for each request. Use the OData query parameter $skip to continue to make subsequent calls until the returned record set falls below a count of 500.

Direction	From InEight
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	All employees assigned to a specified ProjectId
APIM Name	List ProjectEmployees
Application Starting Version	19.1

Supported Filters

This API supports standard OData query parameters of $expand, $filter, $select, $orderby, $top, $skip, and $count.

Filter Name	Data Type	Description
ProjectId	String	This is a required parameter. Use the SourceSystemId of a project to receive all records of Employees assigned to it.

Fields

See Field in the Project Employees (Post) API for details.

Field Descriptions

See Field Description in the Project Employees (Post) API for details.

Sample JSON File

See Sample in the Project Employees (Post) API for details.

Verification

See Verification in the Project Employees (Post) API for details.

Equipment Rate Codes

Direction	To the InEight cloud platform.
Frequency	Determined by external system. Suggested implementation to trigger an update whenever an available rate is changed for a project in the ERP or Pay system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Tens or hundreds of records depending on the complexity of pay rates and union or vendor agreements. Entity delta – Less than 100 records.
APIM Name	Upsert ProjectEquipmentRateCodes
Application Starting Version	18.3

Fields

Depth	Name	Type	Precision	Parent	Req.
1	ProjectId	String	200		Yes
1	EquipmentTypeId	String	250		Yes
1	UnitCost	Decimal	16,5		No
1	IsAvailable	Boolean	N/A		No

Field Descriptions

Name	Description	Example
ProjectId	ID of the project that the equipment type is being assigned.	103444
EquipmentTypeId	ID of the equipment type that is being assigned to the project.	04-01
UnitCost	Unit cost per unit of measure given in the equipment type master data record. This field is left blank except when overriding the master data record unit cost for this equipment type in the assigned project.	85.00
IsAvailable	Determines the status and whether the equipment rate code can be assigned to the project.	true

Error Messages

The following error messages are generated by InEight for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API Validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity Logic errors are performed internally within InEight and look for specific business rule or data integrity issues record-by-record. Failures with Entity Logic validations only cause the individual record to cease processing and are written to internal InEight logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. [Field Name] options are: • ProjectId • EquipmentTypeId	400	The request is invalid. The [Field Name] Field is required.
Entity Logic Validation	Payload has more than one record with same ProjectId and EquipmentTypeId	200	Duplicate combination of ProjectDisplay '{ProjectId}' and EquipmentTypeSourcesystemId '{EquipmentTypeId}' found. Aborting the process.
Entity Logic Validation	Invalid Project used in the payload	200	Ignoring ProjectEquipmentType having EquipmentTypeId {EquipmentTypeId} as it is having invalid ProjectId {ProjectId}
Entity Logic Validation	Invalid EquipmentTypeId in the payload	200	Ignoring ProjectEquipmentType as it is having invalid EquipmentTypeId {EquipmentTypeId}
Entity Logic Validation	Negative UnitCost value used in the payload	200	Ignoring ProjectEquipmentType having EquipmentTypeId {EquipmentTypeId}. Because UnitCost value '{UnitCost}' should not have -ve values
Entity Logic Validation	Payload has a Project that is closed.	200	Ignoring ProjectEquipmentType having EquipmentTypeId {EquipmentTypeId} as the project is closed.
Entity Logic Validation	Payload has inactive records while inserting	200	Ignoring inactive ProjectEquipmentType having EquipmentTypeId {EquipmentTypeId}
Entity Logic Validation	EquipmentType currency not matching with Project currency.	200	Ignoring ProjectEquipmentType having EquipmentTypeId {EquipmentTypeId} for adding. Because Equipmenttype currency not equal to project currency
Entity Logic Validation	ProjectEquipmentType transaction could not succeed.	200	Ignoring invalid ProjectEquipmentType having EquipmentTypeId {EquipmentTypeId} for update

Entity Logic Validation	Valid Payload	202	{
			    Logs: [ ]
			}

Sample JSON

```
[
    {
    "ProjectId": "103367",
    "EquipmentTypeId": "DZR5T",
    "UnitCost": 0.0,
    "IsAvailable": true
    },{
    "ProjectId": "103367",
    "EquipmentTypeId": "DZR10T",
    "UnitCost": 0.0,
    "IsAvailable": true
    },{
    "ProjectId": "103367",
    "EquipmentTypeId": "DZR15T",
    "UnitCost": 0.0,
    "IsAvailable": true
    },{
    "ProjectId": "103367",
    "EquipmentTypeId": "DZR20T",
    "UnitCost": 0.0,
    "IsAvailable": true
    }
]
```

Project Equipments (Post)

NOTE:	If the equipment being assigned to a project does not have the same currency as either the primary currency or any of the allowed currencies for the project, the equipment assignment will fail.

Direction	To the InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Potentially thousands of records depending on project size.
APIM Name	Upsert ProjectEquipments
Application Starting Version	18.2

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectNumber	String	200		Yes
1	EquipmentNumber	String	100		Yes
1	IsActive	Boolean	N/A		No
1	UnitCost	Decimal	11,5		No
1	SourceSystemName	String	50		No
1	SourceSystemId	String	50		No
1	IsAvailableDailyPlan	Boolean	N/A		No
1	EquipmentProjectStartDate	String	Date		No
1	EquipmentProjectEndDate	String	Date		No
1	ContractReference	String	50		No
1	ContractLineNumberReference	String	20		No
1	ContractURL	String	250		No
1	BillingClass	String	255		No
1	ProjectEquipmentBillingClassSourceSystemId	String	100		No

1 - For numeric data types, precision is given as total digits allowed in the field and the number of those digits that exist to the right of the decimal. For example, 16,5 represents a total of 16-digits allowed in the field with 5 of those digits existing as decimal places and 11 digits on the left of the decimal. The decimal is not counted as a digit.

Field Descriptions

Name	Description	Example
ProjectNumber	DisplayId of the project where the equipment is being assigned.	103361
EquipmentNumber	SourceSystemId of the equipment that is being assigned to the project.	3444555
IsActive	This field is not recommended for use. Sending a value of false in this field will cause the record to be soft deleted from InEight and unrecoverable without assistance from InEight Professional Services.	true
UnitCost	Sets the hourly rate of the equipment specifically for the project assignment. If this value is not provided, the unit cost is inherited from the equipment type associated to the equipment record.	95.25
SourceSystemName	Name of the system where the assignment is being managed.	JDE
SourceSystemId	Unique internal ID of the association record from the system where the associations are managed.	3444.103361
IsAvailableDailyPlan	Determines if the equipment is available for selection in a Daily Plan in the Plan and Progress applications	True
EquipmentProjectStartDate	First date on which the equipment record is available to the project.	2019-25-11 00:00:00
EquipmentProjectEndDate	Last date on which the equipment record is available to the project.	2022-01-06 00:00:00
ContractReference	Optional reference ID for a rental agreement contract	7177373
ContractLineNumberReference	If the equipment has been rented and a ContractReference is provided, this field can be used to indicate a specific line item in the contract that contains the equipment information.	1
ContractURL	Optional field that can be used to provide a URL to an external system where the rental agreement contract can be viewed. This information is only shown in the Assigned operational resources of a project.	https://Contracts?projectId=444#/contract/header/98200001464
BillingClass	Project level free-form text value entered manually via the UI or using the integration.	Crawler Tractors
ProjectEquipmentBillingClassSourceSystemId	Unique ID of the system of record that is assigning the project equipment billing class.	3a37eee1-1bb5-4f88-bfb8-e6e309499309

Error Messages

The following error messages are generated by InEight for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API Validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity Logic errors are performed internally within InEight and look for specific business rule or data integrity issues record-by-record. Failures with Entity Logic validations only cause the individual record to cease processing and are written to internal InEight logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. [Field Name] options are: • ProjectNumber • EquipmentNumber	400	The request is invalid. The [Field Name] Field is required.
Entity Logic Validation	Payload has more than one record with same ProjectNumber and EquipmentNumber	200	Duplicate combination of ProjectDisplay '{ProjectNumber}' and EquipmentSourcesystemId '{EquipmentNumber}' found. Aborting the process.
Entity Logic Validation	Invalid Project used in the payload	200	Ignoring ProjectEquipment having EquipmentNumber{EquipmentNumber} as it is having invalid ProjectNumber {ProjectDisplay}
Entity Logic Validation	Invalid Equipment used in the payload	200	Ignoring ProjectEquipment as it is having invalid EquipmentId {EquipmentNumber}
Entity Logic Validation	Payload EndDate occurring before StartDate	200	Ignoring ProjectEquipment having EquipmentNumber {EquipmentNumber} because End date cannot occur before Start date
Entity Logic Validation	Invalid ProjectEquipmentBillingClass in the payload	200	Ignoring ProjectEquipment having EquipmentNumber{EquipmentNumber} as it is having invalid/unavailable ProjectEquipmentBillingClass {ProjectEquipmentBillingClassSourceSystemId}
Entity Logic Validation	ProjectEquipment's BillingClass belongs to a different Project	200	Ignoring ProjectEquipment having EquipmentNumber {EquipmentNumber} because ProjectEquipmentBillingClass {ProjectEquipmentBillingClassSourceSystemId} is belongs to different project
Entity Logic Validation	Payload has a Project that is closed.	200	Ignoring ProjectEquipment having EquipmentNumber{EquipmentNumber} as the project is closed.
Entity Logic Validation	Payload has inactive records while inserting	200	Ignoring inactive ProjectEquipment having EquipmentNumber {EquipmentNumber} for adding

Entity Logic Validation	Equipment currency not matching with Project currency.	200	Ignoring ProjectEquipment having EquipmentNumber {EquipmentNumber} because Equipment currency did not match with project currencies
Entity Logic Validation	Equipment vendor and Billing class does not match	200	Ignoring ProjectEquipment having EquipmentNumber {EquipmentNumber} because Equipment vendor and Billing class vendor are not matching
Entity Logic Validation	EquipmentType currency doesn't match with Project currency	200	Ignoring ProjectEquipment having EquipmentNumber {EquipmentNumber} because Equipmenttype currency not equal to project currency
Entity Logic Validation	ProjectEquipment transaction could not succeed.		Ignoring invalid ProjectEquipment having EquipmentNumber {EquipmentNumber} for update

Entity Logic Validation	Valid Payload	202	{
			    Logs: [ ]
			}

Sample JSON

```
[
    {
    "ProjectNumber": "103361",
    "EquipmentNumber": "101145",
    "IsActive": true,
    "UnitCost": 0.0,
    "SourceSystemName": "JDE",
    "SourceSystemId": "",
    "IsAvailableDailyPlan": true,
    "EquipmentProjectStartDate": "2016-25-11 00:00:00",
    "EquipmentProjectEndDate": "2022-01-09 00:00:00",
    "ContractReference": "98200001464",
    "ContractLineNumberReference": "2",
    "ContractURL": https://Contracts?projectId=444#/contract/header/98200001464
    "BillingClass": "Crawler Tractor",
    "ProjectEquipmentBillingClassSourceSystemId": "3a37eee1-1bb5-4f88-bfb8-
e6e309499309"
    }
]
```

Verification

A master data Equipment record can be deleted even if the equipment is associated to a project. When deleting an equipment via the UI, the system checks if the resource is active or inactive on a project before the record is deleted. If the equipment is also a project equipment for any project, based on end date of equipment versus project end date versus current date, additional information is given to the user before the Equipment record is deleted.

IMPORTANT:	When a master data Equipment record is deleted via the integration, the system does not check if the equipment is associated with a project before deleting the record.

Equipment that has been assigned to a project is shown in the InEight UI. Select the project to verify, and then from the main menu click **Assigned operational resources**.

Apply filtering to the list of records in the grid to verify specific equipment type records.

Project Equipments (Get)

This API uses the standard InEight asynchronous GET pattern (Refer to "Getting Started with InEight Integrations – Developer Guide" for details) and returns 500 records for each request. Use the OData query parameter $skip to continue to make subsequent calls until the returned record set falls below a count of 500.

Direction	From InEight
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	All equipment assigned to a specified Projectnumber
APIM Name	List ProjectEquipments
Application Starting Version	19.1

Supported Filters

This API supports standard OData query parameters of $expand, $filter, $select, $orderby, $top, $skip, and $count.

Filter Name	Data Type	Description
ProjectNumber	String	DisplayId of the project where the equipment is being assigned.
EquipmentNumber	String	SourceSystemId of the equipment that is being assigned to the project.

Fields

See Field in the Project Equipment_Import (Post) API for details.

Field Descriptions

See Field Description in the Project Equipment_Import (Post) API for details.

Sample JSON File

See Sample in the Project Equipment_Import (Post) API for details.

Verification

See Verification in the Project Equipment_Import (Post) API for details.

Cost Centers

Direction	To the InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Potentially tens of records. Entity changes – Cost Centers are not expected to change frequently.
APIM Name	Upsert CostCenters
Application Starting Version	18.2

Fields

Depth	Name	Type	Precision	Parent	Req.
1	CostCenterDisplayId	String	50		Yes
1	CostCenterDescription	String	250		Yes
1	CostCenterTypeDisplayId	String	50		Yes
1	IsActive	Boolean	N/A		No
1	SourceSystemId	String	50		No
1	SourceSystemName	String	50		No

Field Descriptions

Name	Description	Example
CostCenterDisplayId	Unique identifier that is publicly displayed and understood. This field can be the same as SourceSystemId or it can contain a different value.	PLAB
CostCenterDescription	Full public description of the cost center.	Project Labor
CostCenterTypeDisplayId	DisplayId of a cost center type that exists in the InEight cloud platform.	PCOST
IsActive	Sending a value of false in this field causes the record to be soft deleted from the InEight cloud platform. If a value is not provided, the default value true is used.	true
SourceSystemId	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not display in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	10-114

Name	Description	Example
SourceSystemName	Name of the external system that is using the integration. Work with InEight to provision a unique value for this field.	JDE

Error Messages

The following error messages are generated by InEight for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API Validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity Logic errors are performed internally within InEight and look for specific business rule or data integrity issues record-by-record. Failures with Entity Logic validations only cause the individual record to cease processing and are written to internal InEight logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. [Field Name] options are: • CostCenterDisplayId • CostCenterDescription • CostCenterTypeDisplayId	400	The request is invalid. The [Field Name] Field is required.
Entity Logic Validation	Payload has more than one record with same SourceSystemId or DisplayId	200	Duplicate SourceSystemId/DisplayId {SourceSystemIds} found. Aborting the process.
Entity Logic Validation	Payload has conflict with uniqueness on SourceSystemId or DisplayId with existing data	200	Ignoring CostCenter '{DisplayId}' as it is having duplicate SourceSystemId or DisplayId
Entity Logic Validation	Payload has inactive records while inserting	200	Ignoring inactive Costcenter '{DisplayId}' for adding
Entity Logic Validation	Payload is inactivating a CostCenter that has employees or equipments associated.	200	Ignoring CostCenter '{DisplayId}' for deleting as it is associated with employees and pieces of equipment. So you must remove all associated employees and pieces of equipment from this CostCenter before it can be deleted.

Sample JSON

```
[
    {
    "CostCenterDisplayId": "PLAB",
    "CostCenterDescription": "Project Labor",
    "CostCenterTypeDisplayId": "PCOST",
    "IsActive": true,
    "SourceSystemId": "PLAB",
    "SourceSystemName": "JDE"
    },{
    "CostCenterDisplayId": "PMAIN",
    "CostCenterDescription": "Project Maintenance",
    "CostCenterTypeDisplayId": "PCOST",
    "IsActive": true,
    "SourceSystemId": "PMAIN",
    "SourceSystemName": "JDE"
    },{
    "CostCenterDisplayId": "POVR",
    "CostCenterDescription": "Project Overhead",
    "CostCenterTypeDisplayId": "PCOST",
    "IsActive": true,
    "SourceSystemId": "POVR",
    "SourceSystemName": "JDE"
    }
]
```

Verification

To verify cost centers are created, from the main menu go to Master data libraries > **Cost Centers**.

Project Vendors (Import)

NOTE:	Before using this API, make sure that every vendor being imported or exported from the InEight cloud platform is already present in the master list of vendors in InEight Platform, otherwise the import and export APIs will fail.

Direction	To the InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Potentially thousands of records depending on project size.
APIM Name	Upsert ProjectVendors
Application Starting Version	22.2

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectDisplayID	String	200		No
1	VendorCode	String	250		No
1	VendorSourceSystemID	String	250		Yes
1	ProjectSourceSystemID	String	200		Yes
1	IsActive	String	Boolean		No

Field Descriptions

Name	Description	Example
ProjectDisplayID	Unique identifier for each project that is publicly recognizable. This field shows in normal use of InEight applications. The value sent in this field can match SourceSystemId if there is no need to use a public versus private unique identifier.	103361
VendorCode	Vendor code of the vendor if a user is assigned to the vendor.	3444555

Name	Description	Example
VendorSourceSystemID	This field is not recommended for use. Sending a value of false in this field will cause the record to be soft deleted from InEight and unrecoverable without assistance from InEight Professional Services. Unique Identifier for each vendor created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not show in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	1000445467
ProjectSourceSystemID	Unique Identifier for each project created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not show in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	32312243455468678
IsActive	Boolean value that indicates whether the vendor is active or not in the InEight system.	True

Error Messages

The following error messages are generated by InEight for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API Validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity Logic errors are performed internally within InEight and look for specific business rule or data integrity issues record-by-record. Failures with Entity Logic validations only cause the individual record to cease processing and are written to internal InEight logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. [Field Name] options are: • ProjectSourceSystemId • VendorSourceSystemId	400	The request is invalid. The [Field Name] Field is required.
Entity Logic Validation	Payload has duplicate ProjectVendor mapping.	200	Duplicate combination of ProjectSourceSystemId '{ProjectSourceSystemId}' and VendorSourceSystemId '{VendorSourceSystemId}' found. Aborting the process.
Entity Logic Validation	Payload has invalid project	200	Ignoring ProjectVendor having Vendor '{VendorSourceSystemId}' as it is having invalid Project '{ProjectSourceSystemId}'

Entity Logic Validation	Payload has invalid vendor	200	Ignoring ProjectVendor having ProjectSourceSystemId '{ProjectSourceSystemId}' as it is having invalid Vendor '{VendorSourceSystemId}'
Entity Logic Validation	Project and Vendor mapping in the payload already exists in the system.	200	Ignoring ProjectVendor having Vendor {VendorSourceSystemId} because this Vendor is already mapped to the given Project
Entity Logic Validation	Payload has inactive records while inserting	200	Ignoring inactive ProjectVendor having Vendor '{VendorSourceSystemId}' for adding

Sample JSON

```
[
  {
    "ProjectDisplayId": "Bentway Industrial",
    "VendorCode": "65532343",
    "VendorSourceSystemId": "5657777",
    "ProjectSourceSystemId": "6575663434",
    "IsActive": true
  }
]
```

Verification

The API only works if the vendors being assigned to a project are already included in Master data libraries > Vendors before using the API in APIM.

Project Vendors (Get)

IMPORTANT:	Before using this API, make sure that every vendor being imported or exported from the InEight cloud platform is already present in the master list of vendors in InEight Platform, otherwise the import and export APIs will fail.

Direction	To the InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Potentially thousands of records depending on project size.
APIM Name	List ProjectVendors
Application Starting Version	22.2

Fields

See Field in the Project Vendors_Import (Post) API for details.

Field Descriptions

See Field Description in the Project Vendors_Import (Post) API for details.

Sample JSON File

See Sample in the Project Vendors_Import (Post) API for details.

Verification

See Verification in the Project Vendors_Import (Post) API for details.

Project Settings (Post)

Direction	To the InEight cloud platform from an external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – Hundreds or thousands of records. Entity delta – Tens or hundreds of records.
APIM Name	Upsert ProjectSettings
Application Starting Version	20.11

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectSourceSystemId	String	200		Yes
1	ProjectSourceSystemName	String	50		Yes
1	ProjectDisplayId	String	200		No
1	GlobalOptions	Array			No
2	CostItemIntegrationBehavior	String	10	GlobalOptions	No
2	IsAdvancedWorkAndSchedulingFeaturesEnabled	Boolean	N/A	GlobalOptions	No
2	IsTemplateProject	Boolean	N/A	GlobalOptions	No
2	IsValidateEquipmentToProjectCurrency	Boolean	N/A	GlobalOptions	No
1	ProjectInformationTags	Array			No
2	ProjectInformationTag1	String	500	ProjectInformationTags	No
2	ProjectInformationTag2	String	250	ProjectInformationTags	No
2	ProjectInformationTag3	String	250	ProjectInformationTags	No
2	ProjectInformationTag4	String	250	ProjectInformationTags	No
2	ProjectInformationTag5	String	250	ProjectInformationTags	No
2	ProjectInformationTag6	String	100	ProjectInformationTags	No
2	ProjectInformationTag7	String	250	ProjectInformationTags	No
2	ProjectInformationTag8	String	250	ProjectInformationTags	No
2	ProjectInformationTag9	String	255	ProjectInformationTags	No
2	ProjectInformationTag10	String	250	ProjectInformationTags	No
2	ProjectInformationTag11	String	250	ProjectInformationTags	No
2	ProjectInformationTag12	String	250	ProjectInformationTags	No

Depth	Name	Type	Precision[1]	Parent	Req.
2	ProjectInformationTag13	String	250	ProjectInformationTags	No
2	ProjectInformationTag14	String	250	ProjectInformationTags	No
2	ProjectInformationTag15	String	250	ProjectInformationTags	No
1	CustomLists	Array			No
2	ListName	String	250	CustomLists	No
2	FieldValues	String	250	CustomLists	No
1	FuelTypes	Array			No
2	IsEstimatingResourceMapped	Boolean	N/A	FuelTypes	No
2	FuelTypeDisplay	String	50	FuelTypes	No
2	UOMSourceSystemId	String	200	FuelTypes	No
2	CurrencyCode	String	50	FuelTypes	No
2	UnitCost	Number	16,5	FuelTypes	No
2	AccountCodeReference	String	250	FuelTypes	No
2	IsActive	Boolean	N/A	FuelTypes	No
2	SourceSystemID	String	255	FuelTypes	No
2	SourceSystemName	String	50	FuelTypes	No

Field Descriptions

Name	Description	Example
ProjectSourceSystemId	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not show in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	a75d816a-456d-4e94-b4d6-2cfa4fc447de
ProjectSourceSystemName	Name of the external system that is using the integration. Work with InEight to provision a unique value for this field.	SAP
ProjectDisplayId	Unique identifier that is publicly recognizable. This field is displayed in normal use of InEight applications. The value sent in this field can match SourceSystemId if there is no need to use a public versus private unique identifier.	103361
GlobalOptions	Used to set Global Options of the project.	

Name	Description	Example
CostItemIntegrationBehavior	This field determines whether confirmation is required before sending cost item data to the external system. Accepted values are: • ControlWithoutConfirmation • ControlWithConfirmation	ControlWithoutConfirmation
IsAdvancedWorkAndSchedulingFeaturesEnabled	Flag that indicates if this feature is active in Project Settings.	1
IsTemplateProject	If this field is marked '1' then it enables a project to be listed as 'available' in the template project library	1
IsValidateEquipmentToProjectCurrency	Flag that indicates if this feature is active in Project Settings.	1
ProjectInformationTags	Use this to set Information Tags of the Project.	
ProjectInformationTag1	Project Information Tag 1.	Sample Tag 1
ProjectInformationTag2	Project Information Tag 2.	Sample Tag 2
ProjectInformationTag3	Project Information Tag 3.	Sample Tag 3
ProjectInformationTag4	Project Information Tag 4.	Sample Tag 4
ProjectInformationTag5	Project Information Tag 5.	Sample Tag 5
ProjectInformationTag6	Project Information Tag 6.	Sample Tag 6
ProjectInformationTag7	Project Information Tag 7.	Sample Tag 7
ProjectInformationTag8	Project Information Tag 8.	Sample Tag 8
ProjectInformationTag9	Project Information Tag 9.	Sample Tag 9
ProjectInformationTag10	Project Information Tag 10.	Sample Tag 10
ProjectInformationTag11	Project Information Tag 11.	Sample Tag 11
ProjectInformationTag12	Project Information Tag 12.	Sample Tag 12
ProjectInformationTag13	Project Information Tag 13.	Sample Tag 13
ProjectInformationTag14	Project Information Tag 14.	2020-10-06
ProjectInformationTag15	Project Information Tag 15.	2020-10-06
CustomLists	Parent field for Custom lists.	
ListName	Name of the list that is added to the custom label.	Estimating Resources
FieldValues	Comma delimited array of strings with a non-duplicate list of all field values that are part of the custom list.	Phase I, Phase II, Phase III, 0001, 0002
FuelTypes	Parent field for fuel type record.	
IsEstimatingResourceMapped	Flag that indicates if this feature is active in Project Settings.	TRUE
FuelTypeDisplay	Name of the type of fuel used.	Gasoline

Name	Description	Example
UOMSourceSystemId	Value provided in this field must match a valid unit of measure previously provided or defined in the InEight system.	a75d816a-456d-4e94-b4d6-2cfa4fc447de
CurrencyCode	Currency used to purchase the fuel. The value provided in this field must match a currency code in the InEight cloud platform and is selected for the project.	CAD
UnitCost	Value of the fuel per unit of measure.	USD
AccountCodeReference	Account code of the record that will be shown in normal use of product platform applications to assist users when performing searches or selections for the work completed.	AC001
IsActive	Flag that indicates if this feature is active in Project Settings.	TRUE
SourceSystemID	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not display in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	a75d816a-456d-4e94-b4d6-2cfa4fc447de
SourceSystemName	Name of the external system that is using the integration. Work with InEight to provision a unique value for this field.	SAP

Error Messages

The following error messages are generated by the InEight cloud platform and products for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity logic errors are performed internally in the InEight cloud platform and products to look for specific business rule or data integrity issues record-by-record. Failures with entity logic validations only cause the individual record to cease processing and are written to internal logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. Possible [Field Name] options: ProjectSourceSystemId	400	The request is invalid. The [Field Name] field is required.

API/Entity Logic	Condition	Code	Message
API Validation	Provided data exceeds the string size. Possible [Field Name] options: ProjectSourceSystemId ProjectSourceSystemName Fueltypedisplay UOMSourceSystemId	400	The request is invalid. The field [Field Name] must be a string with a maximum length of [Field Length].
Entity Logic Validation	Payload has invalid Project Currency.	200	Ignoring Project Employee {SourceSystemId} due to Employee currency did not match with project currencies
Entity Logic Validation	Payload has Invalid UOMName.	200	Ignoring ProjectEmployee Employee{SourceSystemId} due to invalid UOMName {UOMName}
Entity Logic Validation	Payload has conflict with uniqueness on SourceSystemId with existing data.	200	Ignoring ProjectEmployee having Duplicate Employee SourceSystemId '{SourceSystemId}'
Entity Logic Validation	All the rates are input project employee equal with original project employee and user vendor.	200	Ignoring ProjectEmployee {SourceSystemId} due to Unable to add/update the project employee with new rates if vendor assigned
Entity Logic Validation	Payload project employee UOM Id is not found.	200	Ignoring ProjectEmployee {SourceSystemId} due to Invalid UoM Id
Entity Logic Validation	Payload project employee is not active.	200	Ignoring inactive ProjectEmployee having SourceSystemId {SourceSystemId} for adding
Entity Logic Validation	Payload project employee is not active and project employee vendor id not equal to user vendor id.	200	Ignoring ProjectEmployee having Employee SourceSystemId '{SourceSystemId}' as delete is not allowed
Entity Logic Validation	Payload project employee is not mapped with related entity.	200	Ignoring invalid ProjectEmployee having SourceSystemId '{ SourceSystemId}' for update
Entity Logic Validation	Valid Payload	202	Imported {resultsCount} of {itemsCount} ProjectEmployee entities: Adds({addsCount}) Updates({updatesCount}) Deletes({deletesCount}) Errors(0) Skipped({itemsCount - resultsCount})

Sample JSON

```
[
   {
   "ProjectSourceSystemId": "a75d816a-456d-4e94-b4d6-2cfa4fc447de",
   "ProjectSourceSystemName": "SAP",
   "ProjectDisplayId": "103361",
   "GlobalOptions": {
      "CostItemIntegrationBehavior": "Controlwithoutconfirmation",
      "IsAdvancedWorkAndSchedulingFeaturesEnabled": true,
      "IsValidateEquipmentToProjectCurrency": true
      },
   "ProjectInformationTags": {
      "ProjectInformationTag1": "TAG1",
      "ProjectInformationTag2": "TAG2",
      "ProjectInformationTag3": "TAG3",
```

```
        "ProjectInformationTag4": "string",
        "ProjectInformationTag5": "string",
        "ProjectInformationTag6": "string",
        "ProjectInformationTag7": "string",
        "ProjectInformationTag8": "string",
        "ProjectInformationTag9": "string",
        "ProjectInformationTag10": "string",
        "ProjectInformationTag11": "string",
        "ProjectInformationTag12": "string",
        "ProjectInformationTag13": "string",
        "ProjectInformationTag14": "2021-01-22T19:44:16.861Z",
        "ProjectInformationTag15": "2021-01-22T19:44:16.861Z"
        },
    "CustomLists": [
        {
        "ListName": "Estimating Resources",
        "FieldValues": [
            "Phase I"
        ]
        }
    ],
    "FuelTypes": [
        {
        "IsEstimatingResourceMapped": true,
        "FuelTypeDisplay": "gasoline",
        "UOMSourceSystemId": "a75d816a-456d-4e94-b4d6-2cfa4fc447de",
        "CurrencyCode": "CAD",
        "UnitCost": 1.56,
        "AccountCodeReference": "A00001",
        "IsActive": true,
        "SourceSystemId": "a75d816a-456d-4e94-b4d6-2cfa4fc447de",
        "SourceSystemName": "SAP"
        }
    ]
    }
]
```

Verification

This integration allows a customer to import data from an external system for global options, fuel types, and custom lists that are shown in Edit Project > **Project Settings**. If a project is closed, the information is tagged in the setting values table and project information, details, and attributes cannot be updated using the integration.

The integration allows partial data in one section to be passed, independent of missing or invalid data in a field in a different section. For example, if fields in Global Settings have invalid data, the integration will allow Fuel Types or Custom Lists data to pass if the information in those sections are valid. The integration ignores the invalid data in Global Settings and does not fail the entire record because of invalid data in any one section of that project record. If all required fields are sent and with the correct format, other fields are ignored, and the record is posted.

IMPORTANT:	The entire setting details for a given project must be sent. Additionally, other parameters such as $select, $count, $top, $skip will not work in this integration.

Project Settings (Get)

Direction	From the InEight cloud platform to an external system.
Frequency	Determined by external system.
Trigger Methods	A user can request an export of any set of projects at any time
Average Payload Size	Entire entity – Hundreds or thousands of records. Entity delta – Tens or hundreds of records.
APIM Name	List ProjectSettings
Application Starting Version	20.11

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectSourceSystemId	String	200		Yes
1	ProjectSourceSystemName	String	50		Yes
1	ProjectDisplayId	String	200		No
1	GlobalOptions	Array			No
2	CostItemIntegrationBehavior	String	10	GlobalOptions	No
2	IsAdvancedWorkAndSchedulingFeaturesEnabled	Boolean	N/A	GlobalOptions	No
2	IsValidateEquipmentToProjectCurrency	Boolean	N/A	GlobalOptions	No
1	ProjectInformationTags	Array			No
2	ProjectInformationTag1	String	500	ProjectInformationTags	No
2	ProjectInformationTag2	String	250	ProjectInformationTags	No
2	ProjectInformationTag3	String	250	ProjectInformationTags	No
2	ProjectInformationTag4	String	250	ProjectInformationTags	No
2	ProjectInformationTag5	String	250	ProjectInformationTags	No
2	ProjectInformationTag6	String	100	ProjectInformationTags	No
2	ProjectInformationTag7	String	250	ProjectInformationTags	No
2	ProjectInformationTag8	String	250	ProjectInformationTags	No
2	ProjectInformationTag9	String	255	ProjectInformationTags	No
2	ProjectInformationTag10	String	250	ProjectInformationTags	No
2	ProjectInformationTag11	String	250	ProjectInformationTags	No
2	ProjectInformationTag12	String	250	ProjectInformationTags	No
2	ProjectInformationTag13	String	250	ProjectInformationTags	No

Depth	Name	Type	Precision[1]	Parent	Req.
2	ProjectInformationTag14	String	250	ProjectInformationTags	No
2	ProjectInformationTag15	String	250	ProjectInformationTags	No
1	CustomLists	Array			No
2	ListName	String	250	CustomLists	No
2	FieldValues	String	250	CustomLists	No
1	FuelTypes	Array			No
2	IsEstimatingResourceMapped	Boolean	N/A	FuelTypes	No
2	FuelTypeDisplay	String	50	FuelTypes	No
2	UOMSourceSystemId	String	200	FuelTypes	No
2	CurrencyCode	String	50	FuelTypes	No
2	UnitCost	Number	16,5	FuelTypes	No
2	AccountCodeReference	String	250	FuelTypes	No
2	IsActive	Boolean	N/A	FuelTypes	No
2	SourceSystemID	String	255	FuelTypes	No
2	SourceSystemName	String	50	FuelTypes	No

Field Descriptions

Name	Description	Example
ProjectSourceSystemId	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not display in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	a75d816a-456d-4e94-b4d6-2cfa4fc447de
ProjectSourceSystemName	Name of the external system that is using the integration. Work with InEight to provision a unique value for this field.	SAP
ProjectDisplayId	Unique identifier that is publicly recognizable. This field is displayed in normal use of InEight applications. The value sent in this field can match SourceSystemId if there is no need to use a public versus private unique identifier.	103361
GlobalOptions	Used to set Global Options of the project.	
CostItemIntegrationBehavior	This field determines whether confirmation is required before sending cost item data to the external system. Accepted values are: • ControlWithoutConfirmation • ControlWithConfirmation	ControlWithoutConfirmation

Name	Description	Example
IsAdvancedWorkAndSchedulingFeaturesEnabled	Flag that indicates if this feature is active in Project Settings.	TRUE
IsValidateEquipmentToProjectCurrency	Flag that indicates if this feature is active in Project Settings.	TRUE
ProjectInformationTags	Use this to set Information Tags of the Project.	
ProjectInformationTag1	Project Information Tag 1.	Sample Tag 1
ProjectInformationTag2	Project Information Tag 2.	Sample Tag 2
ProjectInformationTag3	Project Information Tag 3.	Sample Tag 3
ProjectInformationTag4	Project Information Tag 4.	Sample Tag 4
ProjectInformationTag5	Project Information Tag 5.	Sample Tag 5
ProjectInformationTag6	Project Information Tag 6.	Sample Tag 6
ProjectInformationTag7	Project Information Tag 7.	Sample Tag 7
ProjectInformationTag8	Project Information Tag 8.	Sample Tag 8
ProjectInformationTag9	Project Information Tag 9.	Sample Tag 9
ProjectInformationTag10	Project Information Tag 10.	Sample Tag 10
ProjectInformationTag11	Project Information Tag 11.	Sample Tag 11
ProjectInformationTag12	Project Information Tag 12.	Sample Tag 12
ProjectInformationTag13	Project Information Tag 13.	Sample Tag 13
ProjectInformationTag14	Project Information Tag 14.	2020-10-06
ProjectInformationTag15	Project Information Tag 15.	2020-10-06
CustomLists	Parent field for Custom lists.	
ListName	Name of the list that is added to the custom label.	Estimating Resources
FieldValues	Comma delimited array of strings with a non-duplicate list of all field values that are part of the custom list.	Phase I, Phase II, Phase III, 0001, 0002
FuelTypes	Parent field for fuel type record.	
IsEstimatingResourceMapped	Flag that indicates if this feature is active in Project Settings.	TRUE
FuelTypeDisplay	Name of the type of fuel used.	Gasoline
UOMSourceSystemId	Value provided in this field must match a valid unit of measure previously provided or defined in the InEight system.	a75d816a-456d-4e94-b4d6-2cfa4fc447de
CurrencyCode	Currency used to purchase the fuel. The value provided in this field must match a currency code in the InEight cloud platform and is selected for the project.	CAD
UnitCost	Value of the fuel per unit of measure.	USD

Name	Description	Example
AccountCodeReference	Account code of the record that will be shown in normal use of product platform applications to assist users when performing searches or selections for the work completed.	AC001
IsActive	Flag that indicates if this feature is active in Project Settings.	TRUE
SourceSystemID	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not display in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	a75d816a-456d-4e94-b4d6-2cfa4fc447de
SourceSystemName	Name of the external system that is using the integration. Work with InEight to provision a unique value for this field.	SAP

Error Messages

The following error messages are generated by the InEight cloud platform and products for this integration. Errors in the table below are distinguished by the process that checks for the error.

- API validation errors are basic record validations that will be returned to the API request message and cause the entire payload to fail.

- Entity logic errors are performed internally in the InEight cloud platform and products to look for specific business rule or data integrity issues record-by-record. Failures with entity logic validations only cause the individual record to cease processing and are written to internal logging.

API/Entity Logic	Condition	Code	Message
API Validation	Valid Payload	202	
API Validation	Required Fields are not Provided. Possible [Field Name] options: ProjectSourceSystemId	400	The request is invalid. The [Field Name] field is required.
API Validation	Provided data exceeds the string size. Possible [Field Name] options: ProjectSourceSystemId ProjectSourceSystemName Fueltypedisplay UOMSourceSystemId	400	The request is invalid. The field [Field Name] must be a string with a maximum length of [Field Length].
Entity Logic Validation	Payload has invalid Project Currency.	200	Ignoring Project Employee {SourceSystemId} due to Employee currency did not match with project currencies
Entity Logic Validation	Payload has Invalid UOMName.	200	Ignoring ProjectEmployee Employee{SourceSystemId} due to invalid UOMName {UOMName}

API/Entity Logic	Condition	Code	Message
Entity Logic Validation	Payload has conflict with uniqueness on SourceSystemId with existing data.	200	Ignoring ProjectEmployee having Duplicate Employee SourceSystemId '{SourceSystemId}'
Entity Logic Validation	All the rates are input project employee equal with original project employee and user vendor.	200	Ignoring ProjectEmployee {SourceSystemId} due to Unable to add/update the project employee with new rates if vendor assigned
Entity Logic Validation	Payload project employee UOM Id is not found.	200	Ignoring ProjectEmployee {SourceSystemId} due to Invalid UoM Id
Entity Logic Validation	Payload project employee is not active.	200	Ignoring inactive ProjectEmployee having SourceSystemId {SourceSystemId} for adding
Entity Logic Validation	Payload project employee is not active and project employee vendor id not equal to user vendor id.	200	Ignoring ProjectEmployee having Employee SourceSystemId '{SourceSystemId}' as delete is not allowed
Entity Logic Validation	Payload project employee is not mapped with related entity.	200	Ignoring invalid ProjectEmployee having SourceSystemId '{ SourceSystemId}' for update
Entity Logic Validation	Valid Payload	202	Imported {resultsCount} of {itemsCount} ProjectEmployee entities: Adds({addsCount}) Updates({updatesCount}) Deletes({deletesCount}) Errors(0) Skipped({itemsCount - resultsCount})

Sample JSON

```
[
   {
  "ProjectSourceSystemId": "a75d816a-456d-4e94-b4d6-2cfa4fc447de",
  "ProjectSourceSystemName": "SAP",
  "ProjectDisplayId": "103361",
  "GlobalOptions": {
     "CostItemIntegrationBehavior": "Controlwithoutconfirmation",
     "IsAdvancedWorkAndSchedulingFeaturesEnabled": true,
     "IsValidateEquipmentToProjectCurrency": true
     },
  "ProjectInformationTags": {
     "ProjectInformationTag1": "TAG1",
     "ProjectInformationTag2": "TAG2",
     "ProjectInformationTag3": "TAG3",
     "ProjectInformationTag4": "string",
     "ProjectInformationTag5": "string",
     "ProjectInformationTag6": "string",
     "ProjectInformationTag7": "string",
     "ProjectInformationTag8": "string",
     "ProjectInformationTag9": "string",
     "ProjectInformationTag10": "string",
     "ProjectInformationTag11": "string",
     "ProjectInformationTag12": "string",
     "ProjectInformationTag13": "string",
     "ProjectInformationTag14": "2021-01-22T19:44:16.861Z",
     "ProjectInformationTag15": "2021-01-22T19:44:16.861Z"
     },
   "CustomLists": [
```

```
          {
    "ListName": "Estimating Resources",
    "FieldValues": [
        "Phase I"
      ]
      }
    ],
    "FuelTypes": [
        {
    "IsEstimatingResourceMapped": true,
    "FuelTypeDisplay": "gasoline",
    "UOMSourceSystemId": "a75d816a-456d-4e94-b4d6-2cfa4fc447de",
    "CurrencyCode": "CAD",
    "UnitCost": 1.56,
    "AccountCodeReference": "A00001",
    "IsActive": true,
    "SourceSystemId": "a75d816a-456d-4e94-b4d6-2cfa4fc447de",
    "SourceSystemName": "SAP"
      }
    ]
    }
]
```

Verification

This integration allows a customer to export data to an external system for global options, fuel types, and custom lists that are shown in Edit Project > **Project Settings**.

The integration allows partial data in one section to be passed, independent of missing or invalid data in a field in a different section. For example, if fields in Global Settings have invalid data, the integration will allow Fuel Types or Custom Lists data to pass if the information in those sections are valid. The integration ignores the invalid data in Global Settings and does not fail the entire record because of invalid data in any one section of that project record. If all required fields are sent and with the correct format, other fields are ignored.

IMPORTANT:	The entire setting details for a given project must be sent. Additionally, other parameters such as $select, $count, $top, $skip will not work in this integration.

Employee Billing Classes

Project Employee Billing Classes (Import)

New billing classes and updates to billing classes performed using this API are automatically published instead of being staged and are immediately available for use in InEight cloud applications.

NOTE:	When using the import integration, any field that is not a required field can be excluded from the JSON. If a field is included in the JSON, the InEight data will be updated even if it the field is blank or contains a null value.

Direction	To the InEight cloud platform from an external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	Upsert ProjectEmployeeBillingClasses
Application Starting Version	21.3

Project Employee Billing Classes (Get)

Direction	From the InEight cloud platform to an external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List ProjectEmployeeBillingClasses
Application Starting Version	21.3

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectEmployeeBillingClass	String	100		Yes
1	SourceSystemID	String	100		Yes
1	SourceSystemName	String	50		No
1	ProjectID	String	200		Yes
1	Description	String	250		No

Depth	Name	Type	Precision[1]	Parent	Req.
1	VendorSourceSystemID	String	250		No
1	BillingRateST	Number	16,5		No
1	BillingRateDT	Number	16,5		No
1	BillingRateOT	Number	16,5		No
1	IsActive	Boolean	N/A		No

1 – For numeric data types, precision is given as total digits allowed in the field and the number of those digits that exist to the right of the decimal. For example, 16,5 represents a total of 16-digits allowed in the field with 5 of those digits existing as decimal places and 11 digits on the left of the decimal. The decimal is not counted as a digit.

Field Descriptions

Name	Description	Example
ProjectEmployeeBillingClass	Name of the employee billing class used on the project.	Carpenter
SourceSystemID	External identifier of the employee billing class record provided by the system of record for billing classes.	1035576511
SourceSystemName	Name of the system of record that is assigning Employee Billing class to projects.	SAP
ProjectID	External identifier of the project associated with the billing class.	105699973
Description	Optional field to add more detail about the billing class.	true
VendorSourceSystemID	Optional external identifier of the vendor assigned to the billing class.	644534242328
BillingRateST	Standard time rate charged to the owner for the employee.	45.00
BillingRateDT	Double-time rate charged to the owner for the employee.	90.00
BillingRateOT	Double time rate charged to the owner for the employee.	75.00
IsActive	This field is not recommended for use. Sending a value of false in this field will cause the record to be soft deleted from InEight and unrecoverable without assistance from InEight Professional Services.	true

Sample JSON

```
[
{
"ProjectEmployeeBillingClass": "Carpenter",
"SourceSystemId": "10456756",
"SourceSystemName": "SAP",
"ProjectId": "45442726",
"Description": "This field describes the different billing classes for journeyman
carpenters ",
"VendorSourceSystemId": "23242423732",
```

```
"BillingRateST": 45.00,
"BillingRateDT": 90.00,
"BillingRateOT": 75.00,
"IsActive": true
}
]
```

Verification

From the main menu, select a project, and then click **Billing classes**.

Equipment Billing Classes

Project Equipment Billing Classes (Import)

New equipment billing classes and updates to billing classes performed using this API are automatically published (instead of being staged) and are immediately available for use in InEight cloud applications.

NOTE:	When using the import integration, any field that is not a required field can be excluded from the JSON. If a field is included in the JSON, the InEight data will be updated even if it contains a blank or null value.

Direction	From external system to InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	Upsert ProjectEquipmentBillingClasses
Application Starting Version	21.3

Project Equipment Billing Classes (Get)

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List ProjectEquipmentBillingClasses
Application Starting Version	21.3

Fields

Depth	Name	Type	Precision	Parent	Req.
1	ProjectEquipmentBillingClass	String	100		Yes
1	SourceSystemID	String	100		Yes
1	SourceSystemName	String	50		Yes
1	ProjectID	String	200		Yes
1	Description	String	250		No

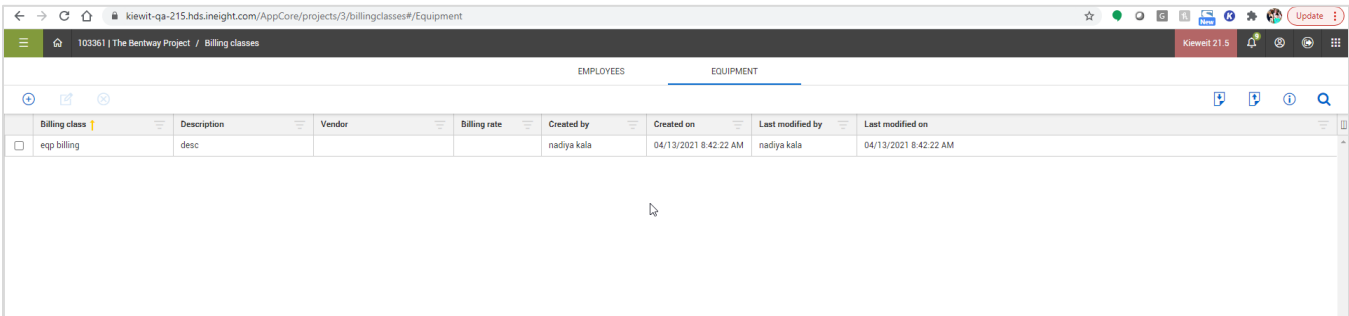| Depth	Name	Type	Precision	Parent	Req.
1	VendorSourceSystemID	String	250		Yes
1	IsActive	Boolean	N/A		No

Field Descriptions

Name	Description	Example
ProjectEquipmentBillingClass	Name of the equipment billing class used on the project	Carpenter
SourceSystemID	External identifier of the equipment billing class record provided by the system of record for billing classes.	1035576511
SourceSystemName	The name of the system of record that is assigning Equipment Billing class to projects.	SAP
ProjectID	The external identifier of the project associated with the billing class	105699973
Description	Optional field to add more detail about the billing class	True
VendorSourceSystemID	Optional external identifier of the vendor assigned to the billing class	644534242328
IsActive	This field is not recommended for use. Sending a value of false within this field will cause the record to be soft deleted from InEight and unrecoverable without assistance from InEight Professional Services	'1' or '0'

Sample JSON

```
[
    {
    "ProjectEquipmentBillingClass": "Frontloader",
    "SourceSystemId": "10456756",
    "SourceSystemName": "SAP",
    "ProjectId": "45442726",
    "Description": "This field describes the different billing classes for
excavators",
    "VendorSourceSystemId": "23242423732",
    "IsActive": true
    }
]
```

Verification

From the main menu, select a project, and then click **Billing classes**.

Project Currency

Direction	To the InEight cloud platform from an external system.
Frequency	Integration occurs each time a project is created in the customers external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – On initial customer implementation, potentially tens of records. Entire entity – As projects are added, potentially several per month.
APIM Name	Upsert ProjectCurrency
Application Starting Version	22.6

Direction	From the InEight cloud platform to an external system.
Frequency	Determined by external system.
Trigger Methods	Triggered manually by user or is triggered through automated process.
Average Payload Size	Entire entity – 1-1000 records. Entity delta – 1 -100 records.
APIM Name	List ProjectCurrency
Application Starting Version	22.6

NOTE:	The fields, field descriptions, and JSON for the ProjectCurrency_Import and ProjectCurrency_Get integrations are identical, however, the purpose of these integrations and how they are used with the InEight cloud platform differ. Refer to the InEight APIM for details.

Fields

Depth	Name	Type	Precision	Parent	Req.
1	ProjectID	String	200		Yes
1	CurrencyCode	String	50		Yes
1	ExchangeRateToBase	Number	16,9		No
1	IsBase	Boolean	N/A		No
1	IsActive	Boolean	N/A		No

1 – For numeric data types, precision is given as total digits allowed in the field and the number of those digits that exist to the right of the decimal. For example, 16,5 represents a total of 16-digits allowed in the field with 5 of those digits existing as decimal places and 11 digits on the left of the decimal. The decimal is not counted as a digit.

Field Descriptions

Name	Description	Example
ProjectID	Identifier created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not display in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	103361
CurrencyCode	Unique Currency identifier.	USD, CAD, AUD
ExchangeRateToBase	Conversion factor to base currency.	1.5, 3, 5
IsBase	Flag that indicates if the currency is the base currency or alternate currency.	true
IsActive	Sending a value of false in this field causes the record to be soft deleted from the InEight cloud platform. If a value is not provided the default value true is used.	true

Sample JSON

```
[{
"ProjectId": 34898,
"CurrencyCode": "USD",
"ExchangeRateToBase": "",
"IsBase": true,
"IsActive": true,
}]
```

Rate Codes

Direction	To the InEight cloud platform from an external system.
Frequency	Integration occurs each time a project is created in the customers external system.
Trigger Methods	Determined by external system.
Average Payload Size	Entire entity – On initial customer implementation, potentially tens of records. Entire entity – As projects are added, potentially several per month.
APIM Name	Upsert ProjectRateCodes
Application Starting Version	22.10

Fields

Depth	Name	Type	Precision[1]	Parent	Req.
1	ProjectSourceSystemId	String	200		Yes
1	ProjectRateCode	String	100		Yes
1	Description	String	255		No
1	UoMName	String	255		Yes
1	VendorCode	String	250		No
1	CurrencyCode	String	50		Yes
1	UnionSourceSystemId	String	50		No
1	StraightTimeRate	Number	Double		No
1	OverTimeFactor	Number	Double		No
1	OverTimeRate	Number	Double		No
1	DoubleTimeFactor	Number	Double		No
1	DoubleTimeRate	Number	Double		No
1	SourceSystemId	String	100		Yes
1	SourceSystemName	String	100		No
1	UseBaseWageFactors	Boolean	N/A		No
1	IsAvailable	Boolean	N/A		No

1 - For numeric data types, precision is given as total digits allowed in the field and the number of those digits that exist to the right of the decimal. For example, 16,5 represents a total of 16-digits allowed in the field with 5 of those digits existing as decimal places and 11 digits on the left of the decimal. The decimal is not counted as a digit.

Field Descriptions

Name	Description	Example
ProjectSourceSystemId	Source system ID of the project that the employees are assigned to. This field is shown as External Project ID in the UI.	123456
ProjectRateCode	Unique identifier of the rate code record provided by the system or record.	123456
Description	Description of the rate code.	ProjectRateCodeDescription
UoMName	Public name of the unit of measure assigned to the rate code.	Acre
VendorCode	InEight ID of the vendor assigned to the rate code.	0010046231
CurrencyCode	ISO code of the currency assigned to the rate code.	USD
UnionSourceSystemId	SourceSystemId of a union assigned to the rate code. Must match a valid union from Master Data in InEight.	U001
StraightTimeRate	Rate for standard pay that should be applied for this rate code.	10.00
OverTimeFactor	Factor used to multiply the standard rate of the employee to determine their overtime rate. Is blank if not using Factors.	2
OverTimeRate	Exact calculated rate paid to the employee for overtime work as per applicable laws. Is blank if using factors.	20.00
DoubleTimeFactor	Factor used to multiply the standard rate of the employee to determine their double time rate. Is blank if not using factors.	3
DoubleTimeRate	Exact calculated rate paid to the employee for double time work as per applicable laws. Is blank if using factors.	30.00
SourceSystemId	Unique identifier created by the system of record and used in all system communications as the primary method of specifying a unique record.	123456
SourceSystemName	Customer provided name of the system of record for the rate code.	ProjectRateCodeSSN
UseBaseWageFactors	Flag that determines whether the record uses absolute values or multiplication factors for pay rates.	true
IsAvailable	Determines the status of the rate code and whether it can be assigned to the project.	true

Sample JSON

```
[
    {
    "ProjectSourceSystemId": "2570802",
    "ProjectRateCode": "55622b",
    "Description": "Exec365",
    "UoMName": "Acre",
    "VendorCode": "6698755",
    "CurrencyCode": "USD",
    "UnionSourceSystemId": "U6345",
    "StraightTimeRate": 25.00,
    "OverTimeFactor": 2.0,
    "OverTimeRate": 50.00,
    "DoubleTimeFactor": 3.0,
    "DoubleTimeRate": 75.00,
    "SourceSystemId": "092407",
    "SourceSystemName": "RateCodeSSN",
    "UseBaseWageFactors": true,
    "IsAvailable": true
    }
]
```

Commodities

Direction	From external system to InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	Upsert Commodities
Application Starting Version	22.10

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List Commodities
Application Starting Version	22.10

Fields

Depth	Name	Type	Precision	Parent	Req.
1	Name	String	100		Yes
1	Description	String	100		No
1	SourceSystemId	String	100		Yes
1	SourceSystemName	String	50		No
1	IsSystem	Boolean	N/A		No
1	IsAvailable	Boolean	N/A		No

Field Descriptions

Name	Description	Example
Name	Name of the commodity.	Lubricant
Description	Description of the commodity.	Silicone dielectric grease

Name	Description	Example
SourceSystemId	Unique identifier of the commodity created by the system of record and used in all system communications as the primary method of specifying a unique record.	COMM0012
SourceSystemName	Customer provided name of the system of record for the commodity.	Commodity_SSN
IsSystem	Identifies whether the commodity is system generated.	false
IsAvailable	Determines the status of the commodity and whether it can be assigned to the project.	true

JSON Sample

```
[
    {
    "Name": "string",
    "Description": "string",
    "SourceSystemId": "COMM0012",
    "SourceSystemName": "Commodity_09705",
    "IsSystem": false,
    "IsAvailable": true,
    }
]
```

Disciplines

Direction	From external system to InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	Upsert Disciplines
Application Starting Version	22.10

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List Disciplines
Application Starting Version	22.10

Fields

Depth	Name	Type	Precision	Parent	Req.
1	Name	String	100		Yes
1	Description	String	100		No
1	SourceSystemId	String	100		Yes
1	SourceSystemName	String	50		No
1	IsSystem	Boolean	N/A		No
1	IsAvailable	Boolean	N/A		No

Field Descriptions

Name	Description	Example
Name	Name of the discipline.	Electrical
Description	Description of the discipline.	TC Discipline

Name	Description	Example
SourceSystemId	Unique identifier of the discipline created by the system of record and used in all system communications as the primary method of specifying a unique record.	TC_Discipline_123456
SourceSystemName	Customer provided name of the system of record for the discipline.	Discipline_SSN
IsSystem	Identifies whether the discipline is system generated.	false
IsAvailable	Determines the status of the discipline and whether it can be assigned to the project.	true

JSON Sample

```
[
    {
    "Name": "Structural",
    "Description": "TC_Discipline_140",
    "SourceSystemId": "Discipline_08",
    "SourceSystemName": "Discipline_SSN",
    "IsSystem": false,
    "IsAvailable": true,
    }
]
```

INEIGHT

Project Values

Direction	From external system to InEight cloud platform.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	Upsert ProjectValues
Application Starting Version	23.4

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List ProjectValues
Application Starting Version	23.4

Fields

Depth	Name	Type	Precision	Parent	Req.
1	ProjectValueTypeSourceSystemId	String	100		Yes
1	ID	String	50		Yes
1	Name	String	200		Yes
1	Description	String	250		No
1	ProjectSourceSystemId	String	200		Yes
1	SourceSystemId	String	100		Yes
1	SourceSystemName	String	50		No
1	IsAvailable	Boolean	N/A		No

Field Descriptions

Name	Description	Example
ProjectValueTypeSourceSystemId	Type of the custom group, such as Area, Segment, etc.	System

Name	Description	Example
ID	Abbreviated description for the given custom group element.	
Name	Name of the custom group of the given type.	
Description	Description for the custom group element.	
ProjectSourceSystemId	Unique Identifier for each project created by the system of record and used in all system communications as the primary method of specifying a unique record. This field does not show in normal use of InEight applications and can contain internal identifiers such as a GUID that is not understandable by typical system users.	25384702
SourceSystemId	Unique identifier of the project value created by the system of record and used in all system communications as the primary method of specifying a unique record.	
SourceSystemName	Customer provided name of the system of record for the custom group.	
IsAvailable	Status that determines whether the project value can be assigned to the project.	true

JSON Sample

```
[
    {
    "ProjectValueTypeSourceSystemId": "System",
    "ID": "",
    "Name": "",
    "Description": "",
    "ProjectSourceSystemId": "",
    "SourceSystemId": "",
    "SourceSystemName": "",
    "IsAvailable": true
    }
]
```

INEIGHT

Project Allowances

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List ProjectPayrollIndicatorsAllowances
Application Starting Version	24.11

Fields

Depth	Name	Type	Precision	Parent	Req.
1	PayrollIndicatorDisplayId	String	50		No
1	PayrollIndicatorSourceSystemId	String	100		Yes
1	ProjectSourceSystemId	String	200		Yes
1	BillingCode	String	10		No
1	IsAvailable	Boolean	N/A		No
1	IsAutoOptIn	Boolean	N/A		No
1	AllowanceRate	Decimal	16,5		No

Field Descriptions

Name	Description	Example
PayrollIndicatorDisplayId	DisplayId of the Payroll Indicator used within master data value.	
PayrollIndicatorSourceSystemId	External System Id of the Payroll Indicator used within master data value.	
ProjectSourceSystemId	SourcesystemId of the project where the payroll indicator is being assigned. This is the same as the SourcesystemId field in the List Projects API.	
BillingCode	Billing code associated with the allowance.	
IsAvailable	Determines the status and whether the payroll indicator can be assigned to the project.	True
IsAutoOptIn	Indicates if this allowance is automatically assigned to all new project employees.	True
AllowanceRate	Currency value for allowance.	

JSON Sample

```
[
    {
    "PayrollIndicatorDisplayId": "string",
    "PayrollIndicatorSourceSystemId": "string",
    "ProjectSourceSystemId": "string",
    "BillingCode": "string",
    "IsAvailable": true,
    "IsAutoOptIn": false,
    "AllowanceRate": 0
    }
]
```

Project Attendance Types

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List ProjectPayrollIndicatorsAttendanceTypes
Application Starting Version	24.11

Fields

Depth	Name	Type	Precision	Parent	Req.
1	PayrollIndicatorDisplayId	String	50		No
1	PayrollIndicatorSourceSystemId	String	100		Yes
1	ProjectSourceSystemId	String	200		Yes
1	BillingCode	String	10		No
1	IsAvailable	Boolean	N/A		No

Field Descriptions

Name	Description	Example
PayrollIndicatorDisplayId	DisplayId of the attendance type used within master data value.	
PayrollIndicatorSourceSystemId	External System Id of the attendance type used within master data value.	
ProjectSourceSystemId	SourcesystemId of the project where the attendance type is being assigned. This is the same as the SourcesystemId field in the List Projects API.	
BillingCode	Billing code associated with the attendance type.	
IsAvailable	Determines the status and whether the attendance type can be assigned to the project.	True

JSON Sample

```
[
    {
    "PayrollIndicatorDisplayId": "string",
    "PayrollIndicatorSourceSystemId": "string",
    "ProjectSourceSystemId": "string",
    "BillingCode": "string",
    "IsAvailable": true,
    }
]
```

Project Employee Premiums

Direction	From InEight cloud platform to external system.
Frequency	Determined by external system.
Trigger Methods	Determined by external system.
Average Payload Size	Determined by external system.
APIM Name	List ProjectPayrollIndicatorsEmployeePremiums
Application Starting Version	24.11

Fields

Depth	Name	Type	Precision	Parent	Req.
1	PayrollIndicatorDisplayId	String	50		No
1	PayrollIndicatorSourceSystemId	String	100		Yes
1	ProjectSourceSystemId	String	200		Yes
1	BillingCode	String	10		No
1	IsAvailable	Boolean	N/A		No

Field Descriptions

Name	Description	Example
PayrollIndicatorDisplayId	DisplayId of the employee premium used within master data value.	
PayrollIndicatorSourceSystemId	External System Id of the employee premium used within master data value.	
ProjectSourceSystemId	Sourcesystemid of the project where the employee premium is being assigned. This is the same as the Sourcesystemid field in the List Projects API.	
BillingCode	Billing code associated with the employee premium.	
IsAvailable	Determines the status and whether the employee premium can be assigned to the project.	True

JSON Sample

```
[
    {
    "PayrollIndicatorDisplayId": "string",
    "PayrollIndicatorSourceSystemId": "string",
    "ProjectSourceSystemId": "string",
    "BillingCode": "string",
    "IsAvailable": true,
    }
]
```