



Guide

Payroll Data Input API Guide for ADP Enterprise HR

Published on
May 01, 2020 4:57PM

Last modified
Feb 16, 2022 3:49PM





ADP Copyright Information

ADP, the ADP logo, and Always Designing for People are trademarks of ADP, Inc.

Windows is a registered trademark of the Microsoft Corporation.

All other trademarks are the property of their respective owners.

Copyright © 2022 ADP, Inc. ADP Proprietary and Confidential - All Rights Reserved. These materials may not be reproduced in any format without the express written permission of ADP, Inc.

These materials may not be reproduced in any format without the express written permission of ADP, Inc. ADP provides this publication "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. ADP is not responsible for any technical inaccuracies or typographical errors which may be contained in this publication. Changes are periodically made to the information herein, and such changes will be incorporated in new editions of this publication. ADP may make improvements and/or changes in the product and/or the programmes described in this publication.

Published on
May 01, 2020 4:57PM

Last modified
Feb 16, 2022 3:49PM

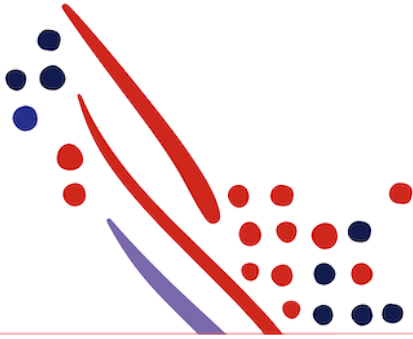


Table of Contents

Chapter 1

About this API

- Summary
- What's New in this Guide?
- Supported Product Version and Customer Base
- Required Setup Steps
- Postman Collection
- Process Overview

Chapter 2

Use Case: Adding a Pay Data Batch

- Use Case Description
- API Usage
- Application Scope
- Supported Actors
- Request Header Parameters
- Sequence of Interactions
- Data Dictionary
- Responses

Chapter 3

Known Issues and Limitations

- Issue 1: <User Story or Defect title goes here>
 - Issue Description
 - Impacted APIs
 - Suggested Workaround

About this API

Summary

ADP Enterprise HR enables users to create a Pay Data batch for a payroll cycle to include the following:

- Hours and Earnings
- Deductions
- Memos
- Taxes
- Temporary Cost Numbers
- Temporary Department Numbers
- Allocations

The Payroll Data Input Application Programming Interface (API) enables you to build a data connector application for clients to transmit data sourced in your application to a specific Pay Data batch in ADP Enterprise HR.

What's New in this Guide?

The content in this guide was just moved from the ADP Enterprise Data Dictionary and expanded upon. In this section, revisions to the Payroll Data Input API and this guide will be announced.

Supported Product Version and Customer Base

More information to come

Required Setup Steps

More information to come

Postman Collection

Postman allows you to import a collection of APIs, created by others, so you can try them out. For more information on Postman, see [Making Your First API Call Using Postman](#).

To download API collections for the Payroll Data Input from the ADP GitHub library and import them to Postman, go to <Postman Collection Link>.

Process Overview

The following table illustrates the key data flow steps.

	Actor	Task Description
1		

Chapter 2

Use Case: Adding a Pay Data Batch

Use Case Description

More information to come

API Usage

Method	Uniform Resource Identifier (URI)	Description	GitHub Sample Request Payload	GitHub Sample Response Payload
POST	/events/payroll/v1/pay-data-input.add		<pre>{ "events": [{ "data": { "eventContext": { "payrollWeekNumber": "", "payrollRunNumber": "", "payrollProcessingJobID": "SID220", "payrollGroupCode": { "codeValue": "ET1" } }, "transform": { "payDataInput": { "payeePayInputs": [{ "associateOID": "G3VVEK1Y1XMFWNM", "payNumber": "1", "payrollProfilePayInputs": [{ "payrollFileNumber": "003001", "payInputs": [{ "payAllocation": { "allocationID": "1", "allocationTypeCode": { "codeValue": "A1Num" } }, "earningInputs": [{ "configurationTags": [{ "tagCode": "Shift Code", "tagValues": ["1"] }] }] }] }] }] } } } }] }</pre>	<pre>{ "events": [{ "data": { "eventContext": { "payrollGroupCode": { "codeValue": "ET1" }, "payrollWeekNumber": "", "payrollRunNumber": "", "payrollProcessingJobID": "SID220" }, "transform": { "payDataInput": { "payeePayInputs": [{ "associateOID": "G3VVEK1Y1XMFWNM", "payNumber": "1", "payrollProfilePayInputs": [{ "payrollFileNumber": "003001", "payInputs": [{ "payAllocation": { "allocationID": "1", "allocationTypeCode": { "codeValue": "A1Num" } }, "earningInputs": [{ "modifierCode": { "codeValue": "2" }, "rateCode": { "codeValue": "" }, "rate": { "rateValue": 600 }, "numberOfHours": 60, "configurationTags": [{ "tagCode": "Shift Code", "tagValues": ["1"] }] }] }] }] }] } } } }] }</pre>

			<pre> }, "rateCode": { "codeValue": "" }, "modifierCode": { "codeValue": "2" }, "rate": { "rateValue": 600 }, "numberOfHours": 60 }] }] }] }] } } }] } </pre>	
--	--	--	---	--

Application Scope

The canonical URI corresponding to the Applicant Onboard API needs to be added in the Consumer Application Registry (CAR) for the subscription following which a user can access this API and make successful API calls.

The following canonical needs to be added to your application scope to enable this use case:

<Canonical goes here>

Supported Actors

ADP APIs have built-in data entitlement and access control. The response contains data based on the **roleCode** parameter value in an API request. For example, a person could have two roles defined in an ADP application, a practitioner who manages all employees, and an employee. Your application will use the practitioner **roleCode** to request the data as the practitioner and the **roleCode** of the employee.

Request Parameter roleCode Value	Is Supported?	Usage when Supported
practitioner		
employee		
manager		
admin		

Request Header Parameters

More information to come

Sequence of Interactions

The following are the steps shown in the previous diagram:

1. Your consumer application makes a request to the ...
2. Next step.
3. Next step.
4. Your consumer application makes a request to the ...
5. Next step.

Data Dictionary

The client practitioner can manage data controls in the ADP Workforce Now user interface (UI) by selecting <menu path goes here>.

Schema Location	Field Name	Is it Required in the Request Payload?	Note
/events/data/eventContext/payrollGroupCode/codeValue	Payroll Group/Company Code	YES	
/events/data/eventContext/payrollWeekNumber	Work Week	NO	
/events/data/eventContext/payrollProcessingJobID	Batch Id	YES	
/events/data/transform/payDataInput/payeePayInputs/associateOID	Employee Associate OID	YES	
/events/data/transform/payDataInput/payeePayInputs/payNumber	Pay Number	YES	
/events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payrollFileNumber	Payroll File Number	YES	
/events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocationTypeCode	Type of Allocation	NO	- ALNUM for 'Allocation number' - CNUM for 'Cost Number' - DEPTID for 'Department Number'

/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/payAllocatio n/allocationID	Depending on the allocationType value: ALA # Or Temp Dept Or Temp Cost Nbr	NO	
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput s/modifierCode	Modifier Code Value	NO	This is the type of earning code. Value are: <ul style="list-style-type: none"> • 1 - Regular (Hours & Earnings) • 2 - OT (Hours & Earnings) • 3 - HOURS_3 • 4 - HOURS_4 • 7 - EARNINGS_3 • 8 - EARNINGS_4 • 9 - EARNINGS_5 • 24 - Temporary Hourly Rate
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput s/earningCode/codeValu e	Earning/Hour Code	NO	This depends on the Modifier Code Value.
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput s/rateCode/codeValue	Rate Code	NO	
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput s/rate/rateValue	Rate Value	NO	
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput s/numberOfHours	Number of Hours	NO	
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput s/configurationTags /tagCode	Shift Indicator	NO	This defaults to Shift Code .
/events/data/transform/ payDataInput/payeePayl nputs /payrollProfilePayInputs /payInputs/earningInput	Shift Value	NO	

s/configurationTags/tagValues			
/events/data/transform/payDataInput/payeePayInputs /payrollProfilePayInputs/payInputs/memoInputs/memoCode/codeValue	Memo Code	NO	
/events/data/transform/payDataInput/payeePayInputs /payrollProfilePayInputs/payInputs/memoInputs/memoAmount/amountValue	Memo Amount	NO	
/events/data/transform/payDataInput/payeePayInputs /payrollProfilePayInputs/payInputs/deductionInputs/configurationTags/tagCode /events/data/transform/payDataInput/payeePayInputs /payrollProfilePayInputs/payInputs/deductionInputs /configurationTags/tagValues	Deduction Type	No	Values are: <ul style="list-style-type: none"> • Temporary for Adjust Deductions • Special for Replacement Deductions
/events/data/transform/payDataInput/payeePayInputs /payrollProfilePayInputs/payInputs/deductionInputs/deductionCode/codeValue	Adjust Ded Cod or Repl Ded Code	NO	
/events/data/transform/payDataInput/payeePayInputs /payrollProfilePayInputs/payInputs/deductionInputs/deductionRate/rateValue	Adjust Ded Amount or Repl Ded Amount	NO	

Responses

You may encounter exceptions outside your common success scenarios. You must account for these exceptions during your initial development.

For more information, see [API Common Exceptions and Tips for Handling](#).

Response Code	Condition	messageTxt	Sample Request	Sample Response	Tips to handle

--	--	--	--	--	--

Chapter 3

Known Issues and Limitations

Issue 1: <User Story or Defect title goes here>

Issue Description

More information to come

Impacted APIs

Method	URI	roleCode

Suggested Workaround

More information to come