



## Chapter 2

# Use Case: Retrieving Payroll Output Details

from Payroll Output API Guide for RUN Powered by ADP

Published on  
Mar 11, 2020, 06:23 PM

Last modified  
Sep 06, 2023, 08:02 AM





## 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.

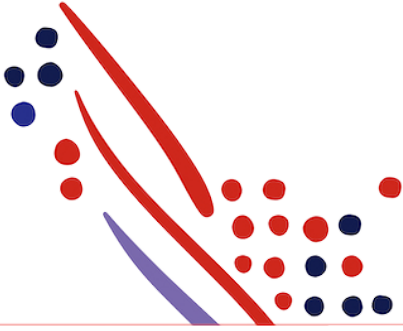
All other trademarks are the property of their respective owners.

Copyright © 2023 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  
Mar 11, 2020, 06:23 PM

Published on  
Sep 06, 2023, 08:02 AM



# Chapter Contents

## Chapter 2

### Payroll Output API Guide for RUN Powered by ADP

- API Usage

- Responses

- Application Scope

- Request Header Parameters

- Supported OData Query Options

- Sequence of Interactions

- Data Dictionary

- About Configuration Tags

  - Regular Payroll (CR), Special Payroll (CS), and Adjustments (AC)

  - Void Checks (V), Manual Checks (M1), and Third Party Sick Pay (T)

  - Identifying Payroll Types from AssociatePayroll Instructions in the API Output Response

- Responses

# Use Case: Retrieving Payroll Output Details

## API Usage

For API related information please refer [Payroll Outputs](#) under API Explorer.

Method	Uniform Resource Identifier (URI)	Description
GET	<a href="#">/payroll/v1/payroll-output</a>	Returns a list of all available payroll outputs the requester is authorized to view.
GET	<a href="#">/payroll/v1/payroll-output/{payruncode}</a>	Returns the payroll output of a <b>payruncode</b> , which includes active and terminated employees' payment information by the <b>payrun</b> period. <b>NOTE:</b> The Payroll Output API has the list of associates information who are active for a particular payroll run. But, it's not mandatory all associates from employees should be displayed in the Payroll Output API. The number of active associates for a payroll run are only seen in the output. We can compare the Payroll Summary Report in RUN Powered by ADP to see how many associates are active for a payroll run with the Payroll Output API response.
GET	<a href="#">/payroll/v1/payroll-output/meta</a>	Requests meta information.

The [/payroll/v1/payroll-output](#) API lists the details of the **payruncodes**. To see the full details of the payroll output information for a **payruncode**, the user needs to make a call based on the **{payruncode}** as follows: [/payroll/v1/payroll-output/{payruncode}](#)

## Responses

For API related information please refer [Payroll Outputs](#) under API Explorer.

Method	URI	Description	Response Code
GET	<a href="#">/payroll/v1/payroll-output/{payruncode}</a>	When a Regular Payroll is run.	200 OK

Method	URI	Description	Response Code
GET	<a href="#">/payroll/v1/payroll-output/finance/payruncode</a>	When a client practitioner voids a check for an associate and provides a Manual Check to an employee in the client's RUN Powered by ADP payroll system. Later, the client practitioner runs a Regular Payroll. The Void Check and Manual Check are included in the Regular Payroll commit.	200 OK
GET	<a href="#">/payroll/v1/payroll-output/finance/payruncode</a>	When a Special Payroll is run.	200 OK
GET	<a href="#">/payroll/v1/payroll-output/finance/payruncode</a>	When a client practitioner voids an associate's check (to make corrections) and runs a Special Payroll to pay the associate again. The Void Check gets included in the Special Payroll. The associate <b>Payment Instruction</b> section of the API Output Response indicates the voided check and Special Payroll values.	200 OK
GET	<a href="#">/payroll/v1/payroll-output/finance/payruncode</a>	When a pending Void Check, Special Payroll, and Third Party Sick Pay are included in a Special Payroll.	200 OK
GET	<a href="#">/payroll/v1/payroll-output/finance/payruncode</a>	When an Adjustment is run. <b>NOTE:</b> An Adjustment doesn't result in any checks and no one gets paid. It's used to make year-end corrections for W2's or to correct balances when a new client is onboarded onto RUN Powered by ADP, and so on.	200 OK

## Application Scope

The canonical URI corresponding to the 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:

`payroll/payrollManagement/payrollProcessingResults/payrollOutputManagement/  
payrollOutput.read`

## Request Header Parameters

There are no request header parameters, in addition to the ADP standard header parameters.

## Supported OData Query Options

This use case supports Query String Options as defined by Open Data Protocol (OData). For more information about OData, see the [Introduction to ADP API Open Data Protocol \(OData\)](#).

For API related information please refer [Payroll Outputs](#) under API Explorer.

Query String Option	Description	Example
\$filter	Filters a collection of resources addressed by a request Uniform Resource Locator (URL). Only items for which the expression evaluates to true are included in the response.	/payroll/v1/payroll-output? \$filter=payrollProcessingJobID eq '99785665-201501090001'

## Sequence of Interactions

The following are the steps shown in the previous diagram:

1. Your consumer application makes a request to the ADP API endpoint to GET Payroll Output Details based on **payruncode**.
2. The ADP API endpoint responds to your consumer application with payroll output information.

## Data Dictionary

All fields listed in the **Data Dictionary** chapter are supported for this use case.

## About Configuration Tags

Tag	Description
CR	Regular Pay Calendar Entry
CS	Special Pay Calendar Entry
AC	Adjustment Calculation
T	Third Party Sick Pay
V	Void Check
M1	Manual Check

## Regular Payroll (CR), Special Payroll (CS), and Adjustments (AC)

RUN Powered by ADP has two types of payroll commits: **Regular Payroll** and **Special Payroll**. Payouts resulting out of these two payroll commits are indicated by a CR (regular) and CS (special).

**Adjustments** are commits where a client needs to have the ability to adjust any piece of payroll information for an associate without producing a check.

## Void Checks (V), Manual Checks (M1), and Third Party Sick Pay (T)

Whenever a Client Practitioner in the RUN Powered by ADP portal voids one or more associate's paycheck, the Void Checks get included in the next Regular Payroll or Special Payroll run. As a result, the Payroll Output API, under the **associatePayments Instructions**, will also include the Void Check instructions for all the associates whose checks were made void by the practitioner. The Configuration Tags are used to identify those voids.

Similarly, Manual Checks (M1) and TPSP get included in the next Special Payroll or Regular Payroll run and are identified by their respective configuration tags in the API response.

## Identifying Payroll Types from AssociatePayroll Instructions in the API Output Response

For each associate, **associatePayroll Instruction** in the Payroll Output API response can have all of the configuration-tags: T, V and M1, including either CS or CR, but not both (to indicate whether it was a Special Payroll or a Regular Payroll, respectively).

For an Adjustment run, AC is the only tag present. This helps in differentiating a payroll output resulting out of an Adjustment.

## 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](#).

For API related information please refer [Payroll Outputs](#) under API Explorer.

Response Code	Condition	messageTxt	GitHub Sample Request Payload
200 OK	Retrieve the payroll output details.	NA	NA
200 OK	Retrieve the payroll output details based on the <b>payruncode</b> .	NA	NA
200 OK	Retrieve the payroll output details with an invalid <b>payruncode</b> .	{"payrollOutputs": []}	NA
200 OK	Incorrect query parameters and value when using filter in the request. For example: <code>/payroll/v1/payroll-output?filter=payrollProcessingJobID eq 'G3P3QYIJXG79N3EB-20190302'</code>	{"payrollOutputs": []}	NA