



Chapter 5

Using the General Deduction Instruction Stop API

from Deduction Instruction Guide for ADP Workforce
Now Next Gen Guide

Published on
May 13, 2021 5:20AM

Last modified
Feb 16, 2022 5:20PM



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.

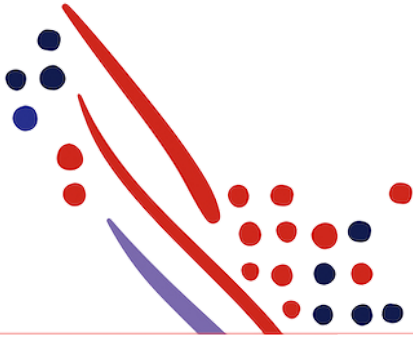
All other trademarks are the property of their respective owners.

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 13, 2021 5:20AM

Last modified
Feb 16, 2022 5:20PM



Chapter Contents

Chapter 5

Using the General Deduction Instruction Stop API

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

Using the General Deduction Instruction Stop API

Use Case Description

This use case is used to stop a general deduction instruction for a single worker. A successful stop request changes the Deduction record from the **Active** to the **Inactive** state.



Note

Deduction STOP api will Inactive the specific deduction code, instead of complete deduction policy

To view **Inactive** deductions in the ADP Workforce Now UI, select **People > Pay > Pay Profile** and follow these steps:

1. Find the worker record.
2. Click **Deductions**.
3. Select data from **Show as of**.
4. Switch **SHOW INACTIVE** to **YES**.

Your program could also retrieve **Inactive** records using APIs. For more information, see [Chapter 2 - Use Case: Retrieving Payroll Instructions \(Payroll Instructions API\)](#).

API Usage

Method	URI	Description	GitHub Sample Request Payload	GitHub Sample Response Payload
GET	/events/payroll/v2/worker-general-deduction-instruction.stop/meta	Returns an event metadata. Note: Sample payload can be built from the meta call.	NA	deduction-stop-meta-200-response.json
POST	/events/payroll/v2/worker-general-deduction-instruction.stop	Stops worker general deduction instruction information.	deduction-stop-200.request.json	deduction-stop-200.response.json



Important

In the given payload ([deduction-stop-200.request.json](#)), the following is the purpose of the **eventContext** and **transform** sections:

- **eventContext**: A set of keys, identifying the subject. In the payload, the **associateID** field is present under **eventContext**. The **associateID** identifies the subject.
- **transform**: Provides the values added or changed with respect to the subject keys defined in the **eventContext** section.

Application Scope

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

/payroll/payrollManagement/payrollInstructionManagement/workerGeneralDeductionInstructionManagement/workerGeneralDeductionInstructionManagement

Supported Actors

Request Parameter roleCode Value	Usage
practitioner	Stops deduction instructions for workers.

Request Header Parameters

None, in addition to the ADP standard header parameters.

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 for a **Worker-general-deduction-instruction Stop** meta.
2. The ADP API endpoint responds to your consumer application with the meta payload, which lists the eligible deductions for the worker under the **/meta/data/transforms/payrollInstruction/generalDeductionInstruction/deductionCode/codeList** collection. Your consumer application needs to validate the Deduction Code to be processed is returned in the **codeList** collection before building the payload for the next call. If the Deduction Code you need to start is not part of the collection, your application needs to provide user error handling and stop or go to the next step.
3. Your consumer application makes a request to the ADP API endpoint for the **Worker-general-deduction-instruction Stop** API with the payload.
4. The ADP API endpoint responds to the consumer application about the details of the **Worker-general-deduction-instruction Stop**.

Data Dictionary

The fields listed in the following table can be found in the ADP Workforce Now UI by selecting **People > Pay > Deductions**.

Schema Location	ADP Workforce Now Field Name	Required (Y/N)?	Note
/events/data/eventContext	Not Displayed	N	
/events/data/eventContext/worker	Not Displayed	N	

/events/data/eventContext/worker/associateOID	Not Displayed	Y	The associateOID will get this from the Worker Call API.
/events/data/eventContext/payrollInstruction	Not Displayed	N	
/events/data/eventContext/payrollInstruction/itemID	Not Displayed	Y	The itemID will get this from the payroll Instructions GET call.
/events/data/eventContext/payrollInstruction/payrollGroupCode	Not Displayed	Y	The payrollGroupCode will get this in the payroll Instructions GET call.
/events/data/eventContext/payrollInstruction/payrollGroupCode/codeValue	Not Displayed	Y	
/events/data/eventContext/payrollInstruction/payrollGroupCode/shortName	Not Displayed	Y	
/events/data/eventContext/payrollInstruction/payrollFileNumber	Not Displayed	Y	The payrollFileNumber will get this in the payroll Instructions GET call.
/events/data/eventContext/payrollInstruction/generalDeductionInstruction	Not Displayed	Y	
/events/data/eventContext/payrollInstruction/generalDeductionInstruction/deductionCode	Not Displayed	Y	
/events/data/eventContext/payrollInstruction/generalDeductionInstruction/deductionCode/codeValue	Not Displayed	Y	
/events/data/eventContext/payrollInstruction/generalDeductionInstruction/deductionCode/shortName	Not Displayed	Y	
/events/data/transform/effectiveDateTime	Effective Date	Y	

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	Response Condition	messageTxt	GitHub Sample Request Payload	GitHub Sample Response Payload	Tips to Handle
200 OK	When the request is processed successfully.	NA	deduction-stop-200.request.json	deduction-stop-200.response.json	
400 Bad Request	When the invalid payrollFileNumber is passed in the request.	{ "userMessage": "messageTxt": "payrollFileNumber is invalid."}]}	deduction-stop-400.invalid-payroll-fileNumber-request.json	deduction-stop-400.invalid-payroll-fileNumber-response.json	Make sure a valid payrollFileNumber is passed in the request.
400 Bad Request	When the invalid itemID is passed in the request.	{ "userMessage": "messageTxt": "itemID is invalid."}]}	deduction-stop-400.invalid-itemID-request.json	deduction-stop-400.invalid-itemID-response.json	Make sure a valid itemID is passed in the request.
400 Bad Request	When the effectiveDate is passed as previous year.	"userMessage": { "messageTxt": "effectiveDate is invalid." }	deduction-stop-400.invalid-effective-date-request.json	deduction-stop-400.invalid-effective-date-response.json	Make sure the effectiveDate is the same as or later than the position start date.