

Chapter 3

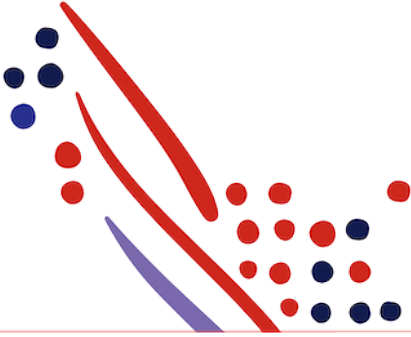
Frequently Asked Questions

from Direct Deposit API Guide for RUN Powered by
ADP Guide

Published on
Feb 24, 2020 1:50PM

Last modified
Feb 11, 2022 4:25PM





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.

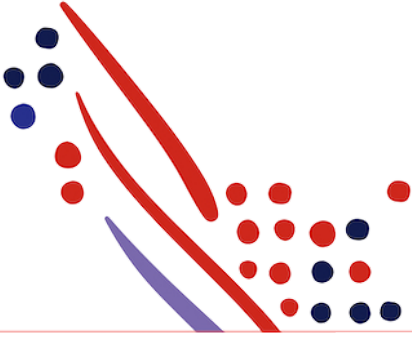
All other trademarks are the property of their respective owners.

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
Feb 24, 2020 1:50PM

Last modified
Feb 11, 2022 4:25PM



Chapter Contents

Chapter 3

Frequently Asked Questions

Question 1: How many types of direct deposits are supported and what is the maximum number of direct deposits that can be added through the API?

Question 2: Which APIs can my application use to retrieve the actual amount getting distributed into each direct deposit accounts for a payroll run?

Question 3: Can my application add new direct deposit configuration for a worker with a future start date?

Question 4: Are direct deposit accounts verified?

Question 5: Are there different API end points to use for add, update, and delete direct deposits?

Question 6: How do you work with precedenceCode in the API Payload?

Frequently Asked Questions

Question 1: How many types of direct deposits are supported and what is the maximum number of direct deposits that can be added through the API?

There are four kinds of direct deposit distributions supported by RUN Powered by ADP and the maximum limit supported through the Direct Deposit API is also four. Users can view these in RUN Powered by ADP by selecting **Employees > Payroll > Direct Deposit > Deposit Distribution**.

- **Full Net** - This gets selected in RUN Powered by ADP when:
 - **distributionAmount/amountValue** is set to employees full net amount
Or
 - **distributionPercentage** is given as 100
Or
 - **remainingBalanceIndicator** is set to true (when there are no other direct deposits configured for the employee)
- **Partial-\$** - This gets selected in RUN Powered by ADP when **distributionAmount/amountValue** is set to some partial amount.
- **Partial-%** - This gets selected in RUN when **distributionPercentage** is given less than 100
- **Remainders** (Deposits the remaining amount to the account configured) - This gets selected in RUN Powered by ADP when **remainingBalanceIndicator** is set to true

Question 2: Which APIs can my application use to retrieve the actual amount getting distributed into each direct deposit accounts for a payroll run?

Your application can use the Payroll Output API. See the sample [PayrollOutputRes.json](#) response.

Question 3: Can my application add new direct deposit configuration for a worker with a future start date?

Yes, it's possible to add a direct deposit for an employee with future hire-date as long as the person is fully onboarded (not in the In Progress state, and was given the Employee Self Service access). See [Process Overview](#) for more information.

Question 4: Are direct deposit accounts verified?

ADP works with a third party to verify employees' direct deposit bank accounts. This verification checks the Account Number & Routing Number, usually in few seconds, whenever a new direct deposit is added to RUN Powered by ADP and marks it as one of the following:

- **Yes** - Validated, which Saves Direct Deposit details in RUN Powered by ADP.
- **No** - Direct deposit should not be saved in RUN Powered by ADP.
- **Don't know** - When it's not possible to confirm or deny, RUN Powered by ADP saves these, but marks them as **not-verified**.

These results are available to the practitioner in the RUN Powered by ADP login on each employee's **Direct Deposit** page.

API has a **prenotebypassIndicator** flag which should be set to **true** if the partner can verify and send the bank account details.

Question 5: Are there different API end points to use for add, update, and delete direct deposits?

The **/events/payroll/v1/worker.pay-distribution.change** end point must be used for add, update, and delete direct deposits as incremental updates are not allowed. For example, if there are three direct deposit accounts to be configured and two are already configured in the system, your application should send details of all three with their precedence code when updating the Direct Deposit Distribution. To remove all direct deposit information, your application should send a blank payload.

To remove a single direct deposit account when there are, for example, three available direct deposit accounts configured, your application should send the new change request with the distribution information of the remaining two accounts.

Question 6: How do you work with precedenceCode in the API Payload?

Precedence code tells RUN Powered by ADP the preferred direct deposit mode for the worker if more than one direct deposits are present. The **precedenceCode** can have values from 1 - 4.

For example, if a partner application is trying to set up four direct deposit accounts for a an employee, they could send details of all four in the **Change Pay Distribution** method call. When building the payload they could have the direct deposits positioned in a random manner, but assign a unique **precedenceCode** (between 1 to 4) for each direct deposit. On a successful POST, direct deposits would appear as **Direct Deposit One, Direct Deposit Two, Direct Deposit Three, and Direct Deposit Four** in the UI mapping to **precedenceCode 1, 2, 3, and 4** respectively.

Note: The lower **precedenceCode** always gets the priority.