

# DHL Integration

In this task, you aim to build a small integration with the [DHL](#) platform. You need to develop a script with a function that uses DHL REST APIs to get the tracking information by tracking number. If time permits, you can also attempt the extra task.

## Setup Steps:

To be able to create the integration with DHL, you will need to create your test account in the [Developer Portal](#).

Once signed up, you will need to create a new app by selecting the APIs needed to complete the task:

### CREATE APP

App name\*

Description

APIs\*

SELECT APIS

API's name

Shipment Tracking - Unified

Environment:

☒ Production (Europe)

CREATING AN APP WILL ALLOW YOU TO REQUEST CREDENTIALS TO THE AVAILABLE APIS \*

PERFORM THE FOLLOWING STEPS TO GET STARTED:

- 1. APP NAME AND DESCRIPTION**  
You need to provide an app name and description
- 2. SELECT THE NEEDED APIS**  
Select the needed APIs from the list and you will be prompted about the available environments and rate limits
- 3. APP STATUS**  
Your app is created and contains a status indicating if your key is already approved for use
- 4. READY TO GO**

## Task:

Your task now is to explore the **Shipment Tracking-Unified** Rest API documentation and develop a function that will return the tracking information (at least the **last tracking event** ) of a package by a tracking number. Please use one of the following default tracking numbers to test your function: **7777777770, 4112889060**

**The function should take as inputs the following parameters:**

Tracking Number - the tracking number of the package to be tracked

API Key - credentials to authenticate with the API

The function must return the **last tracking event**.

**Extra Task:**

Explore the DHL APIs and implement a function to get and return the list of all **DHL service point locations** within the specified radius from the given address.

The function should take as inputs the following parameters:

*Country code* - the country code (can be hardcoded)

*City* - the city (must not be hardcoded, you need to use the output of the previous task)

*Radius* - the function should work even if the value of this input is null/None

The function should output the **names of all found locations**.

**Submission Instructions:**

- You can use any programming language to complete this exercise. Focus on demonstrating your problem-solving skills, code quality, and understanding of core concepts.
- Please submit your code via a GitHub repository. Create a private repository and share access with us ([marcela@digitalgenius.com](mailto:marcela@digitalgenius.com), [erick@digitalgenius.com](mailto:erick@digitalgenius.com), [luke@digitalgenius.com](mailto:luke@digitalgenius.com)). You only need to provide us with the link to the private repository containing your solution.