# Getting started – HIE Testing Tool

Setup

This section outlines the needed software/frameworks for development or execution of HIE automation tests.

1. Install Eclipse (you need this only if you plan to implement automation steps and don’t have an IDE already) –

Download the latest version from <https://www.eclipse.org/downloads/> and follow the installation steps

1. Install Python –

Download python version – 3.7.6 or later from <https://www.python.org/downloads/>

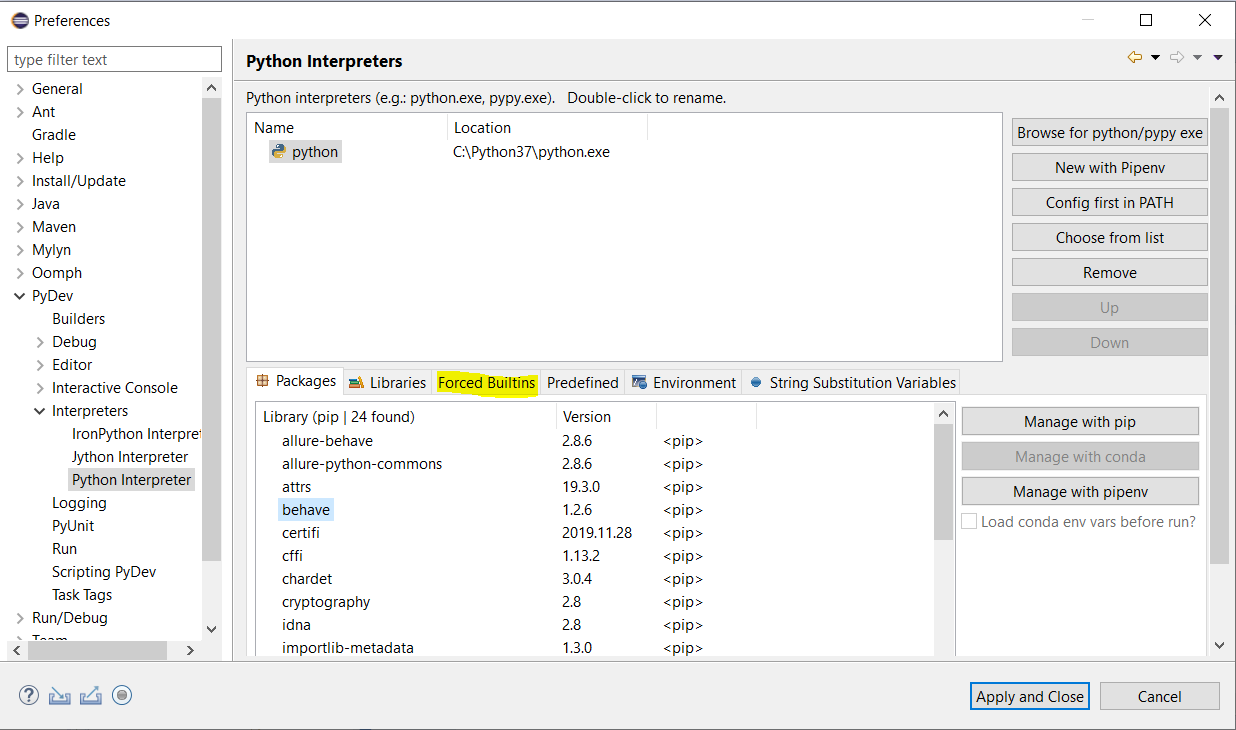
1. Install PyDev (only if you plan to implement automation steps and are using Eclipse ) –

Its better to install it from Eclipse. Please follow instructions under “Installing with the update site” on this page - <https://www.pydev.org/manual_101_install.html>

1. Install Behave – once you have python installed. The pip commands should be working. Just use the below command on your command line

pip install behave

Note: for developer : add behave to Forced Builtins, that way imported behave libraries will be loaded into eclipse



1. Install Allure - Goto section 2. Get started and follow the instructions based on your OS

<https://docs.qameta.io/allure/#_installing_a_commandline>.

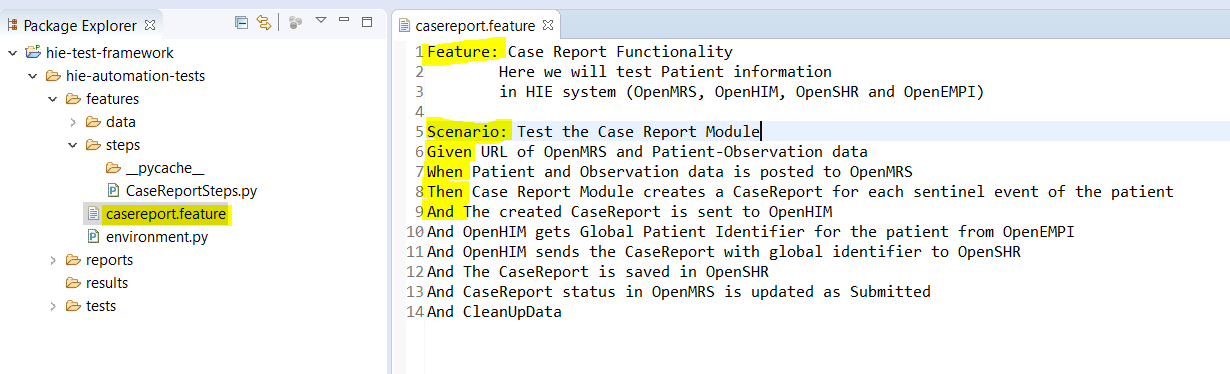
1. Install Allure-Behave: Use the below command from your command prompt

pip install allure-behave

## Scenarios and Steps

Behave uses 2 files to run the automation tests –

1. Scenarios – this file specifies the scenarios to be tested. Each scenario includes steps to test the scenario. These files are developed in Gherkin.

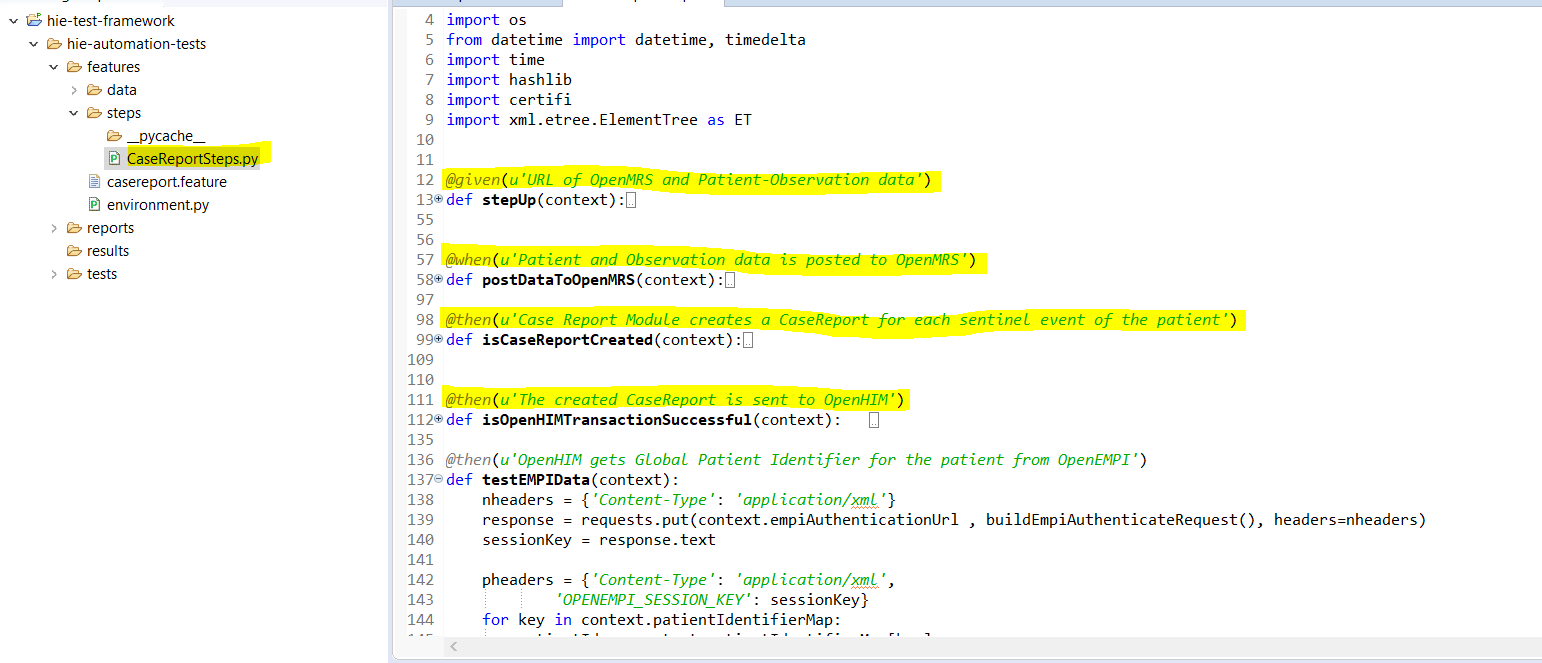


The feature is a functionality to be tested. Scenario is a part of the functionality to be tested. The main keywords of the Gerkin language are – Feature, Scenario, Given, When, Then and And.

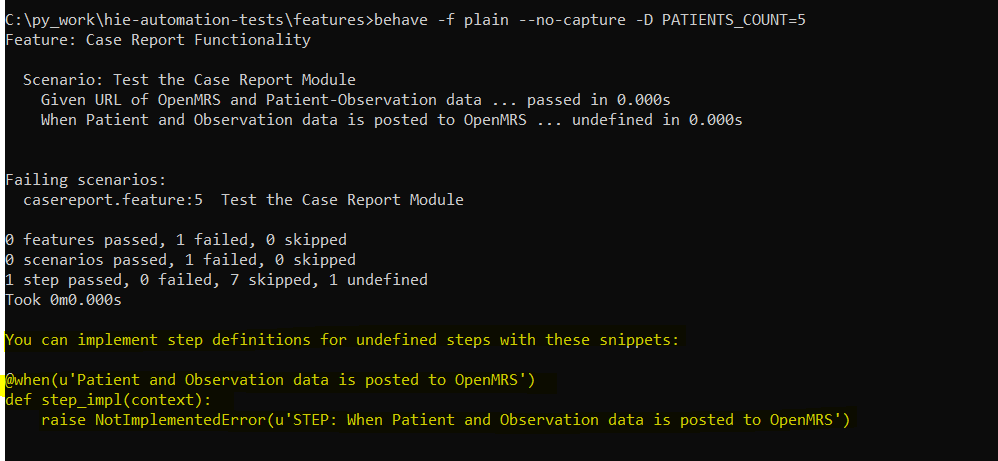
Other points to note for the features file is –

1. The project hierarchy has for the maintained. This file needs to reside under features folder
2. The feature file needs to have .feature extension.
3. Steps – the steps file contains the implementation of steps defined in scenarios.

As shown below the steps file needs to be under features/steps folder. You can also see that every Gerkin step translates to a method in the steps file.



Note for developer: Behave generates method skeleton for each step in the Gherkin file. We can use those method skeletons for step implementation.



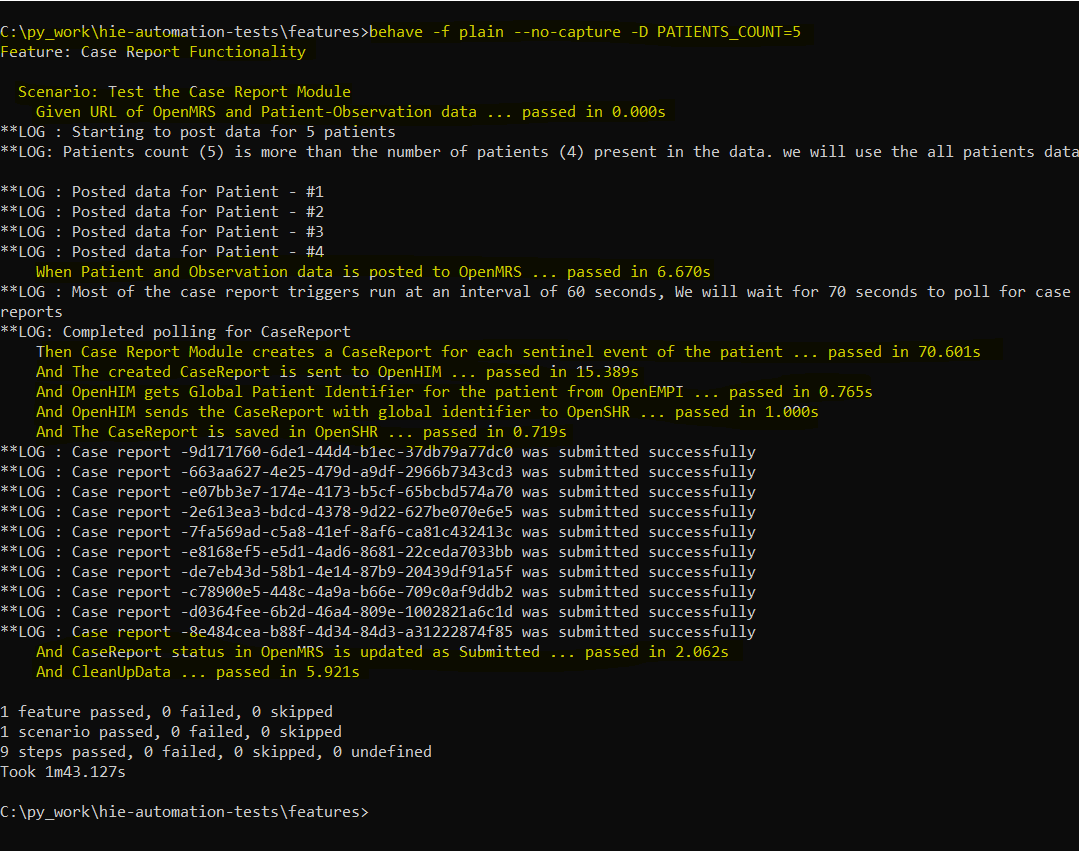
## Executing the tests

1. Download the code from - <https://git.cdc.gov/hie/hie-automation-test>
2. Navigate to the features folders on the command prompt and run the tests using the command
   1. Behave -f plain –no-capture -D PATIENTS\_COUNT=5

Currently, the test takes only 1 input – PATIENTS\_COUNT to specify the number of patients data to process.

Other things to note:

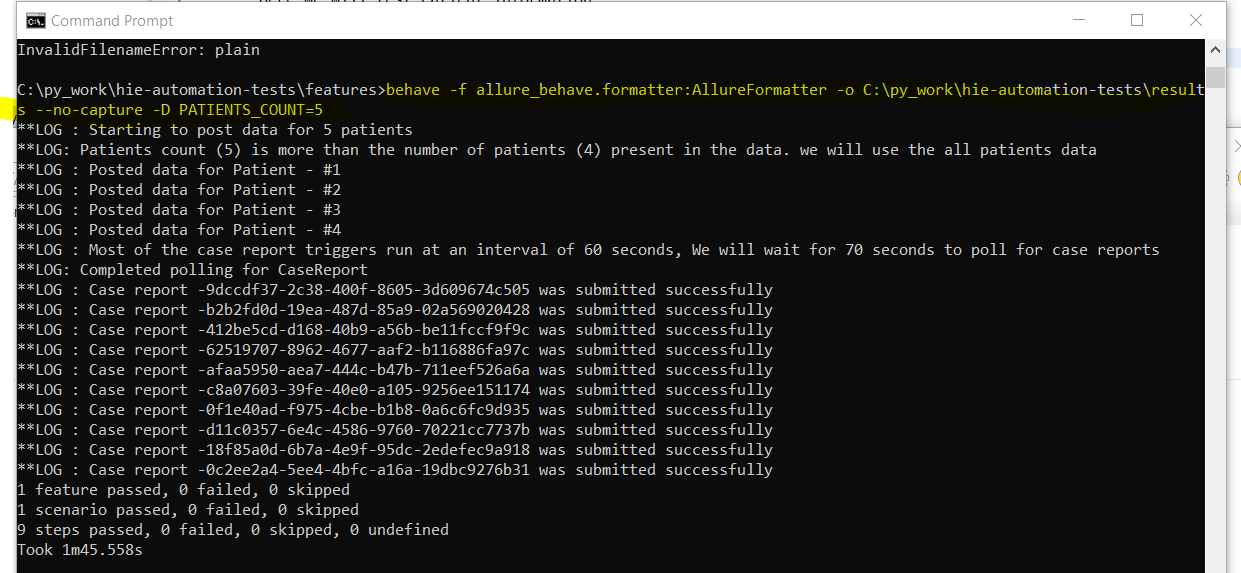
* The high lightened messages below are the Gherkin statements defined in the features file. You can see the status of each test as pass.
* The LOG messages show what’s happening as the tests are executing.

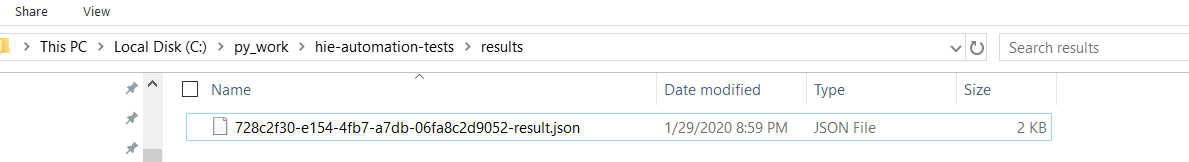


In future as we add more conditions, we can have more input parameters (instead of just PATIENTS\_COUNT) on the behave command line.

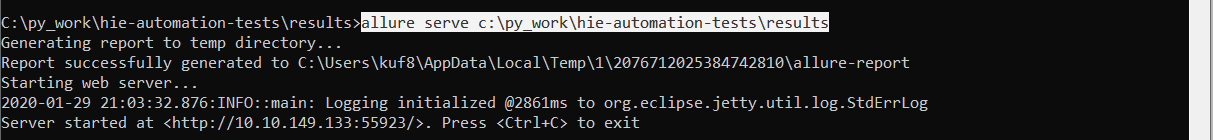
## Generating Reports

To Generate reports, we need to run our behave tests with different options – add allure\_behave.formatter:AllureFormatter before –no-capture. please see high lightened text below. Once the command executes and json file is created in the specified folder.





Navigate to this folder and execute the reports command. This is generate the report in HTML format and allows the user to see the steps executed and time taken.





## Developing advanced tests

Since we can reuse the steps, this reduces time taken to develop new tests. For example, in future if we want to post the patient data in another test, we can just reuse the below steps. Or if we want to test the transaction details in OpenHIM, we can reuse – The created CaseReport is sent to OpenHIM

