**Problem Statement for PAT WebCode-2**

Title: Using Python Selenium and Pytest Framework automate the web application [OrangeHRM (orangehrmlive.com)](https://opensource-demo.orangehrmlive.com/web/index.php/auth/login)

Test Objective: The goal of this project is to build a system that can automatically test the functionality of the Demo CRM web application

<https://opensource-demo.orangehrmlive.com/web/index.php/auth/login>

The system will use the Python Selenium framework, a tool for automating web applications on web browsers like Microsoft Edge, Google Chrome, Mozilla Firefox and Safari to navigate through the website and perform and tests various actions described in Test Description and then will verify that the web application is functioning correctly by checking the resulting web pages for expected content and behaviour and generate the Pytest based HTML reports.

You have to write your Test-case scripts using Page Object Model (POM) only. Use of Python OOPs concept is mandatory. Follow the Pylint framework for the nomenclature of Python variable, function names, Class names, Method Names and Object names.

Use of DDTF is mandatory

Tools: Use only Python Selenium. Pytest, Page Object Model, Data Driven Testing Format, Keyword Driven Testing, Hybrid Testing.

Usage of Explicit Wait is Mandatory!

Preconditions: Test cases contain all **Positive** and **Negative** scenarios.

URL: <https://opensource-demo.orangehrmlive.com/web/index.php/auth/login>

Test Suit:

Test Case-1: 1) Create an Excel file which will comprise of following as given below:

2) Using Data Driven Testing Framework (DDTF) you verify whether the login is successful or not for given usernames and passwords. After successful login, logout from the CRM.

3) Using cookies only verify whether login is successful or not

4) Generate Pytest base HTML reports for the same.

Test-Case-2: 1) Verify whether the home URL is working or not?

2) Generate Pytest base HTML reports for the same.

Test-Case-3 1) Verify whether the username, password input boxes are visible or not.

2) Generate Pytest base HTML reports for the same.

Test-Case-4: 1) Verify after successful login, whether the Menus Admin, PIM, Leave, Time, Recruitment, My Info, Performance, Dashboard are visible and clickable or not?

2) Generate Pytest base HTML reports for the same.

Test-Case-5: 1) Create new user, verify whether the new user is able to log-in into the CRM or not?

2) Generate Pytest base HTML reports for the same.

Test-Case-6 1) Verify from the Admin Menu that the new user exists in the records of the user or not?

2) Generate Pytest base HTML reports for the same.