

# EDYODA DIGITAL UNIVERSITY

**Python Developer Assessment - Test 2**  
***[EdYoda Mini-LMS - Coding Round]***

## Problem Details

The objective of this application is to manage EdYoda Learning Programs using the command line. There will be two types of users:

1. Program Manager: A user type who will have full access to the application. Think of the support team at EdYoda. The support people can perform create, read, update, and delete operations on modules, live sessions, students etc.
2. Students: A user type who will have permission to only view his/her module(s).

## Application Details:

- Program Manager should be able to perform CRUD(Create, Read, Update & Delete) operations for modules, units and students.
- **Modules:**
  - a. Modules are technical or non-technical skills. For example, Python, Databases, Data Analysis etc.
  - b. There can be multiple modules in the application.
  - c. Module creation: Following details should be required to create a module:
    - ModuleID: Auto-generated unique ID to identify the module.
    - Module Name: It should be a string with a maximum of 30 characters.
    - Module Start Date: Data type should be date.
    - Module End Date: Data type should be date.
    - Units: List of all the units in the module.
- **Units:**
  - a. These are activities in a module.
  - b. A unit can be of any of the following types - live sessions, quizzes, assignments.

c. *Please note: In this application, the students will not attend live sessions or write quizzes/assignments. You do not have to implement a quiz or assignment feature.*

d. Unit creation: Following details should be required to create a unit:

- Unit Id: Auto-generated unique ID to identify the unit.
- Unit Type: Live Session, Quiz, Assignment.
- Unit Title: It should be a string with a maximum of 100 characters.
- Unit Start Date: Data type should be date.
- Unit End Date: Data type should be date.
- ModuleID: The moduleID of the module this unit will be a part of.

- **Students:**

- a. Students are the users who will be enrolled in one or more modules and can view the list of modules and units.
- b. There can be multiple students in a module. One student can be a part of multiple modules.
- c. Student Creation: Following details should be required to create a student:

- Full Name: It should be a string with a maximum of 100 characters.
- Mobile Number: It should be a number with exactly 10 digits. The mobile number needs to be unique for a student which means no-two students can have the same mobile number.
- Email: It should be a string with a maximum of 100 characters. The email address needs to be unique for a student which means no-two students can have the same email address.
- Password: It should be a string with a minimum of 8 characters and a maximum of 16 characters.
- ModuleID: The ModuleID of the module in which the

student needs to be enrolled.

### **Program Manager Features:**

- The program manager should see the following options on login:
  - a. Manage Modules
  - b. Manage Units
  - c. Manage Students
  - d. Logout
- **Manage Modules:** When the program manager selects this option they should see the following options:
  - a. Create a new module
  - b. View all modules – This should show list of all the modules with following details:
    - ModuleID
    - Module Name
    - Module Status: Upcoming, Ongoing or Completed based on the module start and end dates. If module start date is in the future then status will be Upcoming. If the module end date is in the past then the status will be Completed. If the current date is on or in between the start and end date of the module then the status will be Ongoing.
  - c. View module details – This should show all module details based on the ModuleID. The details should include the following:
    - Module Id
    - Module Name
    - Module Start Date
    - Module End Date
    - Units
    - Module Status: Upcoming, Ongoing or Completed based on the module start and end dates.

- d. Update module – Based on the ModuleID, the program manager should be able to update all the details of a module.
  - e. Delete module – Based on the ModuleID, the program manager should be able to delete all the details of a module.
- **Manage Units:** When the program manager selects this option they should see the following options:
  - a. Create a new unit
  - b. View all units – This should show list of all the units with following details:
    - UnitID
    - Unit Name
    - Unit Type
    - Module Name
  - c. View module details – This should show all unit details based on the UnitID. The details should include the following:
    - UnitID
    - Unit Name
    - Unit Start Date
    - Unit End Date
    - Module Name
    - Unit Status: Upcoming, Ongoing or Completed based on the unit start and end dates.
  - d. Update unit – Based on the UnitID, the program manager should be able to update all the details of a unit.
  - e. Delete unit – Based on the UnitID, the program manager should be able to delete all the details of a unit.
- **Manage Students:** When the program manager selects this option they should see the following options:
  - a. Create a new student
  - b. View student details – This should show all the details of a student based on the contact number. The details should include the following:
    - Full Name

- Mobile Number
- Email
- Module Name List
- c. Update student – Based on the contact number, the program manager should be able to update all the details of a student.
- d. Delete unit – Based on the contact number, the program manager should be able to delete all the details of a student.

### **Student Features:**

- Student should see following options on login:
  - a. View Today's Schedule – If the current date is between the start and end date of a unit then that unit will be visible in the schedule. This unit should be a part of modules which are enrolled by this student. You will have to show following details:
    - Unit Name
    - Unit Type
    - Module Name
  - b. View My Modules
    - Show a list of all the modules the student is enrolled in.
  - c. Update Profile
    - Give an option to update his/her all the details
  - d. Logout

### **Common Functionalities:**

- Login – Both program manager and student can login to the application.
- Registration:
  - a. By default create one program manager with following credentials – username: admin and password: admin. There is no functionality to register more program managers.

- b. Students cannot register themselves. The program manager will create a student only then that student can login to the application.
- If any of the validations fail then show appropriate error messages. For example, if someone tries to enter a name in place of a contact number then an error should be displayed saying, "Please enter a valid contact number".
- It is not mandatory to use a database for storing data. You can use either a database or json files.

## Grading Details

- **Total Grades:** 200
- **Passing Grades:** 140

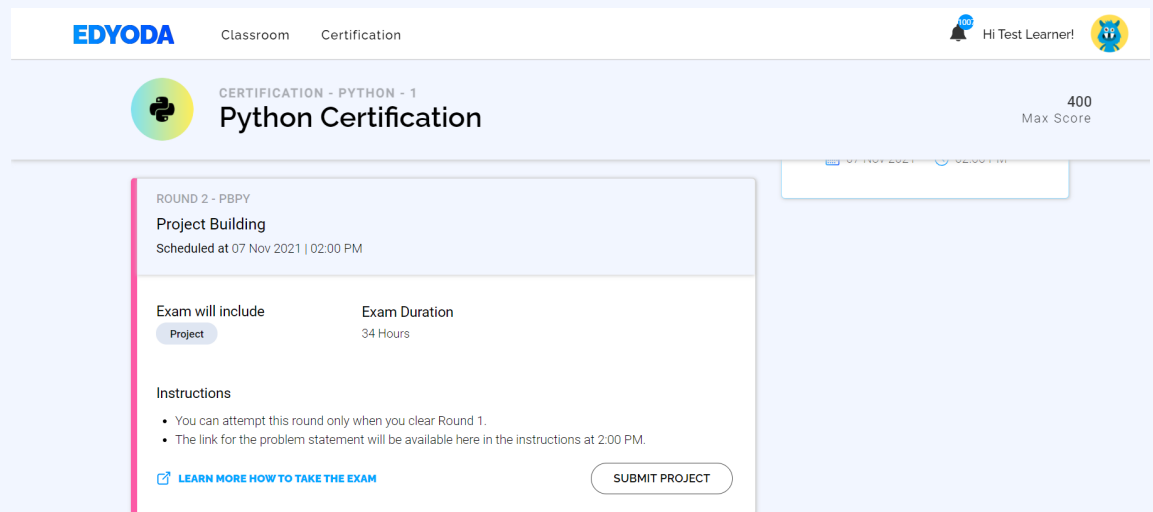
## Result/Report Details

- The result, grades and reports will be shared on 08th January 2022.

## Submission Details

- You have to submit the code URL. Please upload all your code in a folder on Google drive. Please ensure that the folder is available publicly or else the grading team will give you zero marks. To check that please try to open the URL in incognito mode. You have to share the public URL of this google drive folder.
- *Please ensure that you DO NOT COPY the code from your batch mates or the internet. Our proctoring tool will detect the plagiarism and you will receive zero marks in the exam.*
- You have to submit the project video. You can upload the video on Vimeo, Youtube, Google Drive etc. Please ensure that the video is publicly available. To check that please try to open the URL in incognito mode. Please follow the video structure as explained in the video details section below.

- You have to submit both the URLs on the certification details page on the classroom portal. Follow the steps below:
  - Goto Round 2 card on certification details page



EDYODA Classroom Certification

Hi Test Learner!

CERTIFICATION - PYTHON - 1

Python Certification 400 Max Score

ROUND 2 - PBPY

**Project Building**

Scheduled at 07 Nov 2021 | 02:00 PM

Exam will include

Project

Exam Duration

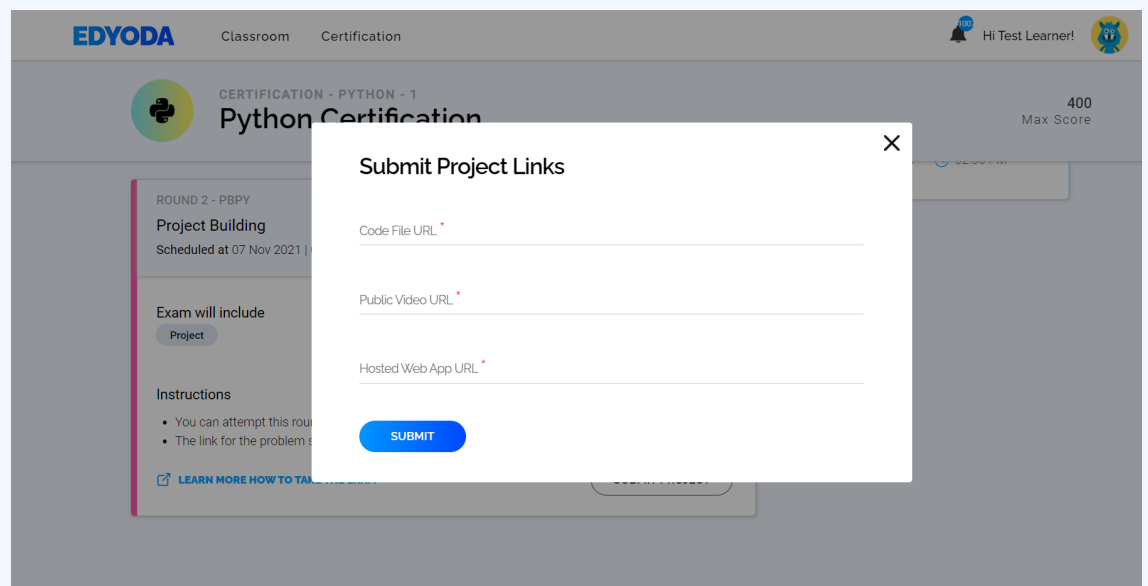
34 Hours

Instructions

- You can attempt this round only when you clear Round 1.
- The link for the problem statement will be available here in the instructions at 2:00 PM.

[LEARN MORE HOW TO TAKE THE EXAM](#) [SUBMIT PROJECT](#)

- Click on “Submit Project” button
- Fill the form in the popup and submit. Your project does not require a hosted web app URL so you can simply paste the video URL itself.



EDYODA Classroom Certification

Hi Test Learner!

CERTIFICATION - PYTHON - 1

Python Certification 400 Max Score

ROUND 2 - PBPY

**Project Building**

Scheduled at 07 Nov 2021 | 02:00 PM

Exam will include

Project

Instructions

- You can attempt this round only when you clear Round 1.
- The link for the problem statement will be available here in the instructions at 2:00 PM.

[LEARN MORE HOW TO TAKE THE EXAM](#) [SUBMIT PROJECT](#)

**Submit Project Links**

Code File URL \*

Public Video URL \*

Hosted Web App URL \*

[SUBMIT](#)

- Please ensure that you make your submissions before **13th Dec'21 11:59 PM IST** sharp. **No late submissions will be accepted.**

## Video Details:



- The duration of the video should be max 20 minutes.
- Total marks for the video is 100.
- You can record the video using zoom, loom or any other screen recording tool.
- Please ensure that your video is also visible on the bottom right side of the screen.
- The video needs to be structured in 3 sections:
  - **Webapp Walkthrough (Max 3 minutes)**
    - You have to give a product demo of your app. Basically, talk about all the functionalities and show them live. Do NOT show code in this step.
  - **Code Walkthrough (Max 10 minutes)**
    - Project Architecture
    - Screenwise code walkthrough
  - **Video QnA:**
    - Explain the class structure and the relation between them.
    - Explain how are you storing the module details provided by the program manager
    - Explain how directories are implemented internally.