Home Assignment <10>: Bug Tracking System

Learning Objective:

The objective of this assignment is to simulate a simple bug tracking system where bugs can be added, updated, and displayed using Python classes and dictionaries. This will help you connect Python programming with real-world QA and project management scenarios.

Expected Completion Time:

Best Case: 20 minutes
Average Case: 30 minutes

Assignment Details:

Create a Python program to manage bug records using a class and a dictionary.

Requirements:

a) Create a class named `BugTracker`.
b) Inside the class, maintain an attribute `bugs` as a dictionary, where each key is a bug ID and the value is another dictionary containing details (`description`, `severity`, `status`).

c) Implement the following methods:

1. `add_bug(bug_id, description, severity)` → adds a new bug with status `"Open"`.
2. `update_status(bug_id, new_status)` → updates the status of a given bug (e.g., Open → In Progress → Closed).
3. `list_all_bugs()` → prints details of all bugs in a readable format.

d) In the main section (`if __name__ == "__main__":`):

- Create a `BugTracker` object.
- Add at least three bugs with different IDs, descriptions, and severities.
- Update the status of one bug to `"In Progress"` and another to `"Closed"`.
- Display all bugs with their details.

Expected Outcome:

Upon completion of this assignment, you should be able to:

- Create and manage a dictionary inside a Python class.
- Add, update, and display structured records.
- Use OOP principles to model a simple bug tracking system.
- Relate Python programming to real-world QA and defect management tasks.