

Python Practical Exam

Section A – Core Python (25 Marks)

1. Functions (10 Marks)

- Write a Python function `is_prime(n)` that checks if a number is prime.
- Then, write another function `prime_in_range(start, end)` that returns a list of all prime numbers between start and end.

2. String & List Operations (15 Marks)

- Given a string: "openAI develops AI tools in Python".
- Write a program to:
 1. Count how many times each word appears.
 2. Reverse the order of words.
 3. Display the unique words in sorted order.

Section B – Object-Oriented Programming (25 Marks)

3. Class & Inheritance (15 Marks)

- Create a class `Employee` with attributes: `emp_id`, `name`, and `salary`.
- Create another class `Manager` (inheriting from `Employee`) that has an extra attribute `department`.
- Write methods to:
 1. Display employee details.
 2. Display manager details (including department).

4. Encapsulation & Polymorphism (10 Marks)

- Add a method `bonus()` in both classes:
 - For `Employee`, return 10% of salary.
 - For `Manager`, return 20% of salary.
- Demonstrate polymorphism by calling `bonus()` on both objects.

Section C – File Handling & Exception Handling (20 Marks)

5. File Handling (10 Marks)

- Write a Python program that:
 1. Reads a text file `data.txt`.
 2. Counts the number of lines, words, and characters.
 3. Writes the result into another file `summary.txt`.

6. Exception Handling (10 Marks)

- Write a program that takes a number as input and divides 100 by that number.
- Handle possible exceptions:
 - Division by zero
 - Invalid input (non-numeric values)

Section D – Database & Advanced Problems (30 Marks)

7. Python with MySQL (15 Marks)

- Using mysql-connector-python, write a Python program that:
 1. Connects to a MySQL database.
 2. Creates a table students(id, name, marks).
 3. Inserts 3 sample records.
 4. Fetches and displays all records.

8. Algorithm Problem (15 Marks)

- Write a Python function longest_substring(s) that returns the longest substring without repeating characters.
- Example:
 - Input: "abcabcbb"
 - Output: "abc"