COMSATS University Islamabad Sahiwal Campus

(Department of Computer Sciences)

Course Title:	Object Oriented Programming				**Course Code:**	CSC461	**Credit Hours:**	4(3,1)
Course Instructor:	Ali Sher Kashif				**Programme Name:**	BS (CS)		
Semester:	3rd	**Batch:**	FA23	**Section:**	E	**Date:**	14-10-2024	
Time Allowed:					**Maximum Marks:**		10	
Student's Name:					**Reg. No.**	CUI/	/SWL	

Important Instructions / Guidelines:

- Submit **hard copy** of your assignment
- Submit your assessment as a typed document (e.g., PDF or Word).
- Include any necessary code snippets, visualizations, or diagrams to support your answers.
- Use proper citations and references if you refer to external sources.
- Ensure clarity, organization, and correct grammar in your responses.

Assignment_02

CLO-02: Apply Object-Oriented Programming principles to solve real-world scenarios.

Question 1:

Scenario:
You are hired as a software developer for a company that manages employee payroll systems. The company wants to ensure that sensitive employee information such as salary and bank account details is not exposed directly to external classes. The system should allow users to update their name and position but only authorized personnel should be able to modify the salary.

Task:

- Design a class named Employee that stores the following details: employeeID, name, position, salary, and bankAccountNumber.
- Ensure that sensitive information (salary and bankAccountNumber) cannot be directly accessed or modified by external classes.
- Provide appropriate public methods to:
    - Change the employee's name and position.

    - Update salary, but ensure that this can only be done through a special method, say updateSalary(), that simulates authorized access.

Implement the class in Java and explain how the concept of encapsulation ensures data protection in this scenario.

Question 2:

Scenario:
You are tasked with designing a software system for a university to manage different types of users such as Students, Professors, and Administrators. The university wants to keep track of general information like userID, name, and email, but each type of user has specific attributes:

- Students have a major and yearOfStudy.
- Professors have a department and researchArea.
- Administrators have a role and officeLocation.

Task:

- Create a base class UniversityUser with the general attributes userID, name, and email.
- Derive three classes: Student, Professor, and Administrator from UniversityUser.
    - Student should have additional attributes: major and yearOfStudy.
    - Professor should have additional attributes: department and researchArea.
    - Administrator should have additional attributes: role and officeLocation.
- Implement a method displayDetails() in each class to print the user's details, including their specific attributes.

Write the Java implementation for the above scenario and explain how inheritance allows you to avoid redundancy in your code.