ourdiu.com | Shared by Jobayer Ahmed Usamah - SWE (251)

Daffodil International University
Faculty of Science & Information Technology
Department of Software Engineering
Course Code: SE216; Course Title: Object Oriented Programming

Time: 40 Mins **Marks: 15**

1.	*a)*	A university is developing a **Research Grant Management System** to manage research funding for faculty members. Each faculty member is assigned a research grant, and the system must protect all information from unauthorized access. To achieve this, implement the program using the **principle of encapsulation.**
		
		The ResearchGrant class should contain the following **private data members**:
		
		• facultyId (String)
		• facultyName (String)
		• department (String)
		• availableGrant (double)
		
		The system must satisfy the following requirements:
		
		1. The data members must **not be directly accessible** from outside the class.
		2. Provide appropriate **getter** and **setter** methods to retrieve and update the faculty information (facultyId, facultyName, and department).
		3. The availableGrant must **not** be modified using a setter method. Instead, it should only be updated through the following methods:
		  ○ addGrant(double amount) – increases the available grant amount.
		  ○ useGrant(double amount) – decreases the available grant amount only if the requested amount does not exceed the available grant balance. Otherwise, display an appropriate message indicating insufficient grant funds.
		4. Implement a method displayDetails() to display all faculty and grant information.
		
		Task:
		
		Analyze the above scenario and implement the ResearchGrant class in Java using appropriate **access modifiers** to achieve **data encapsulation**. Then, create another class containing the main() method, instantiate at least one ResearchGrant object, and demonstrate how the object is accessed and modified **only through the provided methods**, without directly accessing its data members.
		
		Note: Write the complete Java program, including all required methods and the main() method..
			[8]

	b)	A university is developing an **Academic Project Management System** to manage faculty supervisors and student research projects. Every person in the system shares common information such as **name** and **age**, so the system should include an **abstract class** named Person that contains the attributes name (String) and age (int), along with an abstract method displayInfo(). The system has two specialized classes, Supervisor and Student, both of which inherit from the Person class. A Supervisor has an additional attribute researchArea (String) and maintains a list of supervised students using an array of Student objects (Student[]). The class also provides the methods assignStudent(Student s) to assign a student under supervision and getStudentCount() to return the total number of supervised students. A Student has the additional attributes studentId (String) and projectTitle (String). Based on the above requirements, **draw a UML class diagram** that clearly illustrates the classes, attributes, constructors, methods, access modifiers, inheritance relationship, and the association between the Supervisor and Student classes.	**[7]**