ourdiu.com | Shared by Idba

Daffodil International University
Faculty of Science & Information Technology
Department of Software Engineering
Midterm Examination, Fall 2025
Course Code: SE 311; Course Title: Design Pattern
Sections & Teachers: RJM(A,B), STA(C,D,M), NML(E,F,G,H), MRN(I,J,K,L), DB(N)
Time: 1:30 Hrs. **Marks: 25**

Answer ALL Questions
[The figures in the right margin indicate the full marks and corresponding course outcomes. All portions of each question must be answered sequentially.]

1.	*a)*	You are designing a restaurant order system. The current **OrderManager** class is messy:
		class OrderManager {
		String order1, order2, order3;
		int qty1, qty2, qty3;
		double price1, price2, price3;
		void addOrder(String o1, int q1, double p1,
		String o2, int q2, double p2,
		String o3, int q3, double p3) {
		order1=o1; qty1=q1; price1=p1;
		order2=o2; qty2=q2; price2=p2;
		order3=o3; qty3=q3; price3=p3;
		}
		double calculateBill(String orderName) {
		switch(orderName) {
		case "Pizza": return qty1*price1 + qty2*price2 + qty3*price3;
		case "Burger": return qty1*price1 + qty2*price2 + qty3*price3;
		default: return 0;
		}}}
		Detect the code smells in the given code and Refactor the system to remove all the above code smells while keeping functionality intact.
			[Marks-6]	*CLO-1 Level-3*

	b)	You are developing a notification system for a messaging application. The system allows users to send different types of notifications depending on user preference: • Some notifications are sent via email. • Some are sent as text messages. • Some appear as push alerts. Each notification type has its own delivery mechanism, but the main application should not create these notifications directly. Instead, there is a part of the system responsible for producing notifications. When a notification is needed, the system decides which type to create based on the user's choice and then sends it, without the requester needing to know the exact type. Classes/Interface to Use • Notification (interface) • EmailNotification, SMSNotification, PushNotification (concrete classes) • NotificationFactory (class for suitable pattern) • AppClient (client class) **Build** the classes using proper JAVA code so that the client can request and send notifications without needing to know the exact notification type.	[Marks-7]	

ourdiu.com | Shared by Idba

	c)	**Show** the output and Identify which exception in the code is a checked exception and which is an unchecked exception.
		`import java.io.IOException;`
		`class StudentScores {`
		`  public static void main(String[] args) {`
		`  try {`
		`  readScores(false);`
		`  } catch (IOException e) {`
		`  System.out.println("An Exception occurred!");`
		`  }`
		`  int[] scores = {85, 90, 120};`
		`  for (int s : scores) {`
		`  if (s < 0 || s > 100) {`
		`  throw new RuntimeException("Invalid score: ");`
		`  }`
		`  }`
		`  System.out.println("All scores are valid!");`
		`  }`
		`public static void readScores(boolean fileExists) throws IOException {`
		`  if (!fileExists) {`
		`  throw new IOException();`
		`  }`
		`  System.out.println("File read successfully.");`
		`  }`
		`}`
			[Marks-2]	

2.	a)	**Classify** the difference between lazy and eager loading with examples.	[Marks-2]	
	b)	In an application, there is a single logging system called Logger that records events and messages. All modules in the application must use the same Logger object, and no module should be able to create another logging object. **Draw** the class diagram showing the Logger class and a client module that uses it, clearly indicating how the client accesses the single Logger object.	[Marks-2]	
	c)	You are developing a graphic editor application where users can create various shapes. The application currently supports Circle and Rectangle shapes. Users should be able to duplicate existing shapes and then modify their properties (such as color or size) without changing the original shape. The classes that must be used are: • Shape (interface) • Circle (concrete class) • Rectangle (concrete class) • GraphicEditor(client class) **Construct** the above classes using suitable design pattern so that a user can create a shape, duplicate it, and modify the duplicate independently of the originals.(proper JAVA code)	[Marks-6]	*CLO- 2 Level-3*