ISLAMIC UNIVERSITY OF TECHNOLOGY(IUT)

ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

CSE 4604: Software Testing & Quality Assurance Lab
Lab 3

Objective:

- Demonstrate the learning of Junit
- Advanced JUnit topics
- Introduction to Mocking in unit testing

Scenario 1:

The order must go through a fixed flow:

1. Validate cart
2. Apply discount
3. Calculate total

Class Under Test (OrderService)

```java
public class OrderService {
    public boolean validateCart(List<String> items) {
        return !items.isEmpty();
    }

    public double applyDiscount(double total, double discountPercent) {
        return total - (total * discountPercent / 100);
    }

    public double calculateTotal(List<Double> itemPrices) {
        return itemPrices.stream().mapToDouble(Double::doubleValue).sum();
    }
}
```

Tasks 1:

- Write three test methods in the expected business order.
- Use naming or annotations to enforce the order: `test_1_validateCart`, `test_2_applyDiscount`, `test_3_calculateTotalAfterDiscount`..
- Print messages to confirm execution order.

Scenario 2:

Different promotions offer different discount rates.

Tasks 2:

- Use parameterized tests to verify the `applyDiscount(total, discountPercent)` method.
  Provide test cases like:
  (100.0, 10.0) → 90.0
  (200.0, 25.0) → 150.0
  (50.0, 5.0) → 50.0
- Add boundary values for `discountPercent`

Scenario 3:

The `PaymentProcessor` class connects to a third-party service. You need to mock this in your test.

PaymentProcessor Interface

```java
public interface PaymentProcessor {
    String processPayment(String userId, double amount);
}
```

Class Under Test:

```java
public class CheckoutService {
    private final PaymentProcessor processor;

    public CheckoutService(PaymentProcessor processor) {
        this.processor = processor;
    }

    public String checkout(String userId, double totalAmount) {
        return processor.processPayment(userId, totalAmount);
    }
}
```

Task 3:

- Mock PaymentProcessor to return "SUCCESS" when called with any user ID and amount.
- Verify:
  - That the processPayment() method is called exactly once.
  - That the returned message is "SUCCESS".