**Exercise 1: Identify Functional vs. Non-Functional Requirements**

For each requirement below, determine whether it is **Functional (F)** or **Non-Functional (NF)**.

1. The system should allow users to reset their password via email.
2. The application should load within 2 seconds for 95% of users.
3. A user should be able to register using their email and phone number.
4. The system should support up to 10,000 simultaneous users.
5. The user should receive an email confirmation after placing an order.
6. The website should be accessible for users with disabilities as per WCAG 2.1 guidelines.
7. The application should use AES-256 encryption for sensitive data.
8. A logged-in user should be able to update their profile details.
9. The system should be available 99.9% of the time (uptime requirement).
10. The shopping cart should automatically save selected items for up to 24 hours.

**Exercise 2: Test Case Writing for Functional Requirements**

Write **two test cases** for the following functional requirement:

📌 **Requirement:**  
*"A user should be able to reset their password via email by receiving a reset link."*

✍ **Write test cases considering positive and negative scenarios.**

**Exercise 3: Test Plan for a Non-Functional Requirement**

Consider the following **non-functional requirement**:

📌 **Requirement:**  
*"The system should handle up to 10,000 concurrent users without performance degradation."*

**Task:** Write a **basic performance test plan** including:

1. **Test Objective**
2. **Test Strategy**
3. **Tools Used**
4. **Success Criteria**

**Exercise 4: Exploratory Testing for Non-Functional Aspects**

📌 **Scenario:** You are testing a mobile banking application. List five exploratory test ideas for **non-functional** aspects such as security, usability, and performance.