Manual Testing:

Question 1: You are assigned to test a new login feature for a web application. How would you approach creating test cases for this feature? What aspects would you consider?

Solution:

1. Understanding Requirements
2. Identifying Test Scenarios for positive test scenarios and negative test scenarios
3. Writing Detailed Test Cases ex: ID, Test Steps, Expected Result, Actual Result etc.
4. Prioritizing Test Cases
5. Test Execution and Reporting
6. Regression Testing
7. Automation Consideration

Question 2: A new e-commerce website has a shopping cart feature. Describe the steps you would take to manually test this feature from adding items to the cart to completing a purchase.

Solution: When manually testing the shopping cart feature of a new e-commerce website, it's essential to ensure that the entire workflow from adding items to the cart to completing a purchase is smooth, functional, and user-friendly.

1. Understanding Requirements
2. Preparing Test Data
3. Test Scenarios:
   3.1 Adding Items to the Cart for Single Item/ Multiple Items/ Adding Out-of-Stock Items
   3.2 Viewing and Updating the Cart - Updating Quantities/ Removing Items
   3.3 Applying Discounts and Coupons – Valid/Invalid
   3.4 Proceeding to Checkout - Starting Checkout/Guest vs. Registered User/Payment Information
   3.5 Completing the Purchase
4. Reporting and Logging Issues
5. Retesting and Regression Testing

Question 3: You've been given a mobile app with a search functionality. How would you ensure that the search feature works correctly across different inputs, including invalid and edge cases?

Solution: To ensure that the search functionality of a mobile app works correctly across different inputs, including invalid and edge cases:

1. Understanding the Search Feature Requirements - Search Criteria/Input type/ Expected Behavior
2. Preparing Test Data – Valid/Invalid data
3. Test Scenarios:
   3.1 Functional Testing - Valid Search Queries: Single Keyword/ Multiple Keywords/ Case Sensitivity
   3.2 Invalid Input Handling - Special Characters/No input/ Unsupported Formats like emoji
   3.3 Edge Case Testing - Very Long Search Terms
   3.4 Performance Testing

4. Test Execution - Execute the test cases across different devices and screen sizes to ensure consistency.
5. Regression Testing
6. Cross-Platform Testing - If the app is available on multiple platforms (iOS, Android), ensure the search functionality works consistently across all platforms.

Question 4: A company is launching a new feature that allows users to reset their password. How would you plan and execute the testing of this feature?

Solution: Testing a password reset feature is crucial as it directly impacts user security and experience. Here's how you can plan and execute testing for this feature:

1. Understanding the Feature Requirements
2. Identifying Test Scenarios
3. Requesting a Password Reset - Valid Email Address/Invalid Email Address
4. Receiving the Reset Instructions - Verify that the reset email is delivered promptly to the registered email address. And ensure that the reset link has a reasonable expiration time.
5. Resetting the Password - Valid Password Reset/ Invalid Password Attempts/ Mismatch in Password and Confirmation
6. Logging in with the New Password also check login with the old password to ensure it no longer works.
7. Security Testing
8. User Acceptance Testing (UAT)
9. Reporting and Documentation

Question 5: Imagine you are testing a banking application that includes a money transfer feature. What scenarios would you consider when creating test cases, and how would you prioritize them?

Solution: When testing a banking application with a money transfer feature, it's crucial to consider a variety of scenarios to ensure comprehensive coverage.

1. **Functional Scenarios:**

   1.1 Successful Money Transfer - User successfully transfers money from one account to another
   1.2 Insufficient Funds - User attempts to transfer more money than is available in the account.
   1.3 Transfer to an Invalid Account - User attempts to transfer money to a non-existent or closed account.
   1.4 Transfer Limit Exceeded - User tries to transfer an amount exceeding the daily or transaction limit.

2. **Negative Scenarios:**

   2.1 Transfer During System Downtime - User tries to transfer money when the system is under maintenance.
   2.2 Transfer With Invalid Credentials - User attempts to transfer money using invalid login credentials or session expiration.
   2.3 Error Handling and Messages - User receives clear and actionable error messages when a transfer fails.

3. **Security Scenarios:**

   3.1 Secure Authentication - User is required to authenticate securely (e.g OTP) before completing a transfer.
   3.2 Data Encryption - Ensure that all sensitive data (e.g account details, transaction information) is encrypted during the transfer process.

4. **Performance Scenarios:**

   4.1 High Volume Transactions - System handles a large number of transfer requests simultaneously.
   4.2 Transfer Processing Time - Measure the time taken to complete a transfer under normal and peak conditions.

5. **Edge Cases** - Transfer of Minimum Amount