# ISTQB Concepts

1. **Software testing activities should start**
   1. As soon as the code is written
   2. During the design stage
   3. When the requirements have been formally documented
   4. As soon as possible in the development life cycle
2. **In which order should tests be run?**
   1. the most important tests first
   2. the most difficult tests first (to allow maximum time for fixing)
   3. the easiest tests first (to give initial confidence)
   4. the order they are thought of
3. **When should you stop testing?**
   1. when time for testing has run out.
   2. when all planned tests have been run
   3. when the test completion criteria have been met
   4. when no faults have been found by the tests run
4. **A program validates a numeric field as follows:**

**Values less than 10 are rejected, values between 10 and 21 are accepted, values greater than or equal to 22 are rejected, which of the following input values cover all the equivalence partitions?**

1. 10,11,21
2. 3,20,21
3. 3,10,22
4. 10,21,22
5. **Using the same specifications as question 4, which of the following covers the MOST boundary values?**
   1. 9,10,11,22
   2. 9,10,21,22
   3. 10,11,21,22
   4. 10,11,20,21
6. **Regression testing should be performed:**
   1. every week
   2. after the software has changed
   3. as often as possible
   4. when the environment has changed
   5. when the project manager says
7. **Order numbers on a stock control system can range between 10000 and 99999 inclusive. Which of the following inputs might be a result of designing tests for only valid equivalence classes and valid boundaries?**
   1. 1000, 50000, 99999
   2. 9999, 50000, 100000
   3. 10000, 50000, 99999
   4. 10000, 99999, 100000
8. **Beta testing is:**
   1. performed by customers at their own site
   2. performed by customers at the software developer's site
   3. performed by an Independent Test Team
   4. performed as early as possible in the lifecycle
9. **The difference between re-testing and regression testing is:**
   1. re-testing ensures the original fault has been removed; regression testing looks for unexpected side-effects
   2. re-testing looks for unexpected side-effects; regression testing ensures the original fault has been removed
   3. re-testing is done after faults are fixed; regression testing is done earlier
   4. re-testing is done by developers; regression testing is done by independent testers
10. **An important benefit of code inspections is that they:**
    1. enable the code to be tested before the execution environment is ready.
    2. can be performed by the person who wrote the code.
    3. can be performed by inexperienced staff.
    4. are cheap to perform.
11. **The most important thing about early test design is that it:**
    1. makes test preparation easier.
    2. means inspections are not required.
    3. can prevent fault multiplication.
    4. will find all faults.
12. **Enough testing has been performed when:**
    1. time runs out.
    2. the required level of confidence has been achieved.
    3. no more faults are found.
    4. the users won’t find any serious faults.
13. **How would you estimate the amount of re-testing likely to be required?**
    1. Metrics from previous similar projects
    2. Discussions with the development team
    3. Time allocated for regression testing
    4. a & b
14. **Reviews, static analysis, and dynamic testing have the same objective**
    1. identifying defects.
    2. fixing defects.
    3. a and b
    4. None
15. **Which is the BEST outcome from complete testing:**
    1. You have discovered every bug in the program.
    2. You have tested every statement, branch, and combination of branches in the program.
    3. You have completed every test in the test plan.
    4. You have reached the scheduled ship date.

# Manual Testing

This form to edit your personal profile, you need to change your photo and edit your profile info

1. Find 5 or more testcases and write 3 of them in details.

2. Find at least 2 existing issues and write them in detail.

Graphical user interface, application, Teams

Description automatically generated

# DB Testing

In a specific school, given that teachers can create assignments to be answered by students. Each teacher can create several assignments, and each assignment can target many students.

Students can also receive assignments from different subjects.

Teacher Table:

|  |  |
| --- | --- |
| ID | Name |
| 1 | Teacher 1 |

Assignments Table:

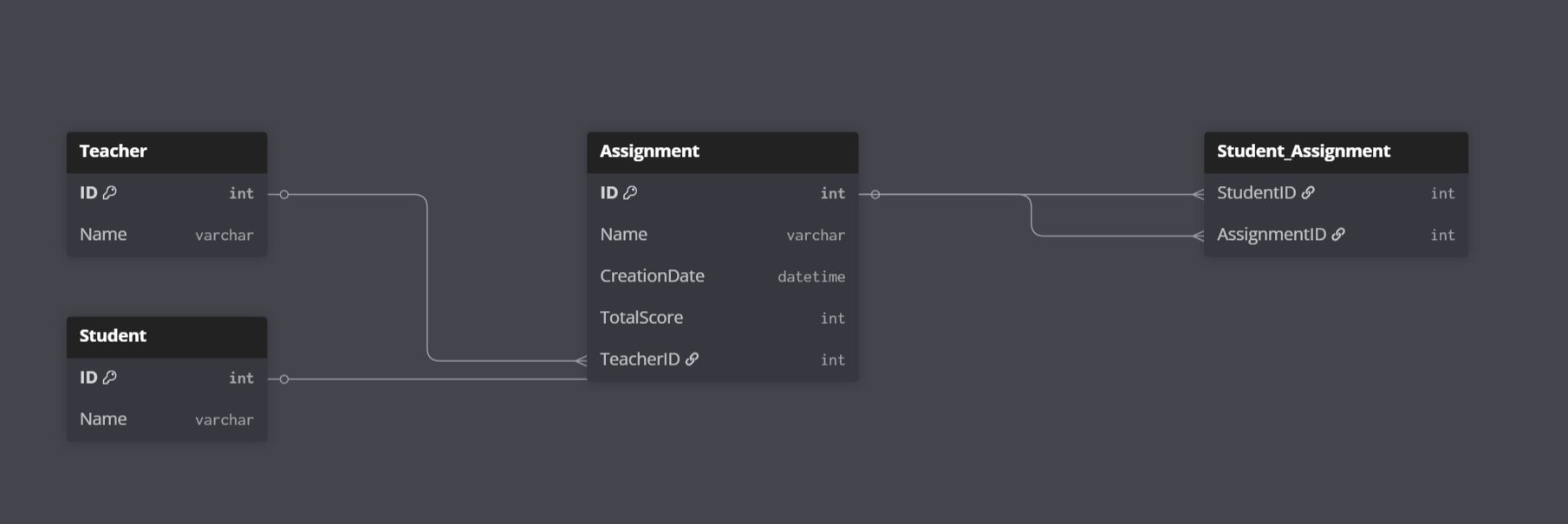
|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Creation Date | Total Score |
| 1 | Assignment 1 | 21/6/2021 04:00 PM | 100 |
| 2 | Assignment 2 | 23/6/2022 12:01 Am | 50 |

Students Table:

|  |  |
| --- | --- |
| ID | Name |
| 1 | Student 1 |

**Given the above info,**

1. How can these tables be related, mention the type of these relations, and draw DB Diagram?



1. Write SQL statement to retrieve all the teachers’ names that created assignments in ‘21/6/2021’.

SELECT DISTINCT t.Name

FROM Teachers t

JOIN Assignments a ON a.TeacherID = t.ID

WHERE DATE(a.CreationDate) = '2021-06-21';

1. Write SQL statement to retrieve all the teachers’ names and the number of assignments created by each teacher.

SELECT t.Name, COUNT(a.ID) AS AssignmentsCount

FROM Teachers t

LEFT JOIN Assignments a ON a.TeacherID = t.ID

GROUP BY t.ID, t.Name

ORDER BY AssignmentsCount DESC;

# Code Assessment

1. An ATM program is developed to check card type as the ATM will accept payment with Master Cards only would you mention what is the best practice for this piece of code:

If(Card.Type == “Premium”)

Return False;

Else If(Card.Type == “Master”)

Accept Transaction;

Else if(Card.Type == “Youth”)

Return False;

Else

Return Error;

**Solution:**

public class ATMProgram {

enum CardType {

MASTER, PREMIUM, YOUTH, OTHER

}

public static boolean processTransaction(CardType cardType) {

switch (cardType) {

case MASTER:

System.out.println("Transaction Accepted");

return true;

case PREMIUM:

case YOUTH:

System.out.println("Transaction Declined");

return false;

default:

throw new IllegalArgumentException("Invalid Card Type");

}

}

1. Find the error in the following piece of code:

int n = 0;

While( n<10) 🡪

{

int a =n+2;

int b =a+n;

System.out.println(“a = ”+a);

n=5;

}

System.out.println(“a = “+a);

System.out.println(“b = “+b);

System.out.println(“n = “+n);

Problems identified:

1. While should be lowercase while.
2. Inside the loop n = 5; sets n to 5 every iteration → infinite loop because n never increases beyond 5. Probably intended n = n + 1 or n++.
3. Variables a and b are declared inside the loop and not accessible outside — System.out.println("a = " + a); after the loop will not compile.
4. Improper quote characters in the example (ensure ASCII quotes).

The code after correct:

public static void main(String[] args) {

int n = 0;

int a = 0;

int b = 0;

while (n < 10) {

a = n + 2;

b = a + n;

System.out.println("a = " + a);

n++; // increment n, not set to 5

}

System.out.println("a = " + a);

System.out.println("b = " + b);

System.out.println("n = " + n);

}

# Automation Testing

1. What are the different types of locators in Selenium?
   1. id, name, className, tagName, linkText, partialLinkText, cssSelector, xpath.
2. What are the different types of Drivers available in WebDriver?
   1. ChromeDriver, FirefoxDriver, EdgeDriver, InternetExplorerDriver (legacy), SafariDriver, plus remote drivers like RemoteWebDriver / Grid and Appium for mobile.
3. What are the different types of waits available in WebDriver?
   1. **Implicit wait** (driver-wide default)
   2. **Explicit wait** (WebDriverWait + ExpectedConditions)
   3. **Fluent wait** (custom polling / timeout)
4. What is the difference between driver.quite() and driver.close()?
   1. driver.close() closes the current browser window/tab.
   2. driver.quit() closes all windows and terminates the WebDriver session.
5. Please automate the below task:   
   The tests should utilize best practice coding standards for maintainability, be aware of design patterns in your solution (POM).  
   The submitted code should be on GitHub.   
   **Task:**1-Navigate to " <https://opensource-demo.orangehrmlive.com/>"  
   2-Enter "Admin" as username   
   3-Enter "admin123" as password   
   4-Click on the login button   
   5-Click on Admin tab on the left side menu   
   6-Get the number of records found   
   7-Click on add button   
   8-Fill the required data   
   9-Click on save button   
   10-Verify that the number of records increased by 1   
   11-Search with the username for the new user   
   12-Delete the new user   
   13-Verify that the number of records decreased by 1   
   Feel free to use Selenium Java or Cypress JavaScript.  
   Inputs and Outputs values must not be hard codded in the script.
6. **Bonus task:**   
   Consider website   
   https://opensource-demo.orangehrmlive.com/   
   and step1 navigate to here,   
   Add candidate thru web apis using rest assured   
   https://opensource-demo.orangehrmlive.com/web/index.php/recruitment/viewCandidates   
   step 2:   
   delete any record through web apis using rest assured  
   https://opensource-demo.orangehrmlive.com/web/index.php/recruitment/viewCandidates