SQL:

1.Joins are used to join two or more tables an display as single table.

Or

Displaying the information from multiple tables at a time is known as join

Joins are different types:

- Equi join/inner join
- Non equi join
- Self join
- Cross join
- Outer join

Ex:select*from emp join dept on emp,deptno=dept.deptno;

2.select column1,column2,olumn3,count(*)as duplicate from table name GROUP BY column1,column2,column3 HAVING COUNT(*)>1;

3. ALTER TABLE mytable RENAME COLUMN old_column_name TO new_column_name;

4.Distinct keyword is used display the unique data present in the table.

Ex:select distinct(job)from emp;

5. select max(sal)from dept;

6. SELECT *FROM employee LIMIT 5;

7. SELECT * FROM employee ORDER BY employee_id DESC LIMIT 5;

Selenium

1. Write code for handling multiple windows

Solution: public class MultipleWindows {

public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver", "path/to/chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.get("https://example.com");

String main_window = driver.getWindowHandle();

WebElement new_window_link = driver.findElement(By.linkText("Open new window"));

new_window_link.click();

```
    try {
        Thread.sleep(2000);
      } catch (InterruptedException e) {
        e.printStackTrace();
Set<String> handles = driver.getWindowHandles();
      for (String handle : handles) {
        if (!handle.equals(main_window)) {
          driver.switchTo().window(handle);
          break;
        }
      }
String new_window_title = driver.getTitle();
      System.out.println("New window title: " + new_window_title);
driver.switchTo().window(main_window);
driver.quit();
   }
}
```

2. Write code for Cross Browser Testing

Solution:

```
public class CrossBrowserTest {
WebDriver driver;

@BeforeClass
   @Parameters("browser")
   public void beforeTest(String browser) {

 if (browser.equalsIgnoreCase("chrome")) {
        System.setProperty("webdriver.chrome.driver", "path/to/chromedriver.exe");
```

```
            driver = new ChromeDriver();
        }
        else if (browser.equalsIgnoreCase("firefox")) {
            System.setProperty("webdriver.gecko.driver", "path/to/geckodriver.exe");
            driver = new FirefoxDriver();
        }
        else if (browser.equalsIgnoreCase("ie")) {
            System.setProperty("webdriver.ie.driver", "path/to/IEDriverServer.exe");
            driver = new InternetExplorerDriver();
        }
    }

    @Test
    public void test() {
        driver.get("http://www.google.com");
    }
    @AfterClass
    public void afterTest() {
        driver.quit();
    }
}
```

3. Write code for handling Frames

Solution:

```
public class Frame {
public static void main(String[] args) {
    System.setProperty("webdriver.chrome.driver", "path/to/chromedriver");
    WebDriver driver = new ChromeDriver();
```

```
    driver.get("https://www.example.com");
driver.switchTo().frame(0);
     WebElement element1 = driver.findElement(By.id("element1"));
      element1.click();
     driver.switchTo().defaultContent();
     driver.switchTo().frame("frame2");
    WebElement element2  = driver.findElement(By.id("element2"));
      element2.sendKeys("some text");
      driver.switchTo().defaultContent();
     driver.quit();
  }
}
}
```

8. Write code for taking Screenshot

Solution:

```
class Screenshot {

 public static void main(String[] args) {
    System.setProperty("webdriver.chrome.driver", "path/to/chromedriver.exe");
    WebDriver driver = new ChromeDriver();
    driver.get("https://www.google.com");
     File screenshotFile = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
     try {
        FileUtils.copyFile(screenshotFile, new File("path/to/screenshot.png"));
         System.out.println("Screenshot saved successfully.");
     } catch (Exception e) {
         System.out.println("Failed to save screenshot: " + e.getMessage());
```

```
}
```

10. Write test scenario for Credit Cards

Solution:

1. open the payment page for credit cards.
2. enter correct credit card information Card Number, Expiry Date, CVV, Cardholder Name
3. Check to verify that the information entered is accurate and appears on the payment page.
4. Press the "Submit" button.
5. Make that the transaction was successful.
6. Verify the amount and other payment information on the payment receipt.
7. Examine the website to ensure the payment receipt is generated and displayed.
8. The payment receipt can be downloaded by selecting the "Download" option.
9. Check to see if the payment receipt downloaded properly.

11. Write code for handling multiple browsers and switch to new windows?

Solution:

```
public class MultipleBrowserHandling{
  public static void main(String[] args) {
      System.setProperty("webdriver.chrome.driver","path/to/chromedriver.exe");
      WebDriver chromeDriver = new ChromeDriver();
    chromeDriver.get("https://www.google.com");
     WebElement newWindowLink = chromeDriver.findElement(By.linkText("Click Here"));
      newWindowLink.click();
String currentWindowHandle = chromeDriver.getWindowHandle();
      for (String windowHandle : chromeDriver.getWindowHandles()) {
        if (!windowHandle.equals(currentWindowHandle)) {
          chromeDriver.switchTo().window(windowHandle);
          break;
```

```
        }
      }
 chromeDriver.findElement(By.id("someId")).click();
 chromeDriver.close();
      chromeDriver.switchTo().window(currentWindowHandle);
 System.setProperty("webdriver.gecko.driver", "path/to/geckodriver.exe");
      WebDriver firefoxDriver = new FirefoxDriver();
 firefoxDriver.get("https://www.bing.com");
  WebElement newWindowLink = firefoxDriver.findElement(By.linkText("Click Here"));
      newWindowLink.click();
String currentWindowHandle = firefoxDriver.getWindowHandle();
      for (String windowHandle : firefoxDriver.getWindowHandles()) {
        if (!windowHandle.equals(currentWindowHandle)) {
          firefoxDriver.switchTo().window(windowHandle);
          break;
        }
      }
 firefoxDriver.findElement(By.id("someId")).click();
   firefoxDriver.close();
      firefoxDriver.switchTo().window(currentWindowHandle);
 chromeDriver.quit();
      firefoxDriver.quit();
   }
}
```

13. How to perform upload files using selenium?

Solution:

```
public class FileUploadExample {
   public static void main(String[] args) {
      System.setProperty("webdriver.chrome.driver","C:/path/to/chromedriver.exe");
      WebDriver driver = new ChromeDriver();
      driver.get("http://mywebsite.com/upload");
       WebElement fileInput = driver.findElement(By.id("file-upload"));
       fileInput.sendKeys("C:/path/to/myfile.txt");
      WebElement submitButton = driver.findElement(By.id("submit-button"));
       submitButton.click();
      driver.quit();
   }
}
```

14. How to perform download files using selenium?

Solution:1. find_elements_by_*(locator)

2.Using Xpath

3.Using CSS selectors

Manual Testing

1. Write a Bug Report for a bug in the Whatsapp mobile application that is preventing users from sending video files in group chats.
   (Note: Consider that this workflow is working fine in private chat)

Solution: Product:

Whatsapp Mobile Application

Version:

Bug Description: Sending video files in group chats is not possible for users. Private chats are not impacted, this problem only affects group chats. Users encounter an error message stating that a video file cannot be sent while attempting to send it in a group chat.

Steps to Reproduce:

1. Launch the Whatsapp mobile app and log in with your account.

2. Register in or start a group chat with at least one other person.

3. To select a video file to send, tap the attachment symbol (+).

4. Choose a video from the device's gallery.

5.Press the "Send" button.

6. Take note of the error message and the video file not being sent.

2. Write a bug report for a bug in the login page that the submit button is not working.

Solution:

Title: Login page submit button not working

Severity: High

Description: The login page's submit button is broken. Nothing happens once a user enters their login information and clicks the submit button. Users who are unable to log into their accounts as a result of this bug are inconvenienced.

Steps to Reproduce:

Go to the login page

Enter valid login credentials

Click on the submit button

Expected Result: The user should be able to log into their account and be taken to the main page.

Actual Result: Nothing happens when the user presses the submit button.

7. Write any 5 Negative testcases.

Solution:
Example: Enter characters into an input area that are not allowed.
1. The field requirements message is displayed.
2. No action is taken on the registration.
3. Any other faults are not displayed.
4. There is no app crash.

8. Write a few test cases for the IRCTC Web Application.

Solution:

1: Login functionality

- Launch the IRCTC web application
- Enter valid credentials and click on the 'Login' button
- Verify that the user is redirected to the homepage after successful login
- Verify that an error message is displayed if the user enters invalid credentials4:

Cancel Ticket functionality

- Launch the IRCTC web application
- Login using valid credentials
- Click on the 'Booked Tickets' link in the menu
- Select the ticket to be canceled and click on the 'Cancel Ticket' button
- Enter the reason for cancellation and click on the 'Cancel Ticket' button

5: Seat Availability functionality

- Launch the IRCTC web application
- Enter the source and destination station codes, journey date, train number and class of travel
- Click on the 'Check Seat Availability' button
- Verify that the availability details for the selected train and class are displayed with the number of available seats and fare details.

10. Create 10 test case for amazon Website view

Solution:

- Enter Valid Username.
- Enter Valid Password.
- Click on “Forgot Password” Link and find out forgot password page.
- Click on Create new account button and verify if it takes to new account page.
- Select “Keep me signed in” option to retain login session.

Java

2. How do we reverse a string?

Solution: The input string is first converted to a String Buffer using the String Buffer method because the String class has a reverse () method. The string can then be reversed by using the reverse () method.

Example:

Input : s='abc'

output : s='cba'

3. Write a program that detects the duplicate characters in a string.

Solution:

```
public class Duplicate Characters

{

public static void main(String[] args)

{

String string1 = "programming";

int count;

char string[] = string1.toCharArray();

System.out.println("Duplicate characters: ");

for(int i = 0; i <string.length; i++)

{

count = 1;

for(int j = i+1; j <string.length; j++)

{

if(string[i] == string[j] && string[i] != ' ')

{

count++;

string[j] = '0';

 }
```

```
}
 if(count > 1 && string[i] != '0')
System.out.println(string[i]);
 }
 }
}
```

5. Write a program to demonstrate method overriding?

```
class A{
  public void display()
 {
System.out.println("a is displaying");
 }
}
class b extends A
{
public void display()
{
System.out.println("B is displaying");
}
public static void main( String args[]) {
B obj = new B();
obj.display();
 }
}
```

6. How is an infinite loop declared in Java?

Solution: . A loop statement is used to repeat statements or expression a certain number of times

There are three looping structures in java: for, while and do while.

For loop example:

```
public class A {

public static void main(String[] args)

{

for(int i=1;i<=5;i++)

{

System.out.println(i);

   }

 }

}
```

Example for while:

```
public class A {

public static void main(String[] args)

{

int i = 1;

while (i <= 6)

{

  System.out.println(i);

  i++;

}

   }

 }
```

Example for do-while loop:

```
public class A {
public static void main(String[] args)
{
int i = 0;
do
{
System.out.println(i);
i++;
}
while (i <= 5);
 }
}
```

9. How to read a file in Java?

fundamental methods are

read(): reads a single character.

read(char[]): reads an array of characters.

skip(long): skips some characters.

close(): closes the stream.

following methods are used to read the file

1.InputStreamReader

2.FileReader

3.BufferedReader

11. Write a program to generate the following output in java?

```
*
**
****
*****
******
```

Solution:

```
public class Main {

public static void main(String[] args) {

   int r = 5;

 for (int i = r; i >1; --i)

{

for (int j = 1; j < i; ++j)

{

 System.out.print("* ");

    }

    System.out.println();

  }

 }

}
```

13. How to find duplicate characters in a string in Java?

Solution: To find the duplicate character from the string, We count how many times each character appears in the string in order to identify the duplicate character. A character in the string likely has a duplicate entry if count is larger than 1.

16. How to sort array in java?

Solution:

```
class SA{

   public static void main(String args[])

   {

      int[] arr = { 6,0,=98,76};

      System.out.println("array is: ");

      for (int num : arr) {

         System.out.print(num + " ");
```

```
    }
    Arrays.sort(arr);
    System.out.println("\n sorted array is: ");
    for (int num : arr) {
       System.out.print(num + " ");
    }
  }
}
```

17. Write a program to do bubble sort on an array in java.

Solution:

```
public class BubbleSort {
   static void bSort(int[] arr) {
      int n = arr.length;
      int temp = 0;
       for(int i=0; i < n; i++){
            for(int j=1; j < (n-i); j++){
                 if(arr[j-1] > arr[j]){
                      //swap elements
                      temp = a[j-1];
                      a[j-1] = b[j];
                      b[j] = temp;
                 }
            }
       }
   }
   public static void main(String[] args) {
            int arr[] ={5,80,25,4,40,480,11};
```

```
        System.out.println("Before Bubble Sort");
        for(int i=0; i < a.length; i++){
            System.out.print(a[i] + " ");
        }
        System.out.println();
        bSort(a);
        System.out.println("  After  Bubble  Sort");
        for(int i=0; i < arr.length; i++){
            System.out.print(a[i] + " ");
        }
    }
}
```

20. Write a Java program to show a NullPointerException.

Solution:

```
class Test
{
   public static void main (String[] args)
   {

      String ptr = null;
     try
      {
       if (ptr.equals("gfg"))
            System.out.print("Same");
         else
            System.out.print("Not Same");
      }
```

```
    catch(NullPointerException e)
    {
      System.out.print("NullPointerException Caught");
    }
  }
}
```

23. How do you write an interface with default and static method?

Solution: It is used to create string of characters with a boundary between them. Similar or related like the comma, hyphen and others can now be used to build strings. prefix and suffix can be passed to the char sequence.

24. Write a Java program to print stars using for loop, where the number of stars printed shouldbe equal to the row number?

Solution:

```
public class star
{
  public static void main(String[] args)
  {
    int i, j;
    for(i=0; i<5; i++)
    {
      for(j=0; j<=i; j++)
      {
        System.out.print("* ");
      }
      System.out.print("\n");
    }
  }
}
```

26. Write a Java Program to print the below output: * 1 * 12 * 123 * 1234 * 12345 * 123456 * 1234567

Solution:

```
public class Main
{
   public static void main(String [] args)
   {

Scanner s = new Scanner(System.in);

 System.out.println("Enter the value of 'n' ");
      int n = s.nextInt();
      //for loop to print the series
      for (int i = 1; i <= n; i++)
      {
         for (int j = 1; j <= i; j++)
         {
            System.out.print(j);
         }
         System.out.print(",  ");
      }
   }
}
```

27. Write a Java program to read and write a file?

Solution:There are four methods

1.writeString() method

2.FileWriter Class

3.BufferedWriter Class

4.FileOutputStream Class

28. Implement factorial using recursion

Solution:

```
public class Ftrl {
   public static void main(String[] args) {
      int n = 6;
      long factorial = multiplyNumbers(n);
      System.out.println("Factorial of " + n + " = " + ftrl);
   }
   public static long multiplyNumbers(int n)
   {
      if (n >= 1)
         return num * multiplyNumbers(n - 1);
      else
         return 1;
   }
}
```

29. Implement multiple inheritances using an interface

Solution:

```
interface humanrun {
  void run();
}
interface humanjump {
  void jumpl();
}
class human implements humanrun , humanjump{
  public void eat() {
    System.out.println("human  is  running");
  }
  public void travel() {
    System.out.println("human  is  jumping");
```

```
  }
}
public class Demo {
  public static void main(String args[]) {
    Animal a = new Animal();
    a.eat();
    a.travel();

  }
}
public class Test {
  public static void main(String[] args)
   {
     flowers f1 = flower.rose;
      System.out.println(f1);
   }
}
```