1.Create 2 test cases, disable one using enabled = false, and run only the active test.

Program:

```
import java.time.Duration;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;
public class Task1 {
        WebDriver driver;
        WebDriverWait wait;
        @BeforeTest
          public void setUp() {
            driver = new ChromeDriver();
            driver.manage().window().maximize();
            wait = new WebDriverWait(driver, Duration.ofSeconds(20));
            driver.get("https://www.ebay.com/");
        }
  @Test(enabled = true)
  public void login() throws InterruptedException{
          driver.findElement (By.linkText("Sign in"))).click();
    Thread.sleep(3000);
    driver.findElement(By.id("signin-continue-btn")).click();
    Thread.sleep(3000);
    driver.findElement (By.id("pass"))).sendKeys("Jayanth@2003");
```

```
    driver.findElement(By.id("sgnBt")).click();
    Assert.assertTrue(driver.getPageSource().contains("My eBay"), "Login failed");
  }
  @Test(enabled = false)
  public void logout() {

driver.findElement(By.xpath("//*[@id=\"gh\"]/nav/div[1]/span[1]/div/button/span/span")).cli
ck();
          driver.findElement(By.xpath("//*[@id=\"s0-1-4-9-3[0]-0-9-
dialog\"]/div/div/ul/li[3]/a")).click();
  }
  @AfterTest
  public void afterTest() {
    driver.quit();
  }
}
```

---

2. Write a test to run the same test multiple times.

Program:

```
import java.time.Duration;
import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;
public class Task2_MultipleTimes {
```

```
        WebDriver driver;
          WebDriverWait wait;
          @BeforeTest
          public void setUp() throws InterruptedException {
             driver = new ChromeDriver();
             driver.manage().window().maximize();
             Thread.sleep(3000);
             driver.get("https://www.ebay.com/");
        }
 @Test(invocationCount = 3)//test will run 3 times
 public void multiplerun() {
        WebElement search=driver.findElement(By.id("gh-ac"));
        search.clear();
        search.sendKeys("phones");
        search.sendKeys(Keys.ENTER);
 }
 @AfterTest
 public void afterTest() {
    driver.quit();
 }
}
```

---

3. Write test cases for a dummy login page using @Parameters in testng.xml.

Program:

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
```

```
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Optional;
import org.testng.annotations.Parameters;
import org.testng.annotations.Test;
public class DummyLoginPage {
        WebDriver driver;
        @BeforeClass
          public void setUp() {
            //System.setProperty("webdriver.chrome.driver", "path/to/chromedriver");
            driver = new ChromeDriver();
            driver.manage().window().maximize();
            driver.get("https://example.com/dummy-login");
          }
 @Test
@Parameters({"username","password"})
 public void loginTest(@Optional("admin") String username,
       @Optional("admin123") String password) {
         WebElement userField = driver.findElement(By.id("username"));
    userField.clear();
    userField.sendKeys(username);
    WebElement passField = driver.findElement(By.id("password"));
    passField.clear();
    passField.sendKeys(password);
    driver.findElement(By.id("loginBtn")).click();
 }
 @AfterClass
 public void tearDown() {
    driver.quit();
```

```
  }
}
```

XML Document:

```
<suite name="DummyLoginPageSuite" parallel="classes" thread-count="1">
   <test name="LoginTest">
      <parameter name="username" value="admin"/>
      <parameter name="password" value="admin123"/>
      <classes>
         <class name="testNG_practice.DummyLoginPage"/>
      </classes>
   </test>
```

---

4. Write dependent test cases:

login()

search Product() (depends on login)

logout() (depends on search)

Program:

```
import java.time.Duration;
import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
```

```
import org.testng.annotations.Test;
public class Task4_DependencyTestCases {
    WebDriver driver;
    WebDriverWait wait;
    @BeforeClass
    public void setup() {
        driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.ebay.com/");
        wait = new WebDriverWait(driver, Duration.ofSeconds(20));
    }
    // Test 1: Login
    @Test(priority = 1)
    public void login() throws InterruptedException {
            driver.FindElement(By.linkText("Sign in"))).click();
        driver.findElement (By.id("userid")))
            .sendKeys("jayanthyelisetti@gmail.com");
        driver.findElement(By.id("signin-continue-btn")).click();
        wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("pass")))
            .sendKeys("Jayanth@2003");
        driver.findElement(By.id("sgnBt")).click();
        Assert.assertTrue(driver.getPageSource().contains("My eBay"), "Login failed");
    }
    // Test 2: Search Product (depends on login)
    @Test(priority = 2, dependsOnMethods = {"login"})
    public void searchProduct() {
          WebElement search=driver.findElement(By.id("gh-ac"));
           search.sendKeys("watch");
           search.sendKeys(Keys.ENTER);
```

```
        Assert.assertTrue(driver.getTitle().contains("laptop"), "Search failed");
    }
    // Test 3: Logout (depends on searchProduct)
    @Test(priority = 3, dependsOnMethods = {"searchProduct"})
    public void logout() {

driver.findElement(By.xpath("//*[@id=\"gh\"]/nav/div[1]/span[1]/div/button/span/span")).click();
                driver.findElement(By.xpath("//*[@id=\"s0-1-4-9-3[0]-0-9-dialog\"]/div/div/ul/li[3]/a")).click();
    }
    @AfterClass
    public void tearDown() {
        driver.quit();
    }
}
```

5.Use Data Provider to supply multiple sets of usernames/passwords to a login test.

Program:

```
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.ss.usermodel.WorkbookFactory;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
```

```
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.DataProvider;
import org.testng.annotations.Test;


public class ExcelTest {
	@DataProvider(name="excelData")
	public Object[][] excelDataProvider() throws IOException{
		String excelPath = "C:\\Users\\jayan\\Desktop\\Java_Selenium\\TestNg_selenium\\src\\test\\resources\\TestData.xlsx";
		String sheetName = "Sheet1";

		FileInputStream fis = new FileInputStream(excelPath);
		Workbook workbook = WorkbookFactory.create(fis);
		Sheet sheet = workbook.getSheet(sheetName);

		int rows = sheet.getPhysicalNumberOfRows();
		int cols = sheet.getRow(0).getLastCellNum();

		Object[][] data = new Object[rows-1][cols];

		for(int i=1;i<rows;i++) {
			Row row = sheet.getRow(i);
			for(int j=0;j<cols;j++) {
				Cell cell = row.getCell(j);
				data[i-1][j] = (cell == null)?"":cell.toString();
			}
		}
		workbook.close();
		fis.close();
```

```
        return data;
    }

    @Test(dataProvider = "excelData")

    public void testLogin(String username, String password) {
        System.out.println("Username: "+username+" | Password: "+password);
        WebDriver driver=new ChromeDriver();
         driver.get("http://zero.webappsecurity.com/login.html");
         WebElement usr=driver.findElement(By.id("user_login"));
         usr.sendKeys(username);
         WebElement psw=driver.findElement(By.id("user_password"));
         psw.sendKeys(password);
         driver.findElement(By.cssSelector("input[name=\"submit\"]")).click();

    }
}
```

6.Run test cases in parallel (methods, classes, tests) using parallel attribute in testng.xml.

1st Program:

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.Test;

public class TestClass1 {
    WebDriver driver;
```

```
	@Test
	public void ebay() throws InterruptedException {
		driver = new ChromeDriver();
		driver.get("https://www.ebay.com/");

		String expectedurl = "https://www.ebay.com/";
		String actualurl = driver.getCurrentUrl();

		Assert.assertEquals(actualurl, expectedurl, "Url validation fail");

		Thread.sleep(3000);
		driver.quit();
	}
}
```

2nd program:

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.Test;

public class TestClass2 {
   WebDriver driver;
   @Test
   public void tira() throws InterruptedException {
      driver = new ChromeDriver();
      driver.get("https://www.tirabeauty.com/");
```

```
        String expectedTitle = "Tira - Buy Makeup & Beauty Products Online | Best Offers on Skincare & Haircare";
        String actualTitle = driver.getTitle();


        Assert.assertEquals(actualTitle, expectedTitle, "Title validation fail");


        Thread.sleep(3000);
        driver.quit();
    }
}
```

XML Document:

```
<suite name="ParallelSuite" parallel="classes" thread-count="1">
    <test name="ParallelTestExecution">
        <classes>
            <class name="Assertions.TestClass1"/>
            <class name="Assertions.TestClass2"/>
        </classes>
    </test>
</suite>
```