TestNG Parameters

Test methods don't have to be parameterless. You can use an arbitrary number of parameters on each of your test method, and you instruct TestNG to pass you the correct parameters with the @Parameters annotation.

There are two ways to set these parameters:
with testng.xml or programmatically.

1 - Parameters from testng.xml

2 - Parameters with DataProviders

Parameters from testng.xml:

Example1:

```
package TestNGDataEx;

import org.testng.annotations.Parameters;
import org.testng.annotations.Test;

public class ParameterizedTest1
{
        @Test
        @Parameters("strUN")
        public void parameterTest(String strUN)
        {
           System.out.println("Parameterized value is : " +
strUN);
        }
}
```

```
<suite name="Suite1">
   <test name="test1">

      <parameter name="strUN" value="qaplanet1"/>

      <classes>
```

```
        <class name="TestNGDataEx.ParameterizedTest1" />
      </classes>

    </test>
  </suite>
```

Example2: How to give multiple parameters from XML

```
package TestNGDataEx;
import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.annotations.Test;
import org.testng.annotations.Parameters;

public class TestngParameters {

      private static WebDriver driver;

  @Test

  @Parameters({ "strURL","sUsername", "sPassword" })

  public void test(String strURL,String sUsername, String
sPassword)
  {

        driver = new FirefoxDriver();

      driver.manage().timeouts().implicitlyWait(10,
TimeUnit.SECONDS);

      driver.get(strURL);
driver.findElement(By.name("txtUserName")).sendKeys(sUsername);


driver.findElement(By.name("txtPassword")).sendKeys(sPassword);

    driver.findElement(By.name("Submit")).click();

    driver.findElement(By.linkText("Logout")).click();

    driver.close();

    driver.quit();

 }
}
---------------------------
```

```
<suite name="Suite">

   <test name="PIM">

      <parameter name="strURL" value="http://classroom:90/qahrm/login.php"/>

      <parameter name="sUsername" value="qaplanet1"/>

      <parameter name="sPassword" value="user1"/>

           <classes>

              <class name="TestNGDataEx.TestngParameters" />

           </classes>

   </test>

</suite>
```

Parametrization with DataProvider:

```
package TestNGDataEx;

import org.testng.annotations.DataProvider;
import org.testng.annotations.Test;

public class DataProviderEx1
{
      // A data provider which will provide single value to a test
method thrice.
   @DataProvider(name="TestDataContainer")
   public Object[] DataProvider_Method()
   {
      // Passing 3 set of data and each set contains single value
      Object[] data=  new Object[3];
      data[0]= "qap1";
      data[1]= "qap2";
      data[2]= "qap3";
       return data;

   }

 // You must need to mention data provider method name in
Test method
      @Test(dataProvider="TestDataContainer")
    public void TestMethod1(String a)
       {
        System.out.println(a);

    }

       // You must need to mention data provider method name
in Test method
      @Test(dataProvider="TestDataContainer")
    public void TestMethod2(String a)
       {
        System.out.println(a);
    }
}
```

```
package TestNGDataEx;
import org.testng.annotations.DataProvider;
import org.testng.annotations.Test;
public class ParamTestWithDataProvider1
{


  @DataProvider(name = "test1")
  public static Object[][] data()
  {
    return new Object[][] {
                   {2, true,"qap1"},
                   {6, false,"qap2"},
                   {19, true,"qap3"},
                   {22, false,"qap4"}
                  };
  }


  @Test(dataProvider = "test1")
  public void test1(Integer inputNumber, Boolean
expectedResult,String sData)
  {
    System.out.println(inputNumber + " " + expectedResult+"
"+sData);
  }

  @Test(dataProvider = "test1")
  public void test2(Integer inputNumber, Boolean
expectedResult,String sData)
  {
    System.out.println(inputNumber + " " + expectedResult+"
"+sData);
  }
}
```