

Daily Push and pull

=====

- git add -A >> add all your files to staging area
- git commit -m "first commit" >> commit to your repository
- git push -u origin master >> Push command
- git pull origin master >> Pull command
- git status >> to know the status of repository

Grouping of the test cases in automation

=====

We use groups in Testng when

- We don't want to define test methods separately in different classes (depending upon functionality) and
- At the same time we want to ignore (not to execute) some test cases as if they do not exist in the code.
- So to carry out this we have to Group them. This is done by using the "include" and "exclude" mechanism supported in testNG.

Changes in the test class



```
1 package com.bankingproject.testclass;
2
3 import org.testng.annotations.Test;
4
5
6
7
8
9
10 public class TC_LoginTest_003 extends BaseClass {
11
12     @Test(groups = { "sanity"})
13
14     public void newCustomerTest() throws InterruptedException {
15         driver.get(baseUrl);
16         test = extent.createTest("Verify application login");
17
18         Loginpage login = new Loginpage(driver);
19         login.setUsername(username);
20         test.log(Status.PASS, "username successfully entered");
21
22         login.setPassword(password);
23         test.log(Status.PASS, "password successfully entered");
24
25         login.loginBtn();
26         test.log(Status.PASS, "clicked on login button");
27
28     }
29
30     @Test(groups = {"regression"})
31
32     public void method2() throws InterruptedException
33     {
34         NewCustomer newcustomer = new NewCustomer(driver);
35
36         newcustomer.selectNewCustomer();
37
38         Thread.sleep(5000);
39         test.log(Status.PASS, "clicked on new customer button");
40
41         newcustomer.selectDOB("12", "03", "1992");
42         test.log(Status.PASS, "DOB entered");
43
44         Thread.sleep(5000);
45     }
46
47 }
48
```

Changes in TestNG.xml file

```
TestNG.xml  ExcelUtils.java  Loginpage.java  TC_LoginTest_00  BaseClass.java  *TC_LoginTest_0  NewCustomer.jav
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd" >
3
4  <suite name="BankingProject">
5      <groups>
6          <run>
7              <exclude name="regression" />
8              <include name="Sanity" />
9          </run>
10     </groups>
11     <test name="BankingTest">
12         <parameter name="browser" value="chrome" />
13         <classes>
14             <class name="com.bankingproject.testclass.TC_LoginTest_003" />
15         </classes>
16     </test>
17 </suite>
```

Use of Excel data in framework

=====

Create new class

```
Second Workspace - BankingProject/src/main/java/com/bankingProject/utilities/Exceldata.java - Eclipse IDE
TestNG.xml  BankingProject/  TC_001_loginTes  BaseClass.java  config.property  ReadConfig.java  Testing1.java  Exceldata.java  2
1  package com.bankingProject.utilities;
2
3  import java.io.File;
4  import java.io.FileInputStream;
5  import java.io.IOException;
6
7  import org.apache.poi.xssf.usermodel.XSSFCell;
8  import org.apache.poi.xssf.usermodel.XSSFRow;
9  import org.apache.poi.xssf.usermodel.XSSFSheet;
10 import org.apache.poi.xssf.usermodel.XSSFWorkbook;
11
12 public class Exceldata {
13
14     public static String getdata(int i, int j) throws IOException
15     {
16         File file = new File("/Users/amitparate/Downloads/Login Data (1).xlsx");
17
18         FileInputStream inputStream = new FileInputStream(file);
19
20         @SuppressWarnings("resource")
21         XSSFWorkbook wb=new XSSFWorkbook(inputStream);
22
23         XSSFSheet sheet=wb.getSheet("Sheet1");
24
25         XSSFRow row2=sheet.getRow(i);
26
27         XSSFCell cell=row2.getCell(j);
28
29         String data= cell.getStringCellValue();
30
31         return data;
32     }
33 }
34
35
```

Changes in test class

```
Program_class.jav  A_class.java  TestNG.xml  BankingProject/po  *TC_001_loginTest.  BaseClass.java  config.properties  ReadConfig.java
12  public void loginTest() throws IOException {
13
14      try {
15          LoginPage login = new LoginPage(driver);
16
17          test = extent.createTest("Verify application login");
18
19          driver.get(baseUrl);
20          test.log(Status.PASS, "URL entered succussfully");
21
22
23
24          login.setUsername(Exceldata.getdata(2,0));
25          test.log(Status.PASS, "username entered succussfully");
26
27          login.setPassword(Exceldata.getdata(2,1));
28          test.log(Status.PASS, "password entered succussfully");
29
30          login.clickOnLoginButton();
31          test.log(Status.PASS, "login button clicked succussfully");
32
33          String expectedTitle = "GTPL Bank Home Page";
34
35          String actualTitle = driver.getTitle();
36
37          if (actualTitle.equals(expectedTitle)) {
38              test.log(Status.PASS, "title verifed and user logged in succussfully");
39          } else {
40              test.log(Status.FAIL, "Login failed");
41              String screenshotpath = BaseClass.captureScreen();
42              test.addScreenCaptureFromPath(screenshotpath);
43          }
44      } catch (Exception e) {
```