**Manual Practical Tests**

1. Write test cases to the following code.

Demo (int a) {

If (a> 5)

a=a\*3

Print (a)}  
  
Test case scenario:

TC1. When input is greater than 5(condition) , verify the value of a -> Branch 1

TC2. When input is les s than or equal to 5(condition) ,verify the value of a ->Branch 2

TC3. When user enters non integer input value, verify the method output -> Negative test

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. Write test cases to the following Screen and HTML file.   
  
Functional test case:

Test scenario:

**NAME:**

<< Boundary value and negative test>>

TC1:when user enter the NAME field less than 3 character and submitting registration form , user should see filed validation error   
TC2:when user enter the NAME field more than 25 character and submitting registration form , user should see filed validation error

TC3: when user enter the special character in name field and submitting registration form, user should see filed validation error

<<Positive testing>>

TC4: when user enter name field with more than 3 and less 25 character and submitting registration along with other input field, user should successfully able to register the form

**PHONE NUMBER:**

<< Boundary value and negative test>>

TC1: when user enter the phone number without 0 or + at starting and submitting registration form, user should see filed validation error   
TC2: when user enter the phone number without 0 or + at starting and less than 8 digit , user should see filed validation error

TC3: when user enter the phone number without 0 or + at starting and more than 13 digit, user should see filed validation error

<<positive testing>>

TC4: when user enter phone number start s 0 between 8 to13 digit and submitting registration along with other input field, user should successfully able to register the form  
TC5: when user enter phone number start s + between 8 to13 digit and submitting registration along with other input field, user should successfully able to register the form

**EMAIL Address:**

<< Boundary value and negative test>>

TC1: when user enter the email without @ and submitting registration form, user should see filed validation error   
TC2: when user enter the email which start with @ or ends with @ , user should see filed validation error

<<positive testing>>

TC4: when user enter proper email and submitting registration along with other input field, user should successfully able to register the form  
  
**UserID:**

<< Boundary value and negative test>>

TC1:when user enter the UserID field less than 5 character and submitting registration form , user should see filed validation error   
TC2:when user enter the UserID field more than 8 character and submitting registration form , user should see filed validation error

<<Positive testing>>

TC3: when user enter the special character or numeric in UserID field and submitting registration form, user should successfully able to register the form

TC4: when user enter UserID with both upper /lower case and submitting registration along with other input field, user should successfully able to register the form

**Password:**

<< Boundary value and negative test>>

TC1:when user try to enter password NOT ((one special character) AND (one uppercase letter) AND (length of password between 4 to 10 character) ) and submitting registration form , user should see filed validation error

<<Positive testing>>

TC3: when user enter minimum ((one special character) AND (one uppercase letter) AND (length of password between 4 to 10 character) and submitting registration form, user should successfully able to register the form

TC4: when user enter password and it should not display in front end ,also it should replace by \*

**End to end testing:**

**SUBMIT**TC1: when user enter the required field and click on submit button, user should successfully able to register the form  
<<>Negative test>>

TC2:when user try to submit without filling requires field ,user should not allowed to go next page and should throw error on screen (all required fields should filled)  
**Cancel**TC1: when user enter some text field or without entry any text field, than try to click cancel , user should successfully able to cancel the registration workflow

**Clear**TC1: when user enter some text field or without entry any text field, than try to click clear, all the text fields should be empty or initial state.

**Database test:**

Verify the registration details in DB, it should correctly populate in database.  
  
**Performance testing** :

Stress test:

Test the Extreme workload to registration application and compare with baseline

**Automation testing** :

Using data driven model, verify the different data set combination

**Security testing :**

Verify the ssl certificate check in regular interval  
  
**sanity check /Exploratory test:**

Schedule script to validate the end to end flow in daily basics both begin & end of today.<its scope of automation>

---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Q3. Define a test plan, test strategy, execution report and defect reports for above questions.   
  
Test Plan:

 Objective:  
To make sure the registration process works in production with zero defect and collect the required information from customer

Test strategy:

* **Functional system testing** will be done by manually and test data will be consider with acceptable boundary values.
* **Regression testing** will cover the BAU functionality working as expected and not impact create by of this change  
    
   Timeline:2MD for both(functional and regression)

1MD for performance /automation in parallel

Execution/Defect report:

* Defect will be tracked on test management tools test case/test result will be updated to track progress and forecast  
    
  **UAT:**FST signoff and test result will be shared to business user and then they will be continue **User acceptance test**
* **For below signoff needs to done before prod deployment**
  + FST signoff owner QA lead
  + UST signoff owner Business user(BA)
  + PERF/regression signoff- automation engineer/performance tester