

Day 9

## TestNG

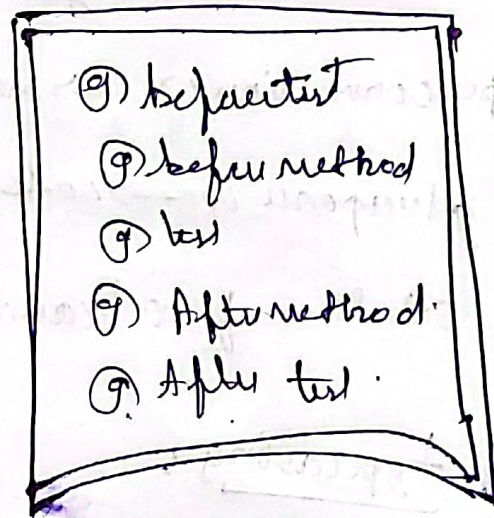
Using framework used for unit / integration / end-to-end, functional testing

- full control over tests. → only for Java.
- Cedric Beust → inspired from JUnit / NUnit
- After JUnit
- open some automated TestNG framework.
- Next generation

### Feature:

① annotations.

- ① before suit
- ① before class.



② Priority setting

③ Invocation count

→ (one test multiple times execute)

→ enable = false → skip

④

→ depend on methods

⑤ By default report generation.

⑥ grouping (Sanity / Regression / smoke)

⑦ Parallel (Execution multiple browser / execute)

⑧ Parameter annotations can read from xml

Help → Eclipse Market place → TestNG → install → uncheck Manager.  
→ confirm → I accept → finish → remember → Always → TestNG select



Install + Add to project

→  
↳ project → configure build path. → class path add library →  
run → next → finish → Apply & close

New Package → New class

(Import to tutorial)

public class tutorial {

@beforeMethod  
public void setup() {

(Before open)

}

@beforeMethod

public void method() {

(void setup)

}

Test mandatory

@test public void test() {

(test 1)

}

@test public void test2() {

(test 2)

}

@AfterMethod

public void tearDown() {

(tearDown)

@AfterTest

(tearDown class)

## ② priority test

① Test (priority = 2)  
public void test1()  
{  
}

② Test (priority = 1)  
public void test2()  
{  
}

## ③ invocation count

① Test (priority = 2, invocation count = 2)  
(2 times Test will Execute)

## skip test:

① Test (priority = 3, enable = false)  
(Test skipped test not Executed)

## Depend on method

① Test (p=2, inv=2, depend on Method = "test")  
(depend on other test)



#### ④ Parameters:

class → xml file → generate

class → Right click → TestNG → connect to testNG →

pat.xml → finish

< test >

< parameter name="a" value="selenium" > </parameter>

< classes > name =

< class > "package name . class name" > </class>

class

① parameter ("a")

② test  
public void main (String v)  
{ sysout (v);

}

#### testNG.xml - file

→ It is the configuration file, Execution information, what are the test case we want to Execute, how we want to Execute them

why?

we cannot go to each of any class and we cannot pick the individual test cases, we need some file where we can mention the list of test case we want to Execute

Project <sup>package</sup> → class → method

Test suite → test cases → test steps.

<xml

>

<suite>

<test name = " " >

<

<classes>

<class> name = " " >

<methods>

<include name = " " />

test suite:

Orange HRM

First class - A, B

Second class - C, D, E

Third class - F, G, H, I, J.

xml →

B, E, H

A, D, I

combine methods