

Mahima Bisht
Empid: 46010069
TEST NG

Date 28/08/19

- testing framework
- unlike Junit it ~~tests~~ runs unit, integration tests, UI tests, functional tests etc.
- open source.
- declare all test methods as public.

→ Assertions

- `assertSame()`
 - check variables pointing to same object.
- `assertEqualsDeep()`
 - checks key value pair in maps of maps or set.
- `assertEqualsNoOrder()`
 - takes arrays and checks contents are equal regardless of order.

(for API tests)

- httpClient dependency
 - allows to send request

HARD ASSERT

- fail and ~~exec~~ terminate the execution when one assert fails.

SOFT ASSERT

- runs all asserts and then collect all the ~~except~~ exceptions

Description field

- add allows to add description about the test.

Syntax :

```
@Test(description="xyz")
public void method() { }
```


Enable

- We can enable/disable test using this field as

ex:

```
@Test(enabled = false)
public void methodName()
{
}
```

Alternative for Enable is @Ignore.

→ Commons-validator (dependency) provides email validator object.

★ In POM.xml we specify the scope as `<scope> test </scope>` so that junit/testing will find tests inside test folder only.

DATA PROVIDERS

we use data providers for parameterization in testing

Syntax: `@Test(dataProvider = "name")`

define data provider as :

```
@DataProvider  
private Object[][] name()  
{  
    return new Object[][]  
    { { "xyz", "xyz" } };  
}
```

GROUPING TESTS

We can group test and also specify dependent test that runs first, after that all tests runs.

Syntax: @Test (groups="sanity")

or @Test (dependsOnGroups
= "sanity")

★ Tests Run in the order
alphabetical order.

→ Set priority to test

Syntax: @Test (priority=1)