1. Testable App.
2. Mockito.
3. **Junit Test**: Noun
   1. Also called **Unit Test.**
   2. JUnit Test Cases.
4. **Unit Testing**: Action
   1. **Unit**: Methods.
   2. **Web App**: Contains all the methods so all the units.
5. **Archetype**:
   1. A way of creating **Sample Maven Project**.
   2. Archetype is a **Maven Project Templating toolkit.**
6. **Naming Convention**:
   1. Case01 🡺 The class to be tested + “Test”
   2. Case02🡺 If for one class, more than one test class, then different naming convention.
7. **Annotations:**
   1. Under **org.junit package**.
      1. **@Test**
         1. **@Test(expected=RuntimeException.class)**
         2. **@Test(timeout=<just\_number\_in\_milliseconds>)**
8. **Methods**:
   1. Under **package org.junit.Assert**
      1. **fail**(“msg”);
      2. **assertEquals**(expected, actual); 🡸 **Overloaded**
      3. **assertFalse(condition);**
      4. **assertTrue(condition);**
      5. **assertArrayEquals(expected, actual)**
   2. Under **package org.junit**
      1. @Before
      2. @After
      3. @BeforeClass
      4. @AfterClass
9. **Runner**
   1. **Parameterized Runner**

**@RunWith(Parameterized.class)**

public class **StringHelperParameterizedTest** {

StringHelper helper = new StringHelper();

private String input;

private String expected;

public **StringHelperParameterizedTest**(String input, String expected) {**//Constructor**

this.input = input;

this.expected = expected;

}

**@Parameters**

public static Collection<String[]> getParameters(){ **//Configuring Parameters**

String expectedOutputs[][]= { {"AACD", "AB"}, {"AEF", "CD"} };

return Arrays.asList(expectedOutputs);

}

**@Test**

public void truncateFirstAInFirstTwoPosition() { **//This test method will be called for each parameter.**

assertEquals(expected, helper.truncateAInFirst2Positions(input));

}

}

* 1. **Suite Runner**

**@RunWith(Suite.class)**

**@SuiteClasses**({ StringHelperParameterizedTest.class, StringHelperTest.class })

public class AllTests {

}



1. **JUnit Guidelines**
   1. Test methods should be **public void** otherwise **initialization error**.
   2. One test method should not have more than one assert statement or condition.
   3. **Method Naming Convention**:  
      **Template: “test”** + **“methodNameToBeTested”** + **“\_ConditionToBeTested”  
      Example:   
      “test”** + **“TruncateAInFirst2Positions()” + “\_AInFirstTwoPositions”  
      testTruncateAInFirst2Positions\_AInFirstTwoPositions()**
2. D

**@RunWith(Parameterized.class)**

public class **StringHelperParameterizedTest** {

StringHelper helper = new StringHelper();

private String input;

private String expected;

public **StringHelperParameterizedTest**(String input, String expected) {**//Constructor**

this.input = input;

this.expected = expected;

}

**@Parameters**

public static Collection<String[]> getParameters(){ **//Configuring Parameters**

String expectedOutputs[][]= { {"AACD", "AB"}, {"AEF", "CD"} };

return Arrays.asList(expectedOutputs);

}

**@Test**

public void truncateFirstAInFirstTwoPosition() { **//This test method will be called for each parameter.**

assertEquals(expected, helper.truncateAInFirst2Positions(input));

}

}