



JUnit Cheat Sheet

Simple Test

```
class UserTests {  
    @Test  
    void isAdult_AgeLessThan18_False() {  
        User paul = new User(17);  
        assertFalse(paul.isAdult());  
    }  
}
```

Basic Assertions

```
assertTrue(condition);  
assertFalse(condition);  
  
assertEquals(expected, actual);           // Equal  
assertNotEquals(expected, actual);        // Not equal  
assertArrayEquals(expected, actual);       // Arrays are equal  
assertSame(expected, actual);              // Same object  
assertNotSame(expected, actual);           // Not the same object
```

Exception Assertions

```
assertThrows(expectedType, executable);  
assertDoesNotThrow(executable);  
  
@Test  
void crash_Normal_ShouldThrow() {  
    Car car = new Car();  
    assertThrows(CrashException.class, () -> car.crash());  
}
```



inovex

Test Fixture

```
class UserTests {  
    @BeforeAll  
    static void initAll() { // runs ones before all tests }  
  
    @BeforeEach  
    void init() { // runs before each test }  
  
    @AfterEach  
    void tearDown() { // runs after each test }  
  
    @AfterAll  
    static void tearDownAll() { // runs ones after all tests }  
  
    User myUser = new User(16);  
  
    @Test  
    void isAdult_AgeLessThan18_False() {  
        assertFalse(myUser.isAdult());  
    }  
}
```

Value Parameterized Tests

```
class UserTests {  
    @ParameterizedTest  
    @ValueSource(ints = { 18, 20, 60 })  
    void isAdult_AgeGreaterThan18_True(int age) {  
        User user = new User(age);  
        assertTrue(user.isAdult());  
    }  
}
```