

1. Introduction in test automation

Gradle settings

```
buildscript {
    repositories {
        jcenter {
            url "http://jcenter.bintray.com/"
        }
    }
}

allprojects {
    repositories {
        jcenter {
            url "http://jcenter.bintray.com/"
        }
    }
}
```

Gradle settings

```
dependencies {  
    testCompile group: 'junit', name: 'junit', version: '4.11'  
    // https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java  
    compile group: 'org.seleniumhq.selenium', name: 'selenium-java', version: '2.53.1'  
    // https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-chrome-driver  
    compile group: 'org.seleniumhq.selenium', name: 'selenium-chrome-driver', version: '3.3.1'  
}
```

Gradle Settings

```
test {
    testLogging {
        // Make sure output from
        // standard out or error is shown
        // in Gradle output.
        showStandardStreams = true
    }
}

tasks.withType(Test) {
    testLogging {
        events 'started', 'passed'
    }
}
```

jUnit Annotation

```
@BeforeClass
public static void createAndStartService() throws IOException {
    service = new ChromeDriverService.Builder()
        .usingDriverExecutable(new File("\\chromedriver.exe"))
        .usingAnyFreePort()
        .build();
    service.start();
}
```

jUnit Annotation

@Before

```
public void setUp() throws Exception {  
    driver = new ChromeDriver(service);  
    driver.manage().timeouts().implicitlyWait(90, TimeUnit.SECONDS);  
    driver.manage().window().maximize();  
}
```

jUnit Annotation

```
@Test
public void openGoogle() throws Exception {
    driver.get("http://google.com.ua");
    driver.findElement(By.id("id")).click();

    driver.findElement(By.xpath(".//*[@id='root']/form/input[2]")).sendKeys("d
adkhhb");
}
```

jUnit Annotation

```
@After
public void closeDriver() throws Exception{
    driver.close();
}
```


jUnit Annotation

```
@AfterClass
public void result() throws Exception{
    System.out.println("Some results");
}
```