**History Of Selenium or what is selenium or selenium v/s QTP**

1. it supports **web or browser based testing** and QTP also does.
2. it does not support desktop applications testing, but QTP does.
3. Selenium is **open source** whereas QTP is licensed.
4. in java, selenium is **API** which is collection of jar files
5. in c#, selenium is API which is collection of dll files
6. in ruby, selenium API which is collection of GEMS
7. Thought works, company, who designed selenium in 2004 by developer Jason Huggins and came up with two components selenium IDE and Selenium RC
8. Main selenium **web site = seleniumhq.org** (hq stands head quarter)
9. there is no need to install selenium, as it is come in the form of dll or jar or gem files, whereas QTP is come in the form of exe file and need to install on system.

**Selenium RC**

1. It is java library
2. It injects java script on our browsers, in order to establish communication with browsers
3. There is only one major class in selenium RC i.e. "Default Selenium"
   1. start method = to launch browser
   2. open method = accepts URL, for navigation
   3. getTitle = to get the title of the web page
   4. click = to click anything link button, submit, radiobutton, checkbox etc

**Selenium IDE**

1. Only good for record and play
2. plug in firefox
3. There is no company, that is working on selenium IDE, to be very honest, companies are only working on web drivers
4. There are some challanges in selenium IDE, because it injects JS in browsers, if browser does not allow JS, then it will become big trouble for us, and library is very small (around collection of max 500 methods)
5. To inject JS in browsers, we have to start the selenium server, which is prerequisite (download selenium = command prompt = type { java jar <name of jar file including extension>)

**Benefits Of Selenium**

1. it has mult lang support like java, c#, ruby, phyton, ruby, perl, php, javascript (QTP has support only VB Script, which in turn such language supports only window operating system, so QTP works only windows operating system)
2. Java is supported in 3 Billion Devices (Got message, when we install JAVA), because java supports in muliple OS like linux, ios, windows, android, solaris etc, due to this reason selenium gets support to run on multiple operating systems.
3. It supports multiple browsers, so that we can run our test cases in IE, firefox, mozilla, safari, chrome, opera, edge browser and so on
4. selenium is come up, with components like (IDE, RC, WebDrivers, Grid)
   1. IDE and RC are part of selenium 1.0 (Deprecated)
   2. WebDrivers are part of selenium 2.0 (Companies are currently working on this) and current version is 2.53.0
   3. GRID 1 is part of selenium 1.0 and GRID 2 is part of selenium 2.0
5. Selenium is known as selenium web driver as selenium RC is deprecated or selenium 2.0 and current version is selenium 2.53.0. Here 2 is the version and 53 is the build number, and 0 is the patch version, as of now no patch version is introduced after version 2.53.0.

**What is WebDriver or Why not Selenium IDE**

1. In this injection of JS in browsers is compeletly removed, so for interaction with browsers, browsers specific drivers classes are introduced like FireFoxDriver, InternetExplorerDriver, ChromeDriver, IPhoneDriver (Web Based Testing, not for native (standalone app) and hybrid {mixture of standalone and web based} application).
2. They have taken required API from browser vendors and integrate their functionality in browser specific driver classes, so that such drivers communiated with browsers without inject javascript in browsers.
3. It is used 25% in web project. It contains those methods, that automate the web sites.
4. Selenium WebDriver API is designed by Simon Stewart, for every browser, he has created seperated class
5. WebDriver is an interface, which is implemented in all browser specific driver classes.
6. Selenium has direct support of firefox, so no need to System.SetProperty, for other browsers, it has to set and download respective broswer driver exe file and pasted them by selecting project

1) for IE, write line System.setProperty("webdriver.ie.driver", "IEDriverServer.exe");

2) for chrome, write line System.setProperty("webdriver.chrome.driver", "chromedriver.exe");

**Grid**

1. Parellel execution of test cases on multiple machine and browsers at the same time, in case of manual testing we allocate resource on different machine with different browser and with different OS as well on each machine
2. It follows HUB and Node sort of architecture, what we do, we will put all test cases on one machine i.e. HUB and when we execute them, then it will be parellel executed on all connected machine i.e. nodes with different browsers and operating systems.

**what is API**

1. I am java dev in abc.com company, got task to develop java application like calculator having functionality add, substract and so on
2. we will design class calculator and define required methods
3. to test any method, will create an object of calculator class and call required methods on it.
4. once u tested, u want to expose such application to outside world or u want that such functionality can be used in other banking applications, we make jar file in java and .dll file in c#, so this jar file or dll file in known as API.
5. in java = functions are known as methods
6. in c# = we use term functions

**Install & Configure JAVA**

1. install jdk1.8 from internet
2. enviornment variable = JAVA\_HOME
3. Path = %JAVA\_HOME%\bin
4. command prompt = verify = java - version

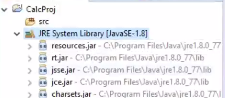
**Editors of JAVA**

1. intellij
2. netbeans
3. webstorm
4. Eclipse IDE = most used, open source code (prerequisite = Have java 1.8 on system)

**Install eclipse**

1. www.eclipse.org
2. download = eclipse IDE for JAVA EE Developers
3. eclipse mars (4.5)
4. once u installed, u will get eclipse.exe, just double click on it to launch eclipse
5. it will ask workspace ( it is the location, where we will be going to maintain project code.)
   1. .metadata folder is created inside workspace folder.

**how to make project in eclipse**

1. launch eclipse
2. right click in project explorer
3. new => project
4. select "java project"
5. next
6. give any project name
7. finish
8. u will find below thing in project explorer
9. 
   1. src = in this folder, we will create some class files.
   2. JRE System Library = this folder, contain, set of jar files.

**how to make new class that contains main method**

1. right click on src folder
2. new
3. class
4. give any name to the class
5. checked = public static void main
6. finish

**how to increase font**

1. window => preferences
2. general => appearance => color and fonts
3. basic => text font => edit

**How to make jar file**

1. -right click on project
2. -export
3. -expand java folder
4. -select "JAR" file
5. -specify the name and location of jar file and click on finish.

**how to include jar file in another project**

1. right click on project
2. build path
3. configure build path
4. librarries
5. add external jars
6. locate ur jar file
7. click on ok

**how to exclude jar file from project**

1. rigth click on jar file
2. build path
3. remove from build path

**how to switch between workspaces**

first create new workspace

file

switch workspace

other

give new name to the workspace

hit enter, u will see eclipse is restarted with new space

now in switch workspace, u will see the name of the workspace, which is already created.

**How To Get The Path Of The Project**

1. -eclipse
2. -select the project
   1. press alt+enter
   2. a window is opened, locate "location" where project path is specified.

**Note**:

1. If we want to convert our code into test cases or testsuite so that we can get to know tht how many test cases have been passed or failed then add testng library into ur project
2. From 3.0 version onwards company has removed the class default selenium, methods which are not required and worked on optimising the methods
3. Ctrl + shift + O = include and exclude used or un-used import statement automatically
4. ctrl + shift + / = multiple line comment
5. ctrl + / = single line comment
6. type syso and then ctrl+space and enter **=** print systsem.out.println statements
7. ctrl + shift + f = to align the code automatically
8. ctrl+f11 = execute project
9. update jar file = first exclude and then include jar file again.