**The complete selenium course:**

**Pom, data driven, hybrid framework, map java with selenium**

1. **Complete Core java**
2. **Selenium Webdriver- Major focus (20 hours+ content)**
3. **Selenium Grid**
4. **Live projects on Selenium**
5. **Interview questions discussion with solutions**
6. **TestNG**
7. **Maven**
8. **ANT**
9. **Jenkins**
10. **Log4j**
11. **Real time challenges in developing framework and how to address them**
12. **Extent Test Execution reports**
13. **Cucumber**
14. **Data driven framework**
15. **Hybrid Framework**
16. **Page object Model Framework**
17. **DataBase testing with Selenium**
18. **Performance testing with Selenium**
19. **Mobile Automation basics only!**
20. **CLoud Automation with Sauce labs**
21. **AutoIT- Desktop Automation integration with Selenium**
22. **Bonus lecture- Bugzilla defect tool management.**

****

**Selenium:**

**Open source automation tool**

**Used for web based application**

**QTP is paid and old one for automation testing**

**Anything based out of browser can be automated**

**It supports multiple browsers**

**Selenium supports different browsers, OS, Multi-language support**

**Selenium is a suite of tools: IDE(Record and playback) and Webdriver(Robust regression testing)**

**Selenium RC is deprecated -> successor is web-driver**

**We also have selenium Grid: Selenium Grid is a smart proxy server that makes it easy to run tests in parallel on multiple machines**

**Web-driver architecture**

**In order to make our browser understand our code/testcases, we need drivers respectively**

**Driver interprets (json) and performs actions on the actual browser**

**Response is sent back to ide to see output**

**We need to mention driver path in client-code for sending json script(Driver acts as server)**

**We have different libraries available for reports, logging and assertions in maven**

**TestNG: Java unit testing framework – Good with assertions and annotations**

**We don’t need java class for writing code once we use testing class**

**ChromeDriver class methods that we can call and use to automate . For this first we need to create an object of that particular class**

**Webdriver is an interface. They have just defined method names but implementations are done by classes.**

**Arraylist is class in java which helps us to overcome the drawbacks of arrays**

**If we make our methods static then that thing will belong to class but not object. We don’t need objects for using our own class methods.**

**Driver: Which makes the browser understand our programming language like python, java, c# etc.**

**We can find line where the error occurred and the exact error from the console screen**

**We can also build our customized xpath and Csslocators using valid format**

**Different ways of constructing xpath:**

1. **Parent – child**
2. **Child – parent**
3. **Sibling(Peer to peer)**

**Xpaths are of 2 types: Absolute(From root) and relative(From middle)**

**Different UI elements: Text fields, buttons, checkboxes, dropdowns, calender format, pictures and text**

**Different locators used in selenium: ID, NAME, CLASS, TAGNAME, CSS SELECTOR, LINK TEXT, PARTIAL LINK TEXT, X PATH.**

**Different dropdowns:**

**Number of checkboxes – Count**

**Assertions using testing library**

**Calender UI – Format selection**

**UI elements state – visible/enabled**

**Handling java alerts – Browser / Window**

**End-end automation flow for vegetables-cart UI**