***NAME :- MALHOTRA DAXIL***

***Module – 5***

**1. What is Automation Testing?**

🡪 Automation testing is a process in which software tests are run automatically using special tools or scripts, instead of being done manually by a person. The goal is to make the testing process faster, more efficient, and more reliable.

* In traditional testing, a human tester performs tests by interacting with the application and checking for errors.
* In automation testing, we write scripts (or use tools) to run these tests on the software. The tools simulate the actions a user would take, like clicking buttons, entering data, and checking results, without a person being involved.

This approach helps:

1. **Save time**: The tests can run quickly and repeatedly, even overnight.
2. **Increase coverage**: More tests can be done in less time, covering more areas of the application.
3. **Improve accuracy**: Since automation avoids human error, the results are more consistent.

**2. Which Are The Browsers Supported By Selenium Ide?**

🡪 Currently, Firefox, Chrome, Edge, IE, and Safari are supported and the Selenium IDE has great documentation on how to configure your environment.

**3. What are the benefits of Automation Testing?**

🡪 Here are the benefits of automation testing in simple terms:

**1. Faster Testing**

* Automation tests are much quicker than manual tests, so you get results faster.

**2. Reuse Tests**

* Once you write a test, you can use it over and over again without doing the same work.

**3. More Testing**

* Automated tests can check more parts of the software in less time, covering more ground.

**4. Consistent Results**

* Since computers do the testing, they never make mistakes. Every test runs the same way.

**5. Saves Money Over Time**

* Even though setting up automation can cost a little at first, it saves money in the long run because it reduces the need for manual work.

**6. Better Use of Human Time**

* People can focus on more important tasks, like testing complex features, while automation handles the repetitive stuff.

**7. Quick Feedback**

* Automated tests give you feedback quickly, which helps find problems faster and fix them.

**8. Tests After Every Change**

* Every time the software changes, automation can check if old features still work. This is called regression testing.

**9. Can Run Anytime**

* Automated tests can run even when you’re not around, like overnight, so testing never stops.

**10. Helps Test Performance**

* Automation can simulate many users at once to see if the software can handle big loads.

In short, automation testing makes the testing process faster, more reliable, and efficient, leading to better software with fewer bugs.

**4. What are the advantages of Selenium?**

🡪 Here are the main **advantages of Selenium** in simple and short terms:

1. **Free & Open Source**: No cost to use.
2. **Cross-Browser Testing**: Works on browsers like Chrome, Firefox, Safari, and more.
3. **Supports Multiple Languages**: You can write tests in Java, Python, C#, and others.
4. **Cross-Platform**: Works on Windows, macOS, and Linux.
5. **Easy Integration**: Can be combined with tools like Jenkins for continuous testing.
6. **Parallel Testing**: Run tests on multiple machines at the same time, speeding things up.
7. **Large Community**: Lots of online resources, tutorials, and support.
8. **Advanced Web Testing**: Automates complex actions like clicking, typing, and more.
9. **Mobile Testing**: Works with tools like Appium to test mobile apps too.

10**. Scalable**: Can handle lots of tests and large applications.

In short, Selenium is fast, flexible, free, and supports many browsers, making it a top choice for automating web testing!

**5. Why testers should opt for Selenium and not QTP?**

🡪 Here’s why testers often choose **Selenium** over **QTP** (UFT) :-

**1. Free vs. Paid**

* **Selenium** is **free** and open-source.
* **QTP/UFT** is **expensive** and requires a paid license.

**2. Multiple Languages**

* **Selenium** supports many programming languages (like Java, Python, etc.).
* **QTP/UFT** uses **VBScript**, which is less flexible.

**3. Cross-Browser Support**

* **Selenium** works on **multiple browsers** (Chrome, Firefox, etc.).
* **QTP/UFT** has limited browser support.

**4. Cross-Platform**

* **Selenium** works on **Windows, macOS, and Linux**.
* **QTP/UFT** works mainly on **Windows**.

**5. Community Support**

* **Selenium** has a **large community** with lots of free resources.
* **QTP/UFT** has **limited support** and is more expensive.

**6. Mobile Testing**

* **Selenium** can be used with **Appium** for **mobile testing**.
* **QTP/UFT** also supports mobile testing but is more costly.

**7. Scalability**

* **Selenium** can easily run tests on **multiple machines** (with Selenium Grid).
* **QTP/UFT** is less flexible for large-scale testing.

**8. CI/CD Integration**

* **Selenium** easily integrates with **CI/CD tools** like Jenkins.
* **QTP/UFT** requires more setup.

**Conclusion:**

**Selenium** is cheaper, more flexible, and works across more platforms and browsers. **QTP/UFT** is more expensive and less flexible.