



|  |  |
| --- | --- |
| **Q.4** | What is known as "a collection of tests that can be executed as a single entity" ? |
| |  |  |  | | --- | --- | --- | | 1. |  | Test cases | | 2. |  | Test scripts | | 3. |  | Test suite | | 4. |  | Test plans | | |
|  |  |

|  |  |
| --- | --- |
| **Q.5** | As of Selenium IDE v1.9.1, test cases can be exported into which of the following formats ? |
| |  |  |  | | --- | --- | --- | | 1. |  | .cs | | 2. |  | .java | | 3. |  | .py | | 4. |  | All | | |
|  |  |

|  |  |
| --- | --- |
| **Q.6** | In Selenium IDE, user CANNOT open more than one test case at a time[True/False]. |
| |  |  |  | | --- | --- | --- | | 1. |  | FALSE | | 2. |  | TRUE | | 3. |  | NA | | 4. |  | NA | | |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Q.7** | The entire test script creation process using Selenium IDE can be classified into three steps. Identify the correct sequence. | |
| |  |  |  | | --- | --- | --- | | 1. |  | Saving-->Recording-->Playing Back | | 2. |  | Recording-->Playing Back-->Saving | | 3. |  | Saving-->Playing Back-->Recording | | 4. |  | Recording-->Playing Back | | | |
| **Q.8** | Select the appropriate context/scenario in which Selenium Web Driver is used ? |
| |  |  |  | | --- | --- | --- | | 1. |  | Multi-browser testing including improved functionality | | 2. |  | AJAX-based UI elements | | 3. |  | Simple page navigation | | 4. |  | All | | |

|  |  |
| --- | --- |
| **Q.9** | Web Driver’s goal is to provide an API that establishes : |
| |  |  |  | | --- | --- | --- | | 1. |  | Improved consistency between browsers | | 2. |  | Additional functionality addressing testing problems not well-supported in Selenium 1.0 | | 3. |  | A well-designed standard programming interface for web-app testing | | 4. |  | All | | |
|  |  |

|  |  |
| --- | --- |
| **Q.10** | Complete the declaration for the FireFoxDriver implementation class : public class FirefoxDriver extends ? implements ? |
| |  |  |  | | --- | --- | --- | | 1. |  | RemoteWebDriver, Killable | | 2. |  | RemoteWebDriver, SearchContext | | 3. |  | FindsByCssSelector, OperaDriver | | 4. |  | ChromeDriver, Killable | | |
|  |  |

|  |  |
| --- | --- |
| **Q.11** | Which WebDriver interface method returns a String, representing the current URL that the browser is looking at ? |
| |  |  |  | | --- | --- | --- | | 1. |  | getCurrentUrl() | | 2. |  | getWindowHandle() | | 3. |  | getPageSource() | | 4. |  | get() | | |
|  |  |
| |  |  | | --- | --- | | **Q.12** | Handling special keyboard and mouse events are done using which of the advanced User Interactions API class and interface ? | | |  |  |  | | --- | --- | --- | | 1. |  | MouseActions | | 2. |  | Actions | | 3. |  | Action | | 4. |  | Events | | |  |  |  | | --- | --- | | **Q.13** | In Web Driver, which method allows to navigate to a given URL ? | | |  |  |  | | --- | --- | --- | | 1. |  | goToUrl("url") | | 2. |  | navigate.to("url") | | 3. |  | getUrl("url") | | 4. |  | get("url") | | | |  |  |  |  |  | | --- | --- | | **Q.14** | Which is the correct code snippet to find an element that looks like : <iframe src="..."></iframe> | | |  |  |  | | --- | --- | --- | | 1. |  | WebElement frame = driver.findElement(tagName("iframe")); | | 2. |  | WebElement frame = driver.findElement(By.tagName("iframe")); | | 3. |  | WebElement frame = driver.findElementById(tagName("iframe")); | | 4. |  | WebElement frame = driver.findElement(By.tagName(iframe)); | | | |  |  |  |  |  | | --- | --- | | **Q.15** | Which Actions class method generates a composite action containing all actions so far executed, ready to be performed ? | | |  |  |  | | --- | --- | --- | | 1. |  | builds() | | 2. |  | action() | | 3. |  | build() | | 4. |  | perform() | | | |  |  |  |  |  | | --- | --- | | **Q.16** | Accessing links using their exact link text is done through which method ? | | |  |  |  | | --- | --- | --- | | 1. |  | By.partialLinkText() | | 2. |  | By.name() | | 3. |  | By.linkText() | | 4. |  | By.text() | | |  |  |  | | --- | --- | | **Q.17** | What are the two ways to generate a report with TestNG ? | | |  |  |  | | --- | --- | --- | | 1. |  | Listeners | | 2. |  | Reporters | | 3. |  | JUnit | | 4. |  | HTML | | |  |  |  | | --- | --- | | **Q.18** | TestNG is designed to cover which different categorie(s) of tests ? | | |  |  |  | | --- | --- | --- | | 1. |  | Unit Testing | | 2. |  | Functional Testing | | 3. |  | Integration Testing | | 4. |  | All | | | |  |  |  |  |  | | --- | --- | | **Q.19** | TestNG supports dependent test methods, parallel testing, load testing, and partial failure [ True/False ]. | | |  |  |  | | --- | --- | --- | | 1. |  | TRUE | | 2. |  | FALSE | | 3. |  | NA | | 4. |  | NA | | | |  |  |  |  |  | | --- | --- | | **Q.20** | If a test script is composed of more than one test method, the execution priority and sequence can be set using which TestNG annotation and by setting which parameter ? | | |  |  |  | | --- | --- | --- | | 1. |  | @TestSuite, priority | | 2. |  | @TestCase, Maxpriority | | 3. |  | @Tests, Maximum | | 4. |  | @Test, priority | | | |  |