The **static keyword** in java is **used for memory management mainly**.

The static can be:

variable

method

block

nested class

**this keyword**

this is a **reference variable that refers to the current object**.

The **super keyword** in java is a reference variable that is used to **refer immediate parent class object**.

**static binding**

When type of the object is determined at **compiled time**(by the compiler), it is known as static binding.

**Dynamic binding**

When type of the object is determined at **run-time**, it is known as dynamic binding.

<http://www.airtel.in/personal/mobile/prepaid/easy-recharge-enter-number>

Random popup

<http://learn-automation.com/handle-alert-in-selenium-webdriver/>

uiautomatorviewer.bat

**Java Abstract** Classes. A Java abstract class is a class which cannot be instantiated, meaning you cannot create new instances of an abstract class. The purpose of an abstract class is to function as a base for subclasses.

What is Java Serialization? Primary purpose of java serialization is to write an object into a stream, so that it can be transported through a network and that object can be rebuilt again. When there are two different parties involved, you need a protocol to rebuild the exact same object again. Java serialization API just provides you that. Other ways you can leverage the feature of serialization is, you can use it to perform a deep copy.

Username: sjeyaram

password: $@rvin

[\\hp-1\common\testing](file:///\\hp-1\common\testing) team\Automation\JAR FILES

C:\Users\sjeyaram\.hudson\jobs\Optus\_Reports\_ANT\workspace

inetmgr

Exception - An exception is a problem sathat arises during the execution of a program. Exceptions are caught by handlers positioned along the thread's method invocation stack.

CheckedException - An exception is a problem that arises during the execution of a program. Exceptions are caught by handlers positioned along the thread's method invocation stack.

RuntimeEception - It is an exception that occurs that probably could have been avoided by the programmer. As opposed to checked exceptions, runtime exceptions are ignored at the time of compliation.

<http://cag.vspl.net/vpn>

adb shell "svc wifi enable"

The **final keyword** in java is used to restrict the user.

A **final class** cannot be extended by any other class

A **final variable** cannot be reassigned to another value

A **final method** cannot be overridden

0 12 \* \* 7

**=TRIM(LEFT(A1,FIND("=",A1)-1))**

<http://appium.io/slate/en/v1.0.0/?ruby#introduction-to-appium>

<http://learn-automation.com/handle-untrusted-certificate-selenium/>

=[DataEngine.xlsx]Fault!B3

if class have entity reference is called as **Aggregation**

hasNext()- "**Returns true if the iteration has more elements**" So you can check if there are more objects you can get from the iterator, and if there are it returns true. If there are no more objects it will return false.

next() - "**Returns the next element in the iteration**" So if there is a next item, it will be returned. If there isn't it'll throw a 'NoSuchElementException', but if you use iterators properly you should never encounter it.

**Constructor** in java is a special type of method that is used to initialize the object.

If a class have multiple methods by same name but different parameters, it is known as **Method Overloading**.

If subclass (child class) has the same method as declared in the parent class, it is known as **method overriding** in java.

**@BeforeSuite:** The annotated method will be run **before all tests in this suite** have run.

**@AfterSuite:** The annotated method will be run **after all tests in this suite** have run.

**@BeforeTest:** The annotated method **will be run before any test method belonging to the classes** inside the <test> tag is run.

**@AfterTest:** The annotated method **will be run after all the test methods belonging to the classes** inside the <test> tag have run.

**@BeforeGroups:** The list of groups that this configuration method will run before. **This method is guaranteed to run shortly before the first test method that belongs to any of these groups is invoked.**

**@AfterGroups:** The list of groups that this configuration method will run after. **This method is guaranteed to run shortly after the last test method that belongs to any of these groups is invoked.**

**@BeforeClass:** The annotated method **will be run before the first test method in the current class is invoked.**

**@AfterClass:** The annotated method **will be run after all the test methods in the current class have been run.**

**@BeforeMethod:** The annotated method **will be run before each test method.**

**@AfterMethod:** The annotated method **will be run after each test method**.

**Constructor** in java is a special type of method that is used to **initialize the object.**

**Abstraction** is a process of hiding the implementation details and showing only functionality to the user.

A class that is declared with **abstract** keyword, is known as **abstract class** in java. It can have abstract and non-abstract methods (method with body).

**sTimeStamp:**

Calendar cal = Calendar.*getInstance*();

Date time = cal.getTime();

String timestamp = time.toString();

System.*out*.println(timestamp);

String systime = timestamp.replace(":", "-");

System.*out*.println(systime);

**Download particular folder:**

String profilePath = "ProfileName";

FirefoxProfile profile = new ProfilesIni().getProfile(profilePath);

profile.setPreference("browser.download.folderList",2);

profile.setPreference("browser.download.manager.showWhenStarting",false);

profile.setPreference("browser.download.dir","/Users/test/Downloads/WebDriver/");

profile.setPreference("browser.helperApps.neverAsk.saveToDisk","text/csv");

DesiredCapabilities dc = DesiredCapabilities.firefox();

dc.setCapability(FirefoxDriver.PROFILE, profile);

**AutoIT**

Runtime.getRuntime().exec("D:\AutoIt\AutoItTest.exe");

**Highlight the element**

JavascriptExecutor js = (JavascriptExecutor) driver;

js.executeScript(

"arguments[0].setAttribute('style', arguments[1]);",

element, "color: red; border: 3px solid red;");

js.executeScript(

"arguments[0].setAttribute('style', arguments[1]);",

element, "");

Map<String, Object> prefs = new HashMap<String, Object>();

prefs.put("download.default\_directory", "C:\\New\_Download");

DesiredCapabilities caps = DesiredCapabilities.chrome();

ChromeOptions options = new ChromeOptions();

options.setExperimentalOption("prefs", prefs);

caps.setCapability(ChromeOptions.CAPABILITY, options);

WebDriver driver = new ChromeDriver(caps);

driver.get("http://www.google.com");

|  |  |  |
| --- | --- | --- |
| **Selenium IDE** | **Selenium RC** | **Selenium WebDriver** |
| It only works in Mozilla browser. | It supports with all browsers like Firefox, IE, Chrome, Safari, Opera etc. | It supports with all browsers like Firefox, IE, Chrome, Safari, Opera etc. |
| It supports Record and playback | It doesn’t supports Record and playback | It doesn’t supports Record and playback |
| Doesn’t required to start server before executing the test script. | Required to start server before executing the test script. | Doesn’t required to start server before executing the test script. |
| It is a GUI Plug-in | It is standalone java program which allow you to run Html test suites. | It actual core API which has binding in a range of languages. |
| Core engine is Javascript based | Core engine is Javascript based | Interacts natively with browser application |
| Very simple to use as it is record & playback. | It is easy and small API | As compared to RC, it is bit complex and large API. |
| It is not object oriented | API’s are less Object oriented | API’s are entirely Object oriented |
| It doesn’t supports of moving mouse cursors. | It doesn’t supports of moving mouse cursors. | It supports of moving mouse cursors. |
| Need to append full xpath with ‘xpath=\\’ syntax | Need to append full xpath with ‘xpath=\\’ syntax | No need to append full xpath with ‘xpath=\\’ syntax |
| It does not supports listeners | It does not supports listeners | It supports the implementation of listeners |
| It does not support to test iphone/Android applications | It does not support to test iphone/Android applications | It support to test iphone/Android applications |

The class loader is responsible for locating libraries, reading their contents, and loading the classes contained within the libraries.

public boolean waitForPageLoad(int waitTimeInSec, ExpectedCondition<Boolean>... conditions) {

boolean isLoaded = false;

Wait<WebDriver> wait = new FluentWait<>(driver)

.withTimeout(waitTimeInSec, TimeUnit.SECONDS)

.ignoring(StaleElementReferenceException.class)

.pollingEvery(2, TimeUnit.SECONDS);

for (ExpectedCondition<Boolean> condition : conditions) {

isLoaded = wait.until(condition);

if (isLoaded == false) {

//Stop checking on first condition returning false.

break;

}

}

return isLoaded;

}

when We are calling the run method by the reference variable of Parent class. Since it refers to the subclass object and subclass method overrides the Parent class method, subclass method is invoked at runtime.

Since method invocation is determined by the JVM not compiler, it is known as runtime polymorphism.

An interface is a **collection of abstract methods** and also **blueprint of a class**.

An interface is not a class. Writing an interface is similar to writing a class, but they are two different concepts. A class describes the **attributes and behaviors** of an object. An interface contains **behaviors** that a class implements.

allure - powerfully and mysteriously attractive or fascinating; seductive.

mysteriously - Arousing wonder or curiosity, especially by being difficult to explain or understand

fascinating - extremely interesting.

seductive -tempting and attractive; enticing

sultry - (of the air or weather) hot and humid.

slinky - (of a garment) fitting closely to the lines of the body

passionate - having, showing, or caused by strong feelings or beliefs.

raunchy - energetically earthy and sexually explicit.

steamy - producing, filled with, or clouded with steam.

coquettish - behaving in such a way as to suggest a playful sexual attraction; flirtatious.

amorous - showing, feeling, or relating to sexual desire.

flirtatious - behaving in such a way as to suggest a playful sexual attraction to someone.

provocative - causing anger or another strong reaction, especially deliberately.

deliberately - consciously and intentionally; on purpose.

come-hither - (of a woman or her manner) playfully showing or expressing sexual interest

agnostic - a person who believes that nothing is known or can be known of the existence or nature of God.