| Fundamentals What is a Java class - Unix |
| --- |

| References |
| --- |

[Supporting Material](https://drive.google.com/drive/u/1/folders/0BzKlcv-uKOM5blV4YWw0QTFrVms)

| **Important words** |
| --- |

1. You should have created the following folder named as per the git lesson **/fundamentals/1\_whatisaclass/**

* Open the command prompt/terminal
* Change directory to this folder using the **cd** command.

1. According to <https://en.wiktionary.org/wiki/class> a class is A group, collection, category or set of things sharing characteristics or attributes.

Examples of using the word class are:

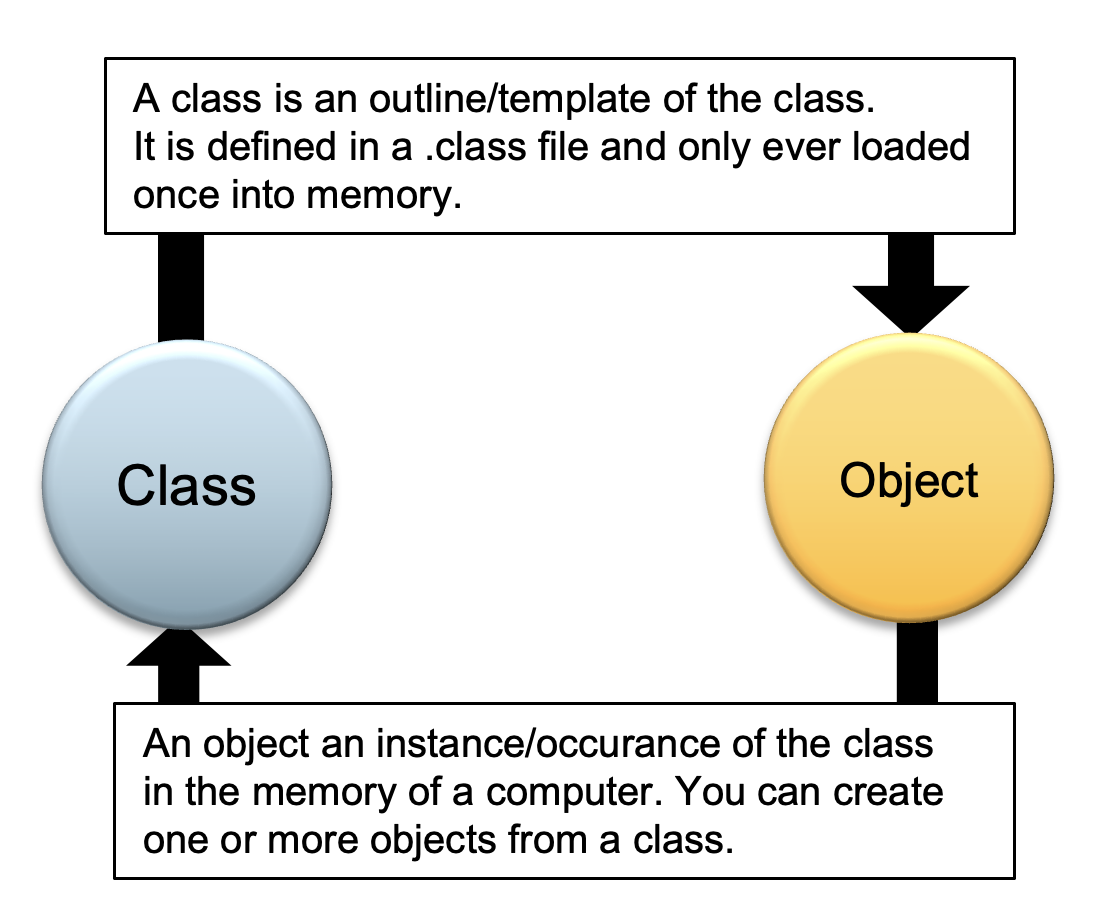
* Ford Fiesta is in a class of small class
* American swimmers were in their own class these olympics
* The main characteristic of this class is that the kids are all naughty.

**P** provide 5 examples of the use of the word class were it refers to a collection or grouping

of things.

| **1.** | **He teaches an english class** |
| --- | --- |
| **2.** | **There are different classes for each subject** |
| **3.** | **Duane Vermeulen is in a class of his own when it comes to the breakdown** |
| **4.** | **That fire is a class A fire** |
| **5.** | **The class of 2024 was well educated** |

1. Ds



The above definition of a class and an object as used in Java leads to an infinite loop as you can see.

A learner ends up not understanding what is either a that or an object because the are intertwined and each comes up in the definition of the other.

For the fundamentals section of this course we will focus on writing classes and less working with objects. In the OOP section we will do a deep dive into objects and clarify some of the details.

For now you will mostly create a class with one main method or more methods that are related. Hence the related methods will be grouped/placed in a single class. A class can hold class and instance/object variables but that is out of scope for the fundamentals section.

Complete the sentences below to show you understanding of the use of classes

| **1.** | My class is not compiling because my method is incorrect |
| --- | --- |
| **2.** | I have two methods in my class that cause conflict with each other. |
| **3.** | The methods in my class are named class methods |
| **4.** | My main methods are lacking information |
| **5.** | This class is missing an object |

* Below is an example of an empty class without a main method.

| public class FirstClass { } |
| --- |

Save the above file into a file called FirstClass.java

Into the folder /fundamentals/1\_whatisaclass/

Confirm that the file is there by running the “**ls -lart” or dir**  command from the fundamentals directory.

Insert screenshot of the ls or dir command:

| **Insert a screenshot of the program running below.** |
| --- |
|  |

1. Compile the above class using the following command. **javac  FirstClass.java**

* This command creates a FirstClass.class file which is not human readable.
* Confirm that the file is there by running the **“ls -lart”** or **dir** command from the fundamentals directory.
* Insert screenshot of the ls or dir command:

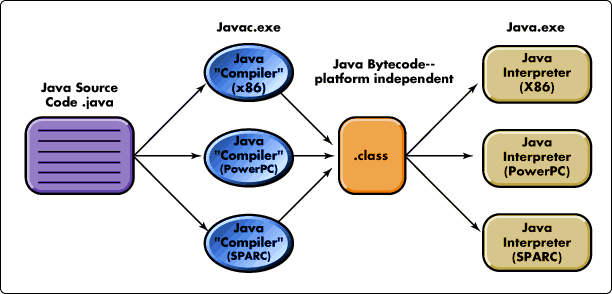
| **Insert a screenshot:** |
| --- |
|  |

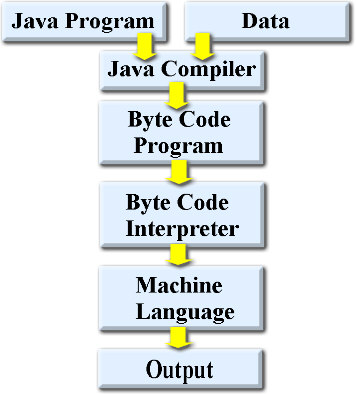
1. Errors may come up in connection with the above file. Check for the following common mistakes.
   * Java is [case-sensitive](http://en.wikipedia.org/wiki/Case_sensitive) that means First Class differs from Firstclass to the compiler.    Therefore make sure that the filename and the contents thereof are as dictated in this tutorial.

* Some editors attach the .txt extension to a file if you select the text documents type. If the file you created was saved as First Class.java.text it will not compile. Removing the .text in the file name will solve the problem.
* If the above step was executed successfully then the javac command must produce a file called FirstClass.class
* Open the FirstClass.class file in the text editor paste the contents of the file here:

| **FirstClass.class content:** |
| --- |
|  |

* The format of the FirstClass.class file is byte code. It is intended to run on all computer platforms without requiring recompilation.





1. There are a couple of steps that are involved in the life cycle of a java program. Following links can help with your understanding of the concept discussed in this chapter.

<http://java.sun.com/docs/books/tutorial/getStarted/cupojava/win32.html#win32-2c>

<http://mathbits.com/MathBits/Java/Introduction/mechanics.htm>

Using the links above provide a definition for the phrase "byte codes".

| **Answer:** |  |
| --- | --- |