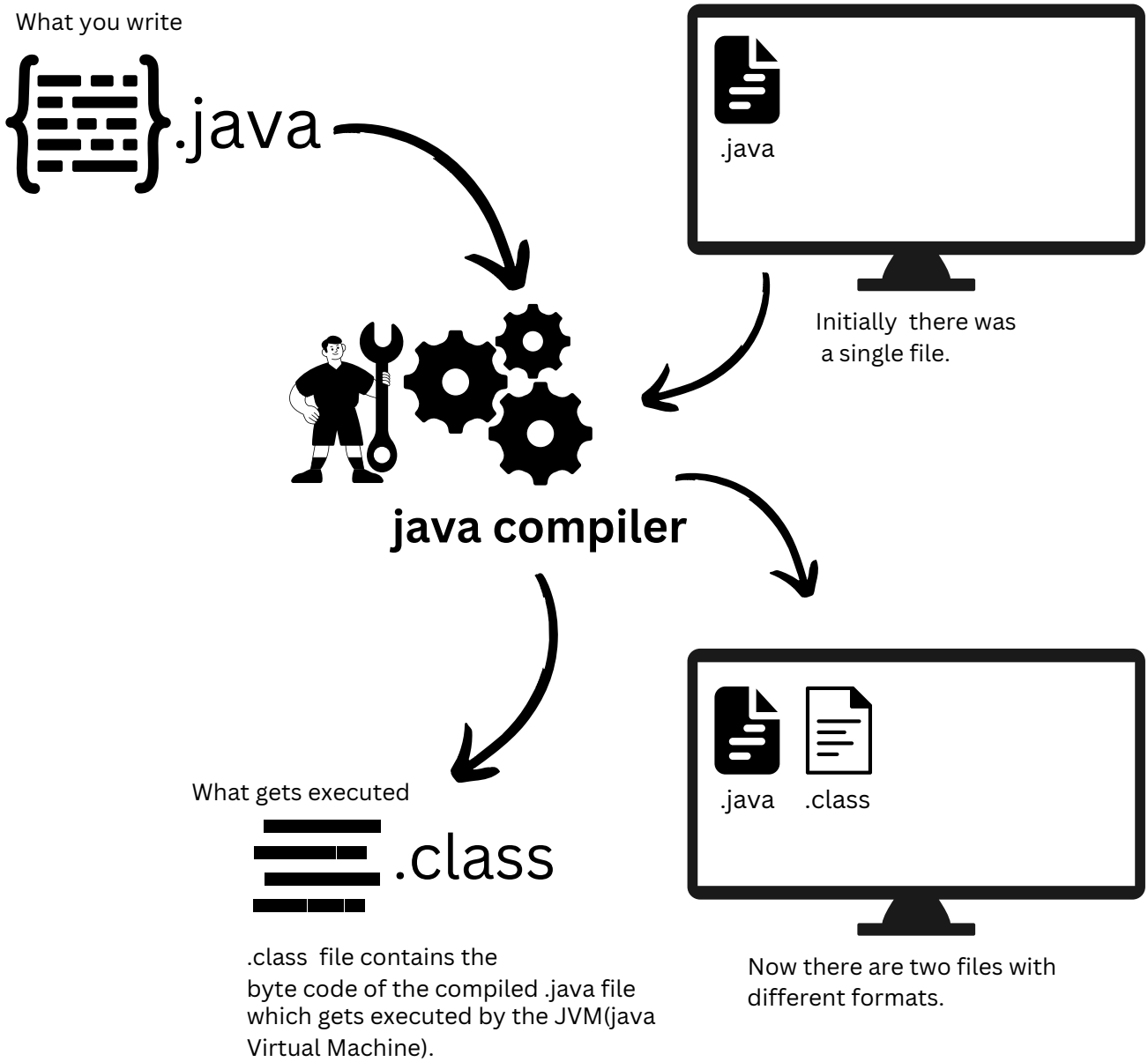
Java Class

#javatheeasyway

Learning java made easy.

/chauhansumitdev

Must Know



JVM??

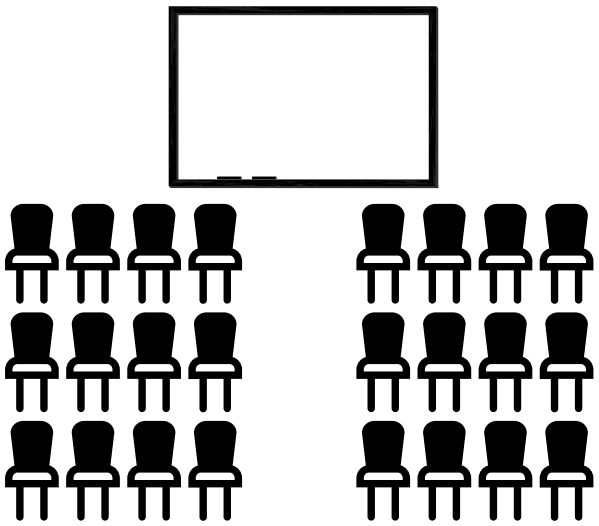
Java Virtual Machine is a virtual machine that **provides runtime environment** for java programs i.e **makes a system run java programs (byte code).**

What is a Class?

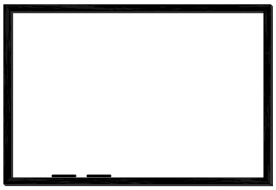
..is probably a class.

A java class is a **template** from which **objects** can be **created.** In simple words it is an **object factory**.

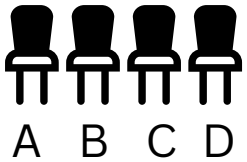
Example:



Consider a classroom with many chairs, and a white-board....
the things written on the board are mere text-scripts but what a student grasps and experiments out from the learnings differs from student to student in the same classroom.
So here the white-board acts as a java class and every student a different object.



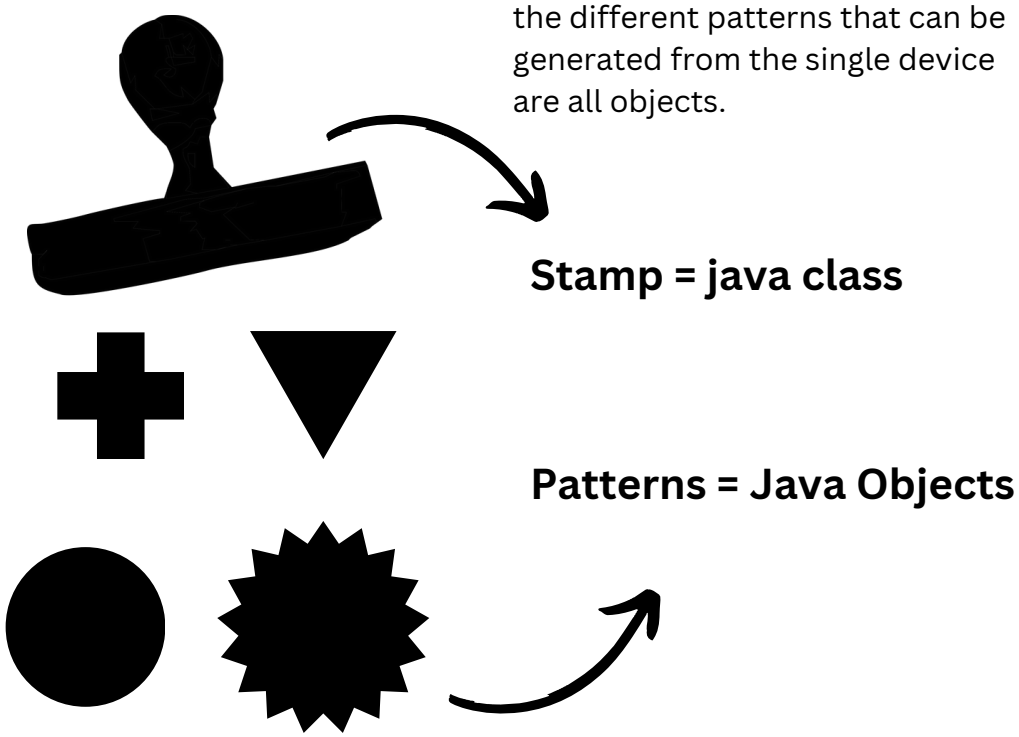
java class



java objects
(more on objects in further pages)

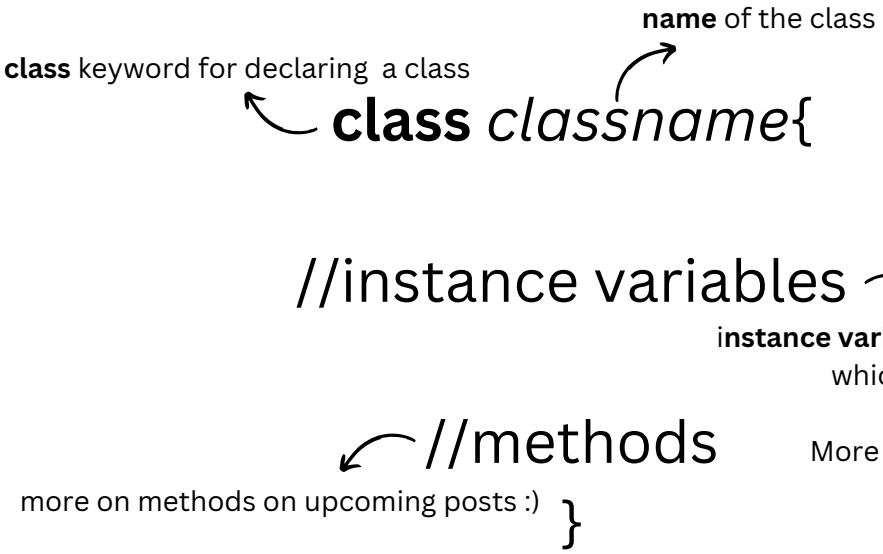
Another Example:

Let's consider another example, a stamp is just a one piece material capable of producing any stamp designs according to the pattern inserted.
So,
The stamp acts as a java class and the different patterns that can be generated from the single device are all objects.



A class in java does not exists or has no practical significance as being a template for objects, what actually gets allocated in the memory are the objects it creates.

Syntax Of Class In Java:



Why is there need for instance variables??
As seen in above examples, for every new object created, new set of values need to be stored. This is achieved by instance variables.

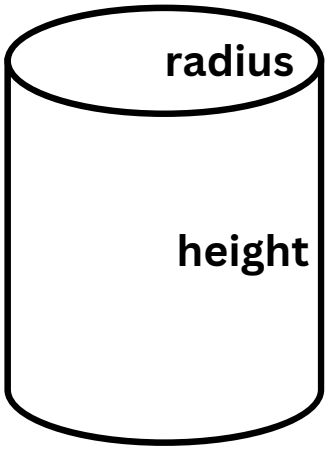
The **methods** and the **instance variables** are called the members of the class.

Practical Application

Let's Code A Class:

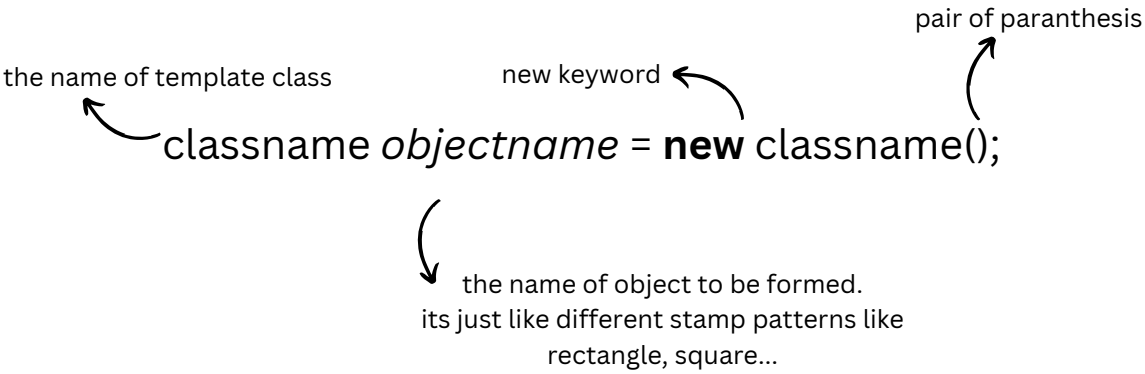
Make a Cylinder Class taking its radius and height as instance variables.

```
class Cylinder{

double radius;
double height;

}
```



This class is just a template like the stamp discussed before, it has no significance unless its used to make objects out of it.

For making objects we use the following syntax:



new keyword **dynamically allocates memory** at **runtime** of the program to create object i.e when the code gets executed, the required memory is allocated during the execution of the program. Therefore dynamic allocation allows the object to take as much space as required so that the memory management can be taken care of.

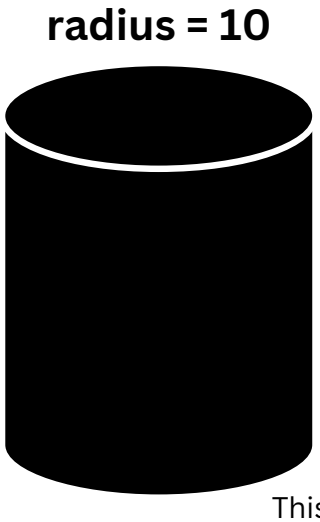
Creating object for Cylinder class:

Cylinder A = new Cylinder();

As described earlier, every object has its own copy of instance variable.
So, to access the instance variables we use the (.) dot operator.

A.radius = 10;
A.height= 100;

These variables were present in the class, but now they store some data, occupy some memory.



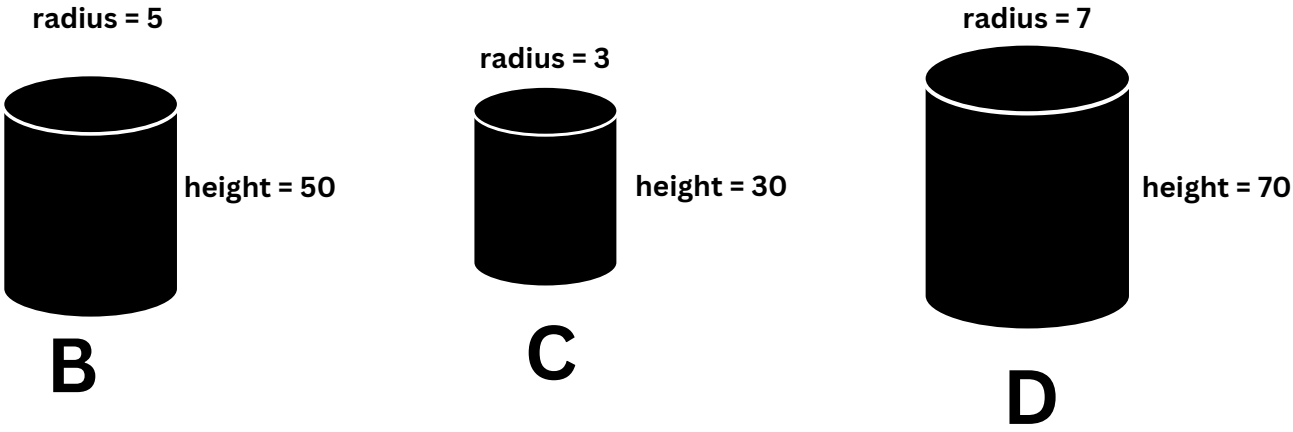
This object created is similar to a pattern created by the stamp.

Now this object actually occupies some space in memory.
By this similar process many objects of Cylinder class can be created.

Cylinder B = new Cylinder();
B.radius = 5;
B.height= 50;

Cylinder C = new Cylinder();
C.radius = 3;
C.height= 30;

Cylinder D = new Cylinder();
D.radius = 7;
D.height= 70;



Up Next:

- **Methods In Java**
- **Working more on the Cylinder class using methods.**

/chauhansumitdev