**Classes and object an overview**

A class is a user defined data types that consists of properties and functions

The functions inside classes are called methods

The properties are everything other than methods, like arrays, variables initialized to certain values etc

**A class is an template or blue print for an object**

**An object is an instance of class**

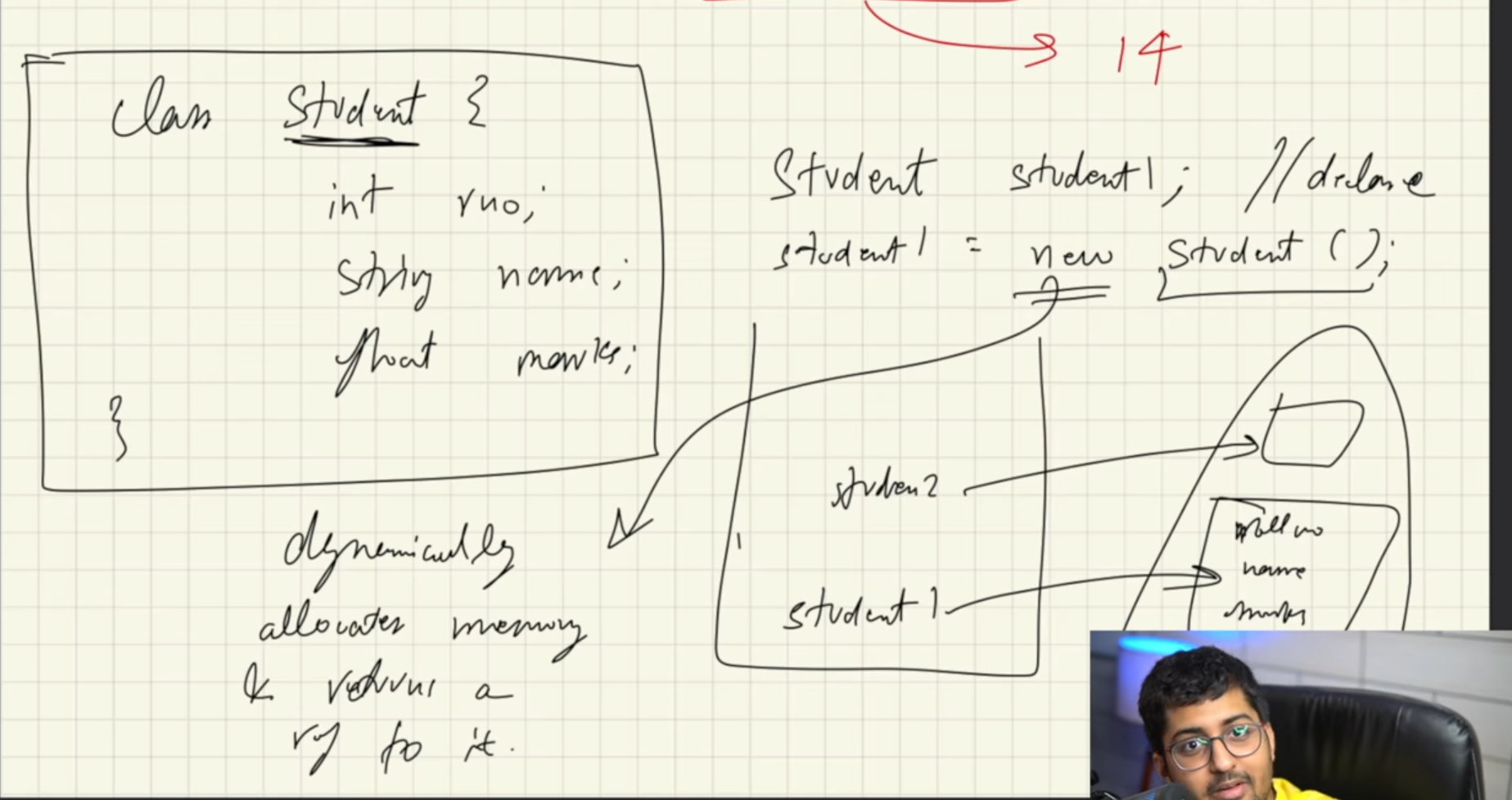
A class is just a logical construct whereas an object is an physical reality, object is something that is actually occupying memory

**Certain Rules**

instant variables : Variables inside the class

Instant variables must be declared inside the class but outside the methods

* Suru maa Class banaunae
* Class bhitra chiyaeko instance variables lai declare garnae
* declare garaeko variables lai constructors bata initialize garnae
* depending upon the need of problem aaru methods haru banaudai Janae



here when we do

**Student student1; // reference variable declaration**

then a reference variable of type student1 is created in the stack during the compile time

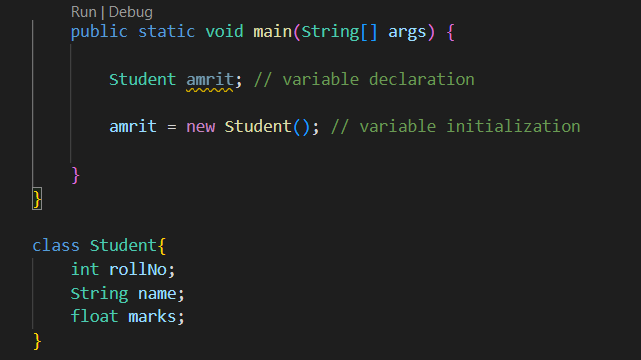
And when we do

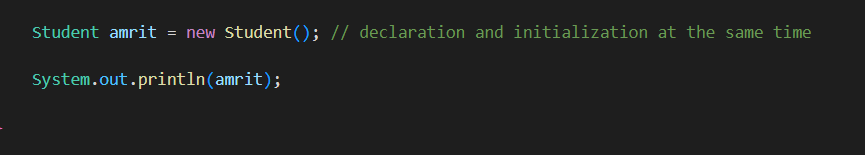
**student1 = new Student( ) ; // variable initialization of object creation**

then the object is created in the heap memory **dynamically** at run time and the reference of that object is returned to the reference variable

So the reference variable living in the stack points to the object living in the heap

Dynamic memory allocation means the memory is allocated during the run time, when the program starts to run



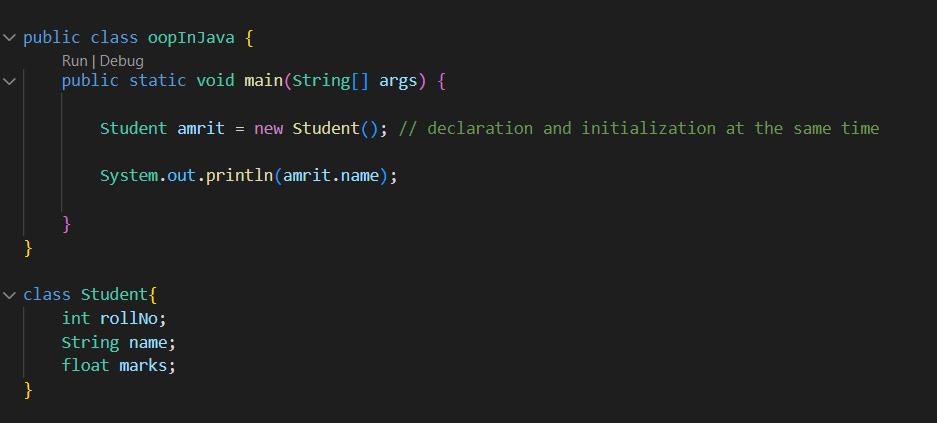


**Constructor**

A constructor is an special method inside the class, having the same name as class, and it is called by Java ( maybe JVM ) itself the moment we create the object

It is used to initialize the variables created inside the class

Default Constructor



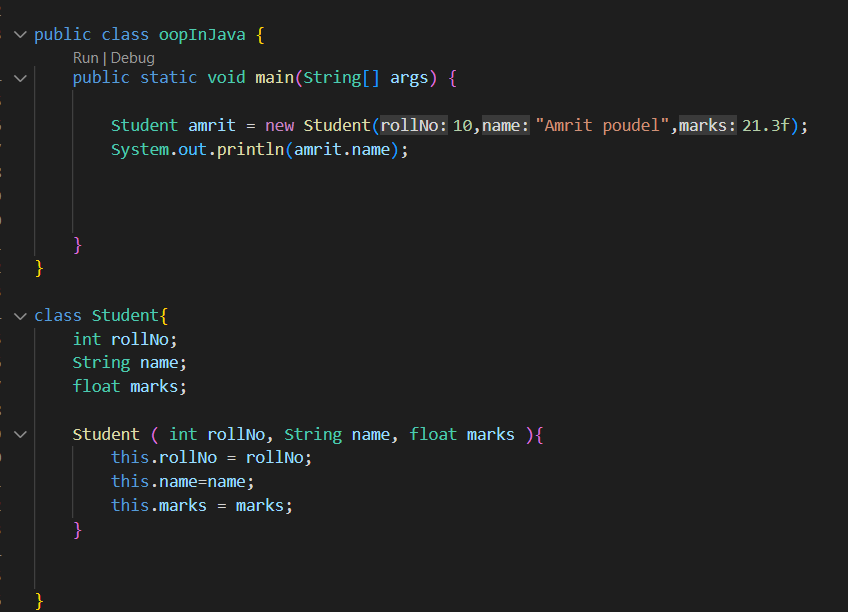
Here we do not see any constructor inside the Student class, but the object amrit is being created

So how come this is possible ?

For the object formation a constructor must be called

Here a default is constructor is automaticly / implicitely being called

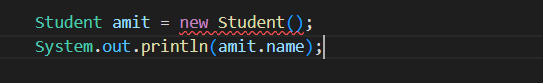
It is called by default by java when we do not have any constructor inside the class

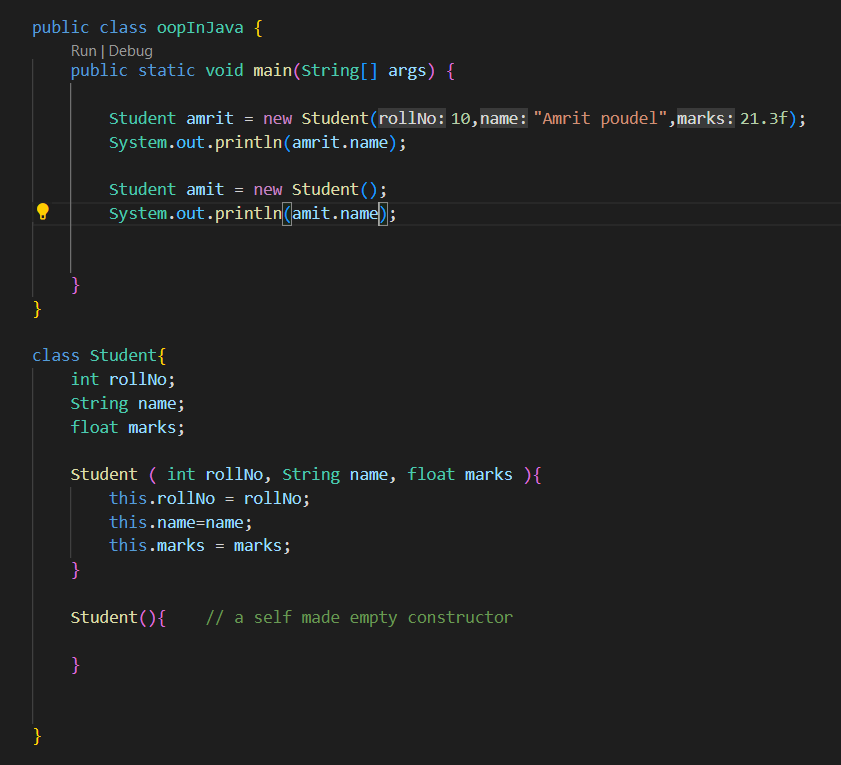


This is how we pass argument to the constructor and construct an object

But

Now we won’t be able to create an object without any values

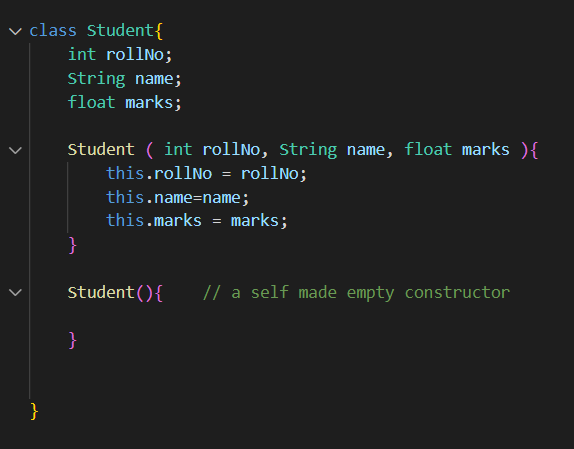




Now on doing this both objects “amrit” and “ amit “ are formed

Looking closely something we can call constructor overloading is happening here

A deeper look in to the constructor



Here a constructor has the same name as the Class

But Student class is a user defined data type

So actually in case of constructor Student seems to be the return type of the function that does not have any name

So

**A constructor is the function that has the same name as the class**

**or**

**A constructor is a nameless function inside class or method that has the same return type as the class name**

jbkjv