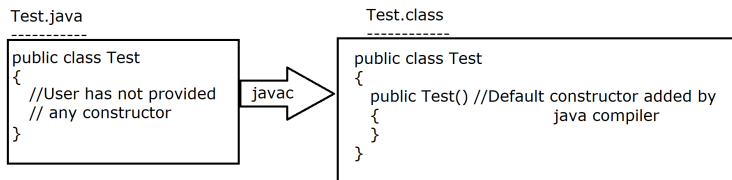


Default constructor added by compiler :

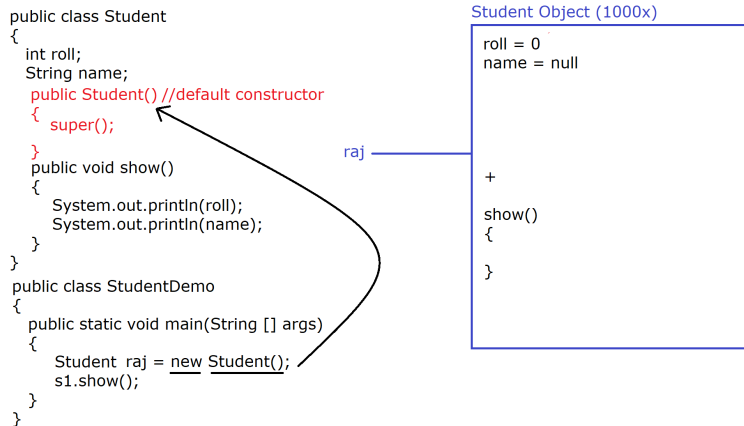
* In java, whenever we write a class and if we don't write any type of constructor in that particular class then automatically javac will add one default constructor to the class.



* Every java class must contain at-least one constructor either explicitly written by developer OR implicitly added by java compiler.

* The access modifier of default constructor depends upon class access modifier that means if class access modifier is public then the default constructor added by java compiler would also be public, on the other hand if class is non public then default constructor access modifier is also non public.

Why default constructor is added by java compiler :



* new keyword is responsible to allocate the memory for all the non static members (non static field + non static method), now with the support of **java compiler**, new keyword is responsible to initialize all the non static field with defaults, This process is known as Instantation process.

* At the time of object creation by using new keyword, we depend upon **constructor** so, we can say without constructor support object creation is not possible in java. default constructor is provided by compiler to support the Object creation in java.

* As we know, new keyword is responsible to initialize the non static field with the support of java compiler but the default values provided by new keyword is not useful for the developer hence by writing user-defined constructor, a developer can modify those default values with user-defined values.

default values are :

Data type	-	Default value
byte		0
short		0
int		0
long		0
float		0.0
double		0.0
char		'\u0000' [space]
boolean		false
String		null
Integer		null
any ref. var		null