SNSCOLLEGEOFTECHNOLOGY

(ANAUTONOMOUSINSTITUTION)

COIMBATORE–35

DEPARTMENTOFCOMPUTERSIENCEAND ENGINEERING

UNITII-CONTROLSTATEMENTSANDCONSTRUCTORS

ConstructorsinJava

InJava,aconstructorisa blockofcodessimilarto themethod. Itiscalled whenaninstanceoftheclassiscreated.Atthetimeofcallingconstructor,memoryfortheobjectisalloca tedinthe memory.

Itisaspecialtypeof methodwhich isusedto initializethe object.

Everytimeanobjectis createdusingthenew()keyword, atleastone constructoris called.

Itcallsadefaultconstructorifthereis noconstructoravailableintheclass.Insuchcase,Java compiler provides a default constructor by default.

Therearetwotypes ofconstructorsinJava: no-argconstructor,andparameterized constructor.

Note: Itiscalledconstructorbecauseitconstructsthevaluesatthetimeofobjectcreation.Itis not necessary to write a constructor for a class. It is because java compiler creates a default constructor if your class doesn't have any.

RulesforcreatingJavaconstructor

Therearetworulesdefinedfortheconstructor.

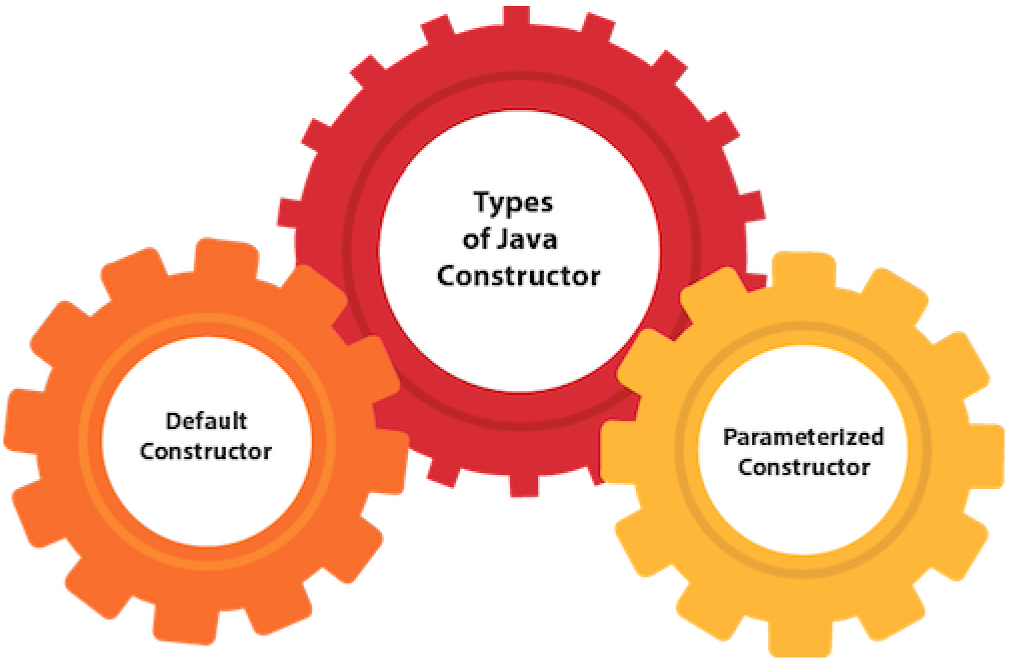
1. Constructornamemustbethesameasitsclassname
2. AConstructormust havenoexplicitreturn type
3. AJavaconstructorcannotbeabstract,static,final,andsynchronized

Note: We can use access modifierswhile declaring a constructor. It controls the object creation.Inotherwords,wecanhaveprivate,protected,publicordefaultconstructorin Java.

Types ofJavaconstructors

Therearetwotypes ofconstructors in Java:

1. Defaultconstructor(no-argconstructor)
2. Parameterizedconstructor

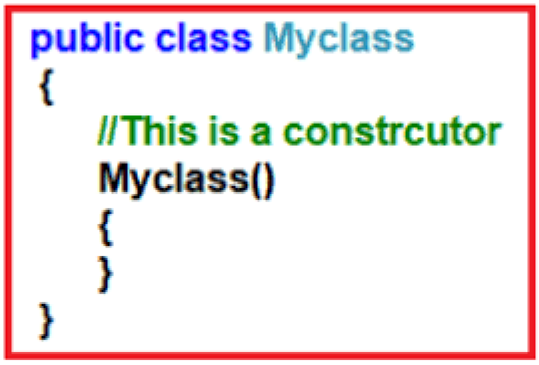


Java DefaultConstructor

Aconstructoriscalled"DefaultConstructor"whenitdoesn'thaveanyparameter.

Syntaxofdefaultconstructor:

1. <class_name>(){}



```
public class Myclass
{
    //This is a constrcutor
    Myclass()
    {
    }
}
```

Example of defaultconstructor

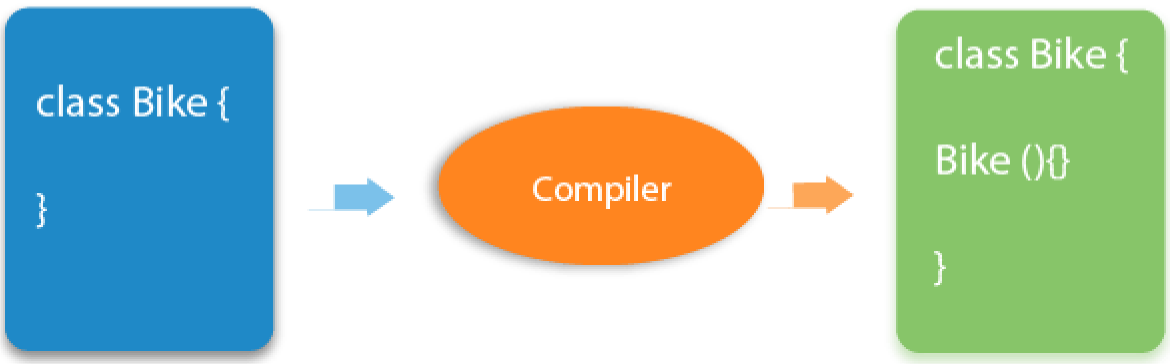
Inthisexample,wearecreatingtheno-argconstructorintheBikeclass.Itwillbeinvokedatthe time of object creation.

```
classBike1

{

        Bike1()//creatingadefaultconstructor

        {

        System.out.println("Bikeiscreated");

        }

        publicstaticvoidmain(Stringargs[])

        {

        Bike1b=new Bike1();     //callingadefaultconstructor

        }

}
```

Output:

```
Bikeiscreated
```

Rule:Ifthereisnoconstructorinaclass,compilerautomaticallycreatesadefault constructor.



Q)Whatisthepurposeofadefault constructor?

Thedefaultconstructorisusedtoprovidethedefaultvaluestotheobjectlike0,null,etc., depending on the type.

Exampleofdefaultconstructorthatdisplaysthedefaultvalues

```
classStudent3
{
intid;
Stringname;
        void display()
        {
        System.out.println(id+""+name);
        }
         publicstaticvoidmain(Stringargs[])
        {
        Student3s1=newStudent3();
        Student3s2=newStudent3();
        s1.display();
        s2.display();
        }
}
```

Output:

```
0null
0null
```

Explanation: Intheaboveclass,youarenotcreatinganyconstructorsocompilerprovidesyoua default constructor. Here 0 and null values are provided by default constructor.

JavaParameterizedConstructor

Aconstructorwhichhasaspecificnumber ofparameters is calledaparameterized constructor.

Whyusetheparameterizedconstructor?

Theparameterizedconstructorisusedtoprovidedifferentvaluestodistinctobjects. However, you can provide the same values also.

Exampleofparameterizedconstructor

Inthisexample,wehavecreatedtheconstructorofStudentclassthathavetwoparameters.We can have any number of parameters in the constructor.

```
classStudent4
{
intid;
Stringname;
            //creatingaparameterizedconstructor
            Student4(int i,String n)
           {
            id = i;
            name=n;
            }
            //methodtodisplaythevalues void
            display()
         {
         System.out.println(id+""+name);
         }

            publicstaticvoidmain(Stringargs[])
           {
            //creating objects and passing values
            Student4s1=newStudent4(111,"Karan");
            Student4s2=new Student4(222,"Aryan");
            //callingmethodtodisplaythevaluesofobject s1.display();
            s2.display();
           }
}
```

Output:

```
111Karan
222Aryan
```

Constructor Overloadingin Java

InJava,aconstructorisjustlikeamethodbutwithoutreturntype.Itcanalsobeoverloaded like Java methods.

Constructor overloading in Javais a technique of having more than one constructor withdifferentparameterlists.Theyarearrangedin awaythateachconstructorperformsadifferent task. They are differentiated by the compiler by the number of parameters in the list and their types.

ExampleofConstructorOverloading

```
//Javaprogram tooverloadconstructors
classStudent5
{
    intid;
    String name;
    intage;
    //creatingtwoarg constructor
    Student5(int i,Stringn)
   {
    id =i;
    name=n;
    }
    //creatingthreeargconstructor
    Student5(int i,Stringn,inta)
   {
    id =i;
    name=n;
    age=a;
    }
    voiddisplay(){System.out.println(id+""+name+""+age);}

    publicstatic void main(String args[])
{
    Student5s1 =newStudent5(111,"Karan");
    Student5s2 =newStudent5(222,"Aryan",25);
    s1.display();
    s2.display();
   }
}
```

TestitNow

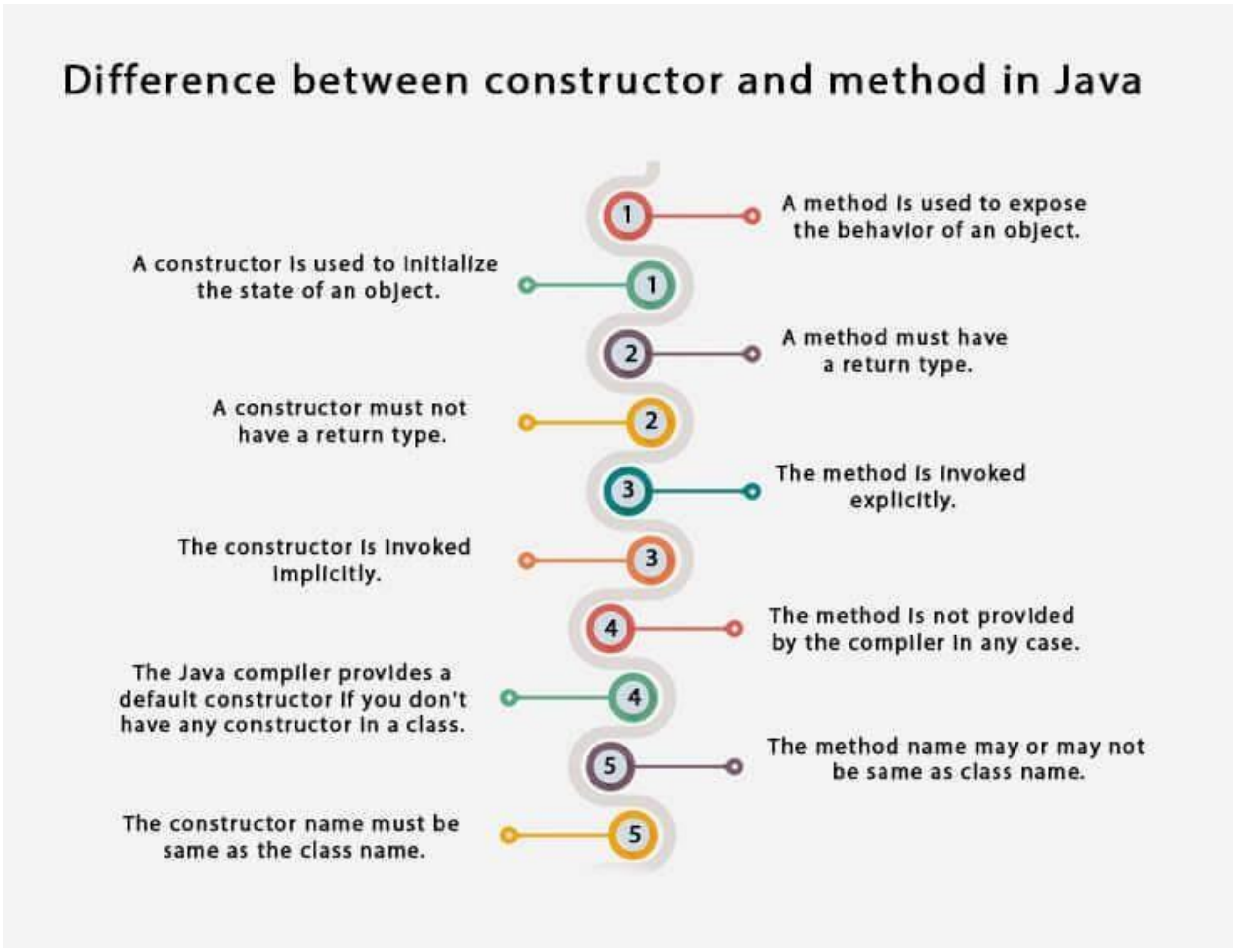
Output:

```
111Karan0
222Aryan 25
```

DifferencebetweenconstructorandmethodinJava

Therearemanydifferencesbetweenconstructorsand methods.Theyare given below.

Java Constructor	Java Method
Aconstructorisusedtoinitializethestateofan object.	Amethodisusedtoexposethebehaviorofanobject.
Aconstructormustnothaveareturn type.	A method musthaveareturn type.
Theconstructor isinvoked implicitly.	Themethod is invoked explicitly.
The Java compilerprovides a defaultconstructorifyoudon'thaveanyconstructor in a class.	Themethodisnotprovidedbythecompilerin any case.
Theconstructornamemustbesameasthe class name.	Themethodnamemayormaynotbesameas the class name.


Difference between constructor and method in Java
1
A method is used to expose the behavior of an object.
A constructor is used to initialize the state of an object.
1
2
A method must have a return type.
A constructor must not have a return type.
2
3
The method is invoked explicitly.
The constructor is invoked implicitly.
3
4
The method is not provided by the compiler in any case.
The Java compiler provides a default constructor if you don't have any constructor in a class.
4
5
The method name may or may not be same as class name.
The constructor name must be same as the class name.
5

JavaCopyConstructor

ThereisnocopyconstructorinJava.However,wecan copythevaluesfromoneobjectto another like copy constructor in C++.

There aremanyways tocopythevalues ofoneobjectinto anotherin Java. Theyare:

- Byconstructor
- Byassigningthe values of oneobject into another
- Byclone()method ofObjectclass

Inthisexample,wearegoingtocopythevaluesofoneobjectintoanotherusingJava constructor.

```
//Javaprogram toinitializethevaluesfromoneobjecttoanotherobject.
classStudent6
{
    intid;
    String name;

    //constructortoinitializeinteger and string
    Student6(inti,Stringn)
   {
    id =i;
    name=n;
    }

    //constructortoinitializeanotherobject
    Student6(Student6 s)
   {
    id =s.id;
    name =s.name;
    }
    voiddisplay(){System.out.println(id+""+name);}

    publicstatic void main(String args[]){
    Student6s1 =newStudent6(111,"Karan");
    Student6s2 =newStudent6(s1);
    s1.display();
    s2.display();
   }
}
```

TestitNow

Output:

```
111Karan
111Karan
```