POLYMORPHISM

Method Overloading

If a class has multiple methods of same name but different in parameters it is known as method overloading

Advantage:- it increases readability of program

2 ways to overload the method

:- by changing the no of arguments

:- by changing the data type of arguments

Method Overriding

If a subclass has the same method as declared in parent class, it is known as method overriding

Adv:- it is used for runtime polymorphism

Static method can not be overridden

It is because static method is bound with the class whereas instance method is bound with an object. Static belongs to class area whereas object belongs to heap area.

Method overloading is an example of compile time polymorphism

Whereas method overriding is an example of runtime polymorphism

Super keyword is used to refer immediate parent class instance variable.

It can be used to envoke parent class method.

Super() can be used to invoke parent class constructor

Instance Initializer Block is used to initialize the instance data member. It runs each time when the object is created.

First parent class constructor is invoked

Then instance initializer block is invoked and then

Current class constructor is invoked

Final keyword is used to restrict the user.

It can be variable, method and class.

Example we set speedlimit to final so that it never changes

If a method is final it can’t be overridden

If a class is final then it can’t be extended

We can only initialize blank final variable in constructor

Class Vehicle{

Final int speed;

Vehicle(int s){

Speed = s;} // final blank variable is being initialised and now it cant be changed

}

A static final variable that is not initialized at the time of declaration is called static blank final variable

Class A{

Static final x ;

static{

x = 10; // possible as initialisation is in static block

}

If we declare any parameter as final we cannot change it.

Int cube(final int n){

N = n+2; // error not allowed as n is final

}

Constructor can never be declared final

Because it is never inherited

Polymorphism is a way by which we can perform a single action in different ways.

Poly means “Many”

Morphs means “Form”

Connecting a method call to method body is called binding

Static binding

Dynamic binding

When the type of object is determined at compile time it is called static binding

If there is any private final static methods in class there is static binding

When the object type is determined at run time it is called dynamic binding