What is class?

A class is user defined blueprint or prototype which is used to create an object.

Class is a logical entity or say its not a real world entity or class is not physical .

Real world example:- Animal ,Birds Vehical, fruits etc.

What is object?

Object is an instance of class.

Object is physical entity or object is real world entity.

OOP’s Concept:-

It is a Object oriented Programming language.

OOP is the programming paradigm based on the concept of objects which contains the data (fields or variables and methods.

It is the most popular programming paradigm used by the programmers.

There are 4 types of features:-

1.Inheritance :-

It is a IS-A Relationship . It is also known as “Inheritance”.

Is-A Relationship or Inheritance is achieved by using “extends”keyword.

Use of relationship between java classes.

1. code reusability

2. less execution time

3. less memory usage.

Super class property access to the child class is called inheritance .

There are 5 types of Inheritance

1.Single Inheritance

2.Multilevel Inheritance

3.Hierarchical Inheritance

4.Multiple Inheritance

5.Hybrid Inheritance.

And Multiple Inheritance is not possible in class but possible in interface.

Polymorphism:-

Poly means many morphism means form.

Advantages:-It provides the flexibility to devlop an application i.e.it allows us to perform a single task by different ways.

Types of polymorphism:-

1.Compile time polymorphism(Method overloading)

2.Runtime polymorphism(Method overriding)

1.Compile time polymorphism:-

It is also known as static polymorphism or early binding.

If the polymorphism is achieved at compile time then it is compile time polymorphism.

Compile time polymorphism can be achieved by “method overloading”.

The process of compiler trying to resolve the method call based on reference type is known as method overloading.

There are 5 rules of method overloading:-

1.same class

2.same method name

3.Diff parameter

4.Diff sequence

5.Is not use in IS-A relationship

2.Runtime Polymorphism:-

It is also known as Dynamic polymorphism or late binding.

If the polymorphism is existed at runtime then it is known as Runtime polymorphism.

Runtime polymorphism can be achieved by “method overriding”.

The process of JVM trying to resolve the method call based on reference type is known as method overriding.

Overriding is the feature by which child class trying to change the implementation of parent class method.

There are 5 rules of method overriding.

1.Diff class

2.same method name

3.Same parameter

4.same sequence

5.Is- A relationship is followed.

Abstraction:-

Abstraction is hiding the details(hiding the implementation part) and just highlight the main services.

Real world example:-Car internal working of breaks,gears etc.

It is achieved by abstract class & Interfaces.

Abstract class:-

Abstract class are those which can contain both concreate methods and abstract methods.

Abstract methods:-

Abstract methods are those whose implementation part is hided.

Abstract methods are those which does not have body or implementation part.

Encapsulation:-

Encapsulation is the process by which variables and methods are wrapped or binding into a single unit.

Encapsulation is technically hiding the data from other classes and these data can be accessed only through the member functions of its own class.

Real world example:-capsule,mobile

Technically every java class is an example of encapsulation.

Encapsulation is achieved by declaring variables as private and public getter and setter methods.

Encapsulation = data hiding + abstraction

Advantages of encapsulation:-

1.Data hiding

2.Increase Flexibility

3.Reusability

Keywords:-

This:- This keyword is “reference variable” that refer to the current object.

Use of this keyword:-

1. this class is use to current class instance variable.

2. this keyword is used to invoke the current

3.this keyword is used to invoke the current class constructor.

Super:-

The super keyword is a reference variable which is used to refer immediate parent class object.

Use of Super keyword:-

1.Super keyword can be used to refer the immediate parent class instance variable.

2.Super keyword is can be used to invoke parent class method.

3.Super keyword is used to invoke parent class constructor.

-> If we don’t provide Super() in constructor the compiler provides it implicitly.

->Super() should always be the first statement in constructor.

Static:-

Static is keyword it is non-access modifier. Use of static keyword.

1.It is used to improve Share-ability.

2.It is used for memory mgmt.

Static keyword can be used with

1.variable

2.block

3.method

4.nested class and inner class (not outer class)

1.Static Variable:\_If we declare any variable as static it is known as static variable.

Static variable gets memory allocated in method area at the time of class loading.

2.Static block:- A static block created using static keyword is known as static block.

Static block is executed at the time of class loading.

3.Static Method:- These are also loaded at time of class loading

In case of method overloading we can overload static methods.

But,we cannot override static methods.

AccessModifier:

public: it is access over all the world.

protected: within the package and out side the package child class.

private: only class.

default: within the package and subclasses.