Abstract method & class; A method without body is called abstra abstract returntype name (---); If a class containing any abstract method then the class is said to be abstract class and it should be declared as abstract. abstract class name abetract returntype name (=),

```
class AbDemo
      abstract
& void noke )
  2 System. out-pointen (" I
abstract void lilly ();
public class AbDemoEr entends AbDemo
  void lilly ()
  System out pointly (4 I am Illy"),"
  void flowers?
     System. out. println (" I am flower");
  public étatic void main (string[] args)
   AbDemoEx a = AbDemoEx();
   a. stoke(); a. lilly(); a. flower();
```

Differences between concrete and abbrard 10, significations concrete extension to It doesnot contain any methods. abstract methods Sub- day, must gress Sub-class may of may not 2. aborters translate, evereide superilass inethole super das. It can be instantiated; 31. It correct bo between that when i Similarities: () pulls blev 1. Both are classes. The many 2. Both contains concrete methods 3. Both are extended by wing extends! 4. We can't extend more than one abstract don't as well as concrete class. I've multiple inheritance is not possible in java through dayses roddA wan da , z is Interface; interface name abstract method * It contains only abstract methods. It shows functionalities and hides implementation. * It is a complete future framework. sarchark It is implemented by wing keywords implements. abeteract methods * Subclass must implement of interface

```
* Interface cannot be instantiated.
public interface Varun
    abstract void coke();
abstract void pepsic);
public class Vanunc endands varun
   public void coke()
   systemioutiprintln (" coke"),
   public void pepsi ()
   System, out, println (a peppi");
    public void sting ()
    System out println (" storing");
    public stataile void main (String[] args)
    d varunce v = new Varunce ();
      v. coke(); v. pepsic); v. steing();
* we can implement any number of interfaces i've multiple inheritance injura
   is possible in java through interface.
```

```
Public interface varun
abstract void coke();
public interface Shiva
abetract void pepsic);
public class varunc implements varun, shira
 public void cokel)
   System.out. println (4 coke");
   public void pepsi()
  of system, out, println (" pepsi"),
   public void sting()
   eystem.out.println (" sting"),
  public static void main (string[] args)
               v = new Varun(食),
       v. coke(); v, pepin(); v, sting();
```

between concrete class and interface. Differences Interface concrete class 1. It contains only 1. It does not contain any abstract methods. abstract methods. 2, Sub-class must implement 2. Sub-class may or may implemented interface. not override superclass methods. 3. Wes can implement 3. we cannot extend more with thany no of than one class. Multiple inheritance is not possible interfaces i.e Mul Inhem is possible in Java through in java through classes interfaces. 4. public abstract snot 4. public abstract is default from methods. default for methods public static final y 5. public static final 13 default for members. not default for members. 6. It can be instantiated 6. It cannot be instantiated. Interface abstract class It contains only abstract 1. It contains concrete as well as methods. abstract methods. 2. We cannot extend more than one 2. we can implement any no. of abstract class, in Multi. Inhe is interfaces. IR M. I is possible not possible through abstract classes in java through interfaces.

3. public abstract is not default 3. public abstract is default

for methods. for methods 4. public static final is not 4. public static final 1s sol default for members. default for members.

Similarities blw abstract class & Interface 1. Both contains abstract methods.

e Both cannot be instantiated. 3. sub class must overside abstract methols & declare subclass as abstract, com

4. Both supports dynamic binding.

```
type catting
                                object
 Primary
implicit explicit cycarling down carring
                                    explicit
                        implicit
                                    down carring
                        uparting
 public interface Add
                            public interface
       void sub();
public dass sreya implements Add
    public void sub();
      System. out, println (" sneeja class");
 public class siit implements Add
public void sub();
   system.out.println (4 Silt clas");
  public static void main (string[] ags)
      That Add a !
      a = new Sreeja ();
      a. mb();
     a = new slit ();
```