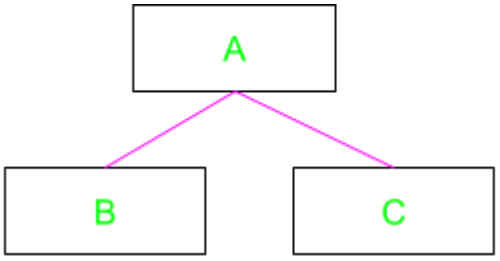
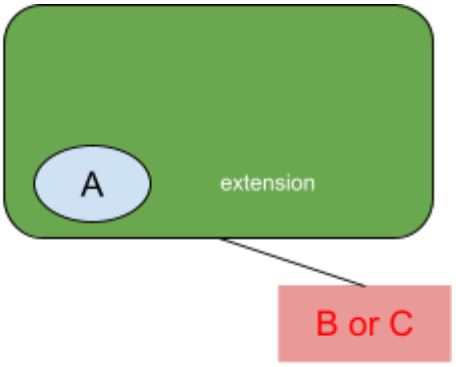
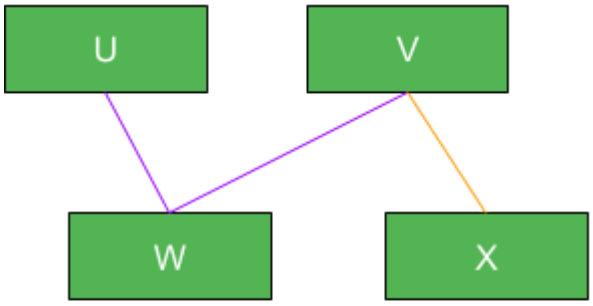
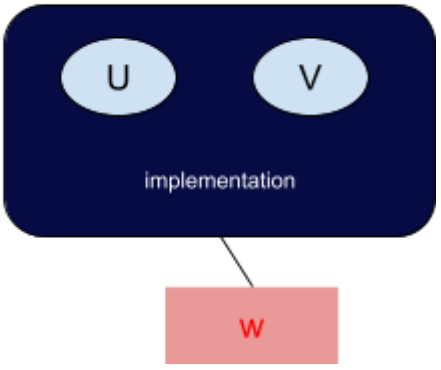
Java Concepts

Inheritance: Extends, Implements









Abstraction,
[example : extends]
Interfacing
[example: implement]

Abstraction : functionality open, but implementation detail hidden

Abstract methods characteristics:

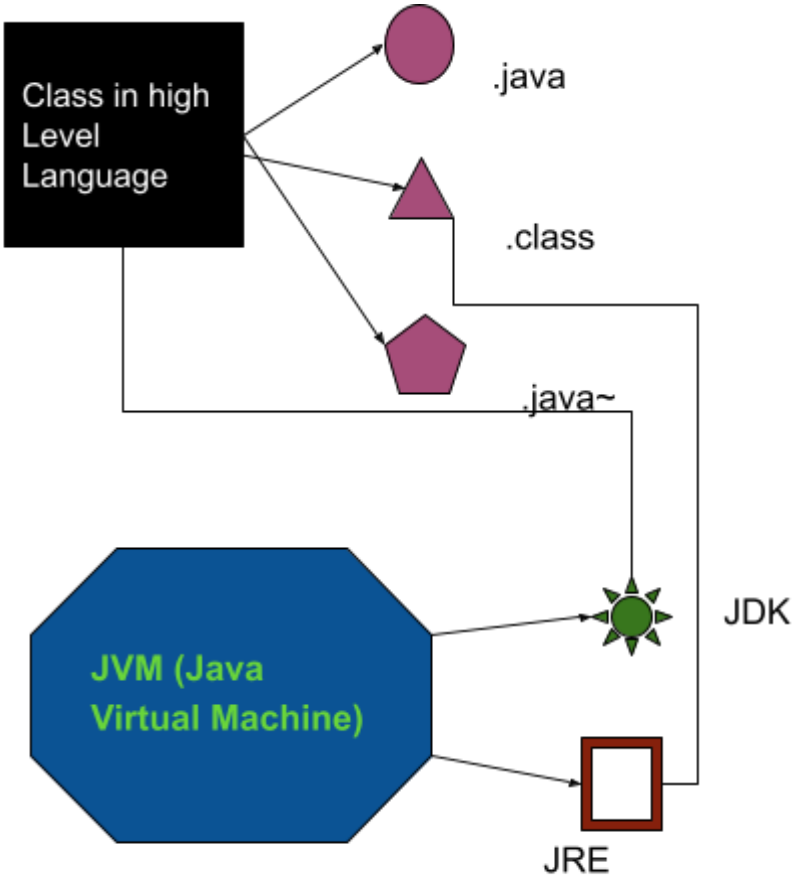
- ❖ Remains inside abstract class,
- ❖ Never final or static,
- ❖ No body to methods,
- ❖ Must be overridden.

Abstract Class:	Interface over class:
❖ No support for multi-inheritance, ❖ Have non-abstract & abstract methods, ❖ Partial or full abstraction	❖ Multiple inheritance, ❖ No instance of interface, ❖ No constructor, ❖ Full abstraction.

```
public class Marks{

   int a;
   int b;

   public Marks(){
   }

   public Marks(int u, int v){
    a = u;
    b = v;
   }

   public void  do(){
      System.out.print(a);
    }

    public double ok(){
     return 15.0;

    }
}
```

```
public class Main{

    public static void main(String arg[]){


      Marks  m = new Marks();

      m.do();

      int  y =  m.ok();

      System.out.println(y);

    }

}
```

```
public class Main{

  public static void main(String arg[]){

      System.out.println(10);

      System.out.println("This a
                                file");

      System.out.println('y');

    }
}
```



Console Output:

10

This is a file

y

```
public class Main{

    public static void main(String arg[]){

      System.out.println(10);

      System.out.println("This a file");
      System.out.println('y');

    }

}
```

Variable

Data type

Integer 1-10,000

Double 1.50

Float 1,55.005005

Byte 100000

Long 1000000000

Char ‘a’ , ‘b’, ‘0’

String “aabsjabbsajk”, “4%%%^%$#$”