* ABSTRACT CLASS IN JAVA
  + Abstract class in java is similar to an interface except that it can contain default method implementation.
* JAVA INTERFACES
  + Java includes a concept called interfaces
  + Is kind of like a class except a java interface can only contain method signatures and fields
  + A java interface cannot contain an implementation of the methods only the signature (name, parameters and exceptions) of the method
  + You can use interfaces in java as a way to achieve polymorphism
* EX. OF JAVA INTERFACE

Public interface MyInterface{

public String hello = “Hello”;

public void sayHello();

}

* EX. TO ACCESS A VARIABLE

System.out.println(MyInterface.hello)

* EX. IMPLEMENTING AN INTERFACE
  + Before you can really use an interface, you must implement that interface in some java class
  + A class that implements an interface must implement all the methods declared in the interface.

Public class MyInterface implements MyInterface{

Public void sayHello(){

System.out.println(MyInterface.hello);

}

}

* POLYMORPHISM
* ENCAPSULATION
  + Is defined as the wrapping up of data under a single unit. It is the mechanism that binds together code and the data it manipulates. Other way to think about encapsulation is, it is a protective shield that prevents the data from being accessed by code outside the shield
* ABSTRACTION
* INHERITENCE
* COMPOSITION OVER INHERITENCE