Chapter 11 - Abstract Classes & Interfaces

What does Abstract (class) mean?

Abstract in english means → existing in thought or as an idea without concrete existance

Abstract method

A method that is declared without an implementation

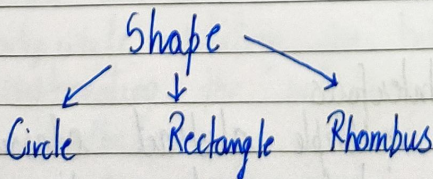   abstract void moveTo (double X, double Y)

Abstract Class

If a class includes abstract methods, then the class itself must be declared abstract, as in:

    public abstract class PhoneModel {
        abstract void switchoff();
        // more Code
    }

When an abstract class is subclassed, the subclass usually provides implementations for all of the methods in parent class. If it doesnt, it must be declared abstract

An Example

                    Shape
            ↙   ↓   ↘
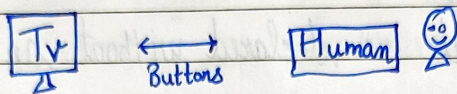   Circle   Rectangle   Rhombus

Note - It is possible to create reference of an abstract class
       It is not possible to create an object of an abstract class

We can also assign reference of an abstract class to the object of a concrete subclass.

Interfaces in Java

Interface in english is a point where two systems meet and interact

In Java interface is a group of related methods with empty bodies

An Example

```
interface Bicycle {
    Void applyBrake (int decrement);
    Void speedUp (int increment);
}


class AvonCycle implements Bicycle {
    int speed = 7;
    Void applyBrake (int decrement) {
        Speed = speed - decrement;
    }
    Void speedUp (int increment) {
        speed = Speed + increment;
    }
```

Abstract class vs Interfaces

We cant extend multiple abstract classes but we can implement multiple interfaces at a time.
Interfaces are meant for dynamic method dispatch

and run time polymorphism

Is multiple inheritance allowed in Java?
Multiple inheritance face problems when there exist
methods with same signature in both the super
classes.
Due to such problems, Java does not support multiple
inheritance directly but the similar Concept can
be acheived using Interfaces
A class can implement multiple Interfaces and
extend a class at the same time.

Note: ① Interfaces in Java is a bit like the Class
         but with a significant difference.
      ② An Interface can only have method Signatures,
         Constant← fields and default methods.
      ③ The class implementing an Interface needs to
         on declare the methods (not fields)
      ④ You can create a reference of Interfaces but
         not the Object
      ⑤ Interface methods are public by default.

Default methods
An interface can have static and default methods.
Default methods enable us to add new functionality to
existing Interfaces.
This feature was introduced in Java 8 to ensure backward
compatibity while updating an Interface.
Classes implementing the interface need not implement the
default methods.
Interfaces can also include private methods for
default methods to use.
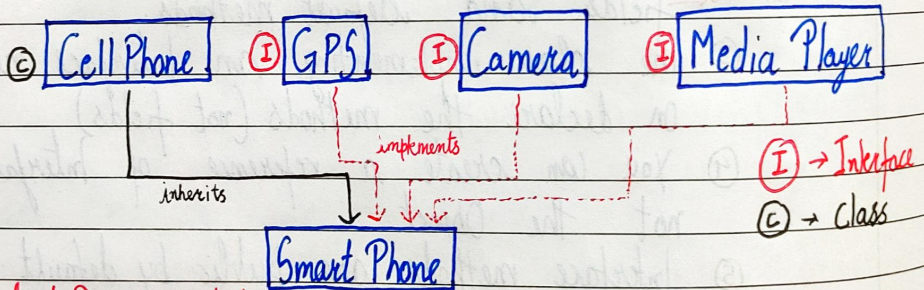
Inheritance in Interfaces
Interfaces can extend another interfaces:

```
public interface Interface 1 {
       Void meth 1 ();
}
```

```
public interface Interface 2 extends Interface 1 {
       Void meth 2 ();
}
```

Remember that interface cannot implement another interface, only classes can do that!

Polymorphism using Interfaces

ⓒ Cell Phone   Ⓘ GPS   Ⓘ Camera   Ⓘ Media Player

inherits   implements

Ⓘ → Interface
ⓒ → Class

Smart Phone

Similar to Dynamic method dispatch in Inheritance

GPS g = new SmartPhone(); → Can only use GPS methods
SmartPhone s = new SmartPhone(); → Can only use SmartPhone methods

Implementing an Interface forces method implementation.

Java Tutorial: Abstract Classes Vs Interfaces

Abstract class	Interface
1. It can contain abstract and non-abstract method	It can only contain abstract methods. We do not need to use the "abstract" keyword in interface methods because the interface is implicitly abstract.
2. abstract keyword is used to declare an abstract class.	interface keyword is used to declare an interface.
3. A sub-class extends the abstract class by using the "extends" keyword.	The "implements" keyword is used to implement an interface.
4. A abstract class in Java can have class members like private, protected, etc.	Members of a Java interface are public by default.
5. Abstract class doesn't support multiple inheritance.	Multiple inheritance is achieved in Java by using the interface.