

{mobidev}

Java Cocepts

Objectives:

- ✓ Overview :- Object, class, inheritance, interface, package
- ✓ Variable types and variable modifier types
- ✓ Basic java application :- create a class, define variables, functions, conditions and loops, arrays, Logging

{mobidev}

Overview:Object

An instance of a class

```
public static void main(String[] args) {  
    Vehicle obj = new Vehicle();  
}
```

{mobidev}

Overview:Class

A class is a blueprint from which objects are created

{mobidev}

Overview: Inheritance

This when a class is able use properties defined in another class.

A class that is used a basis for inheritance is called a superclass or base class.

A class that inherits from a superclass is called a subclass or derived class.

```
public class Inheritance extends Thursday {  
    public static void main(String[] args) {  
        writeSomething("Inheritance");  
    }  
}
```

{mobidev}

Overview: Interface

We use an interface to enforce certain properties in a class

{mobidev}

Variable types:

A variable is storage space

Each variable in java has a specific type, which determines the size and layout of the variable's memory.

Example types: int, long, String, double

{mobidev}

Modifier types:

Modifiers are keywords that you add to definitions to change their meanings.

To use a modifier, you include its keyword in the definition of a class, method or variable.

NB: The modifier precedes the rest of the statement

```
public class className {  
    // ...  
}  
private boolean myFlag;  
static final double weeks = 9.5;  
protected static final int BOXWIDTH = 42;  
public static void main(String[] arguments) {  
    // body of method  
}
```

{mobidev}

Modifier types: Access Control

They set access levels for classes, variables, methods and constructors

- default:- visible to the package
- private:- visible to the class only
- public:- visible to the world
- protected:- visible to the class and all subclasses

{mobidev}

Modifier types: Non Access Modifiers

static :- signifies that a member belongs to the class itself rather than to instances of the class

final:- finalizes the implementation of classes, methods and variables.

{mobidev}

Resourceful links

Java basics: <http://www.tutorialspoint.com/java/index.htm>

Learn java: <http://www.learnjavaonline.org/>