Agenda

- Class,Reference and Object
- Packages
- Access Modifiers
- this reference
- Types of Methods
    - constructor
    - setters
    - getters
    - facilitators
- Constructor Chaning

Class,Reference and Object (Demo01)

- class is a logical entity
- object is a physical entity
- variable of a class is called as reference in java
- process of creating object in java is also called as instantiation
- objects in java are created using new operator
- All the local variables in java need to be initialized before using them.
- default value that is used for reference type is null.
- if your reference is null and we call any methods on this null reference jvm throws an exception called as NULLPointerException.

Points to remember

- Q. Why we cannot write multiple public classes in single .java file?
- As per java language specification all the public classes must be defined inside their own .java files
- we can define multiple non public classess inside single java file.
- we cannot define multiple public classes inside single .java file.
- public classes are used so that they are visible across the multiple differnt packages.
- when we execute a java program 2 threads are started
    1. main thread
        - it is responsible for executing your code
    2. garbage collector thread
        - it is responsible for managing/deleting the obejcts from the heap section.

Packages (cmdLine-> demo02) (Demo02)

- packages are containers which are used to avoid name ambugity and to organise the code.
- packages are also used to contain related classes together

- eg
    - java.lang
    - java.util
    - java.io
- while writing the code into multiple packages, compiling and executing through the cmd line then always set the classpath first.
- remeber that every commnad should be given from the terminal from demo02->src directory

```
export CLASSPATH=../bin

javac -d ../bin Time.java

// dont forget to import the Time class into Program.java

javac -d ../bin Program.java

java main.Program

// If classpath is not set and you want to execute the Program.class then
you can use the below command
java -cp ../bin main.Program
```

```
//package com.sunbeaminfo.eattendance
package com.sunbeaminfo.eattendance.entities
package com.sunbeaminfo.eattendance.db
```

Access Modifiers (Demo03)

- To maintain the visibility of the class fields and methods java have provided the below access modifiers
    1. private
        - Are accessiable within the class.
        - Not accessable outside the class
    2. default (package level private)
        - Are accessiable within the class directly
        - Are accessiable in the sub/derived class directly in same package
        - are accessiable in the same package in differnt class on class object using . operator
        - not accessiable in othe packages.
    3. protected
        - Are accessiable within the class directly
        - Are accessiable in the sub/derived class directly in same package
        - are accessiable in the same package in differnt class on class object using . operator

        - are accessiable in the differnt package in the sub/derived class directly
        - not accessiable in other packages in differnt(non sub/derived) class.
    - 4. public
        - are accessiable within the class and sub/derived class in same or differnt package directly.
        - are accessiable in differnt class in same or differnt package on class object using . operator

Access Modifiers for visibility of class

- we can only apply default and public access modifiers for the visbility of the class
- if class is made default is is accessiable only within the package.
- if class is made public then is accesssiable within and outside the package.

this Reference (Demo04)

- when we call the non static methods of the class on objects then the reference holding that object is passed internally to that non static method.
- the reference that is passed is been copied in the reference called as "this" which is created internally for all the non static methods.
- we cannot change the address of object stored inside this reference.
- using this reference is completly option but as a standard practice it is a good habit of a programmer.
- this reference is used
    - 1. to identity the local variables from the class fields.

Types of methods

- 1. Constructor
- 2. Setters/ Mutators
    - To change the value of individual field of the class.
- 3. Getters / inspectors
    - te get the value of individual field of the class.
- 4. Facilitators
    - Methods that are responsible for business logic which deal with multiple fields of the class.

Constructor

- It is a special method in class
- why so special
    - 1. its name is same as that of the class
    - 2. it has no return type
    - 3. it gets automatically called when object is created.
- types of constructors
    - 1. Default/Parameterless ctor
    - 2. Parameterized Constructor
- If you do not provide any constructor inside our class then their still exists a constructor called as default constructor.
- If we provide any constructor the default constructor gets replaced.

Constructor Chaning

- calling a constructor from another constructor is called as constructor chaning
- for constructor chaning this statement should be the first statement inside the constructor.

LabWork

- assignment

Sunday Work

- Array of pointers
- Array of dynamic objects
- 2D Array