Agenda

- Revision
- Language Fundamentals
    - Naming Convention
    - Comments
    - Keywords
    - Data types
    - Variables
    - Literals
    - Operators
- Narrowing/Widening
- Wrapper classes
- Boxing/Unboxing
- Command Line Arguments
- Control Structures
- Java Method
- Class
- Reference
- Object ~~- Packages~~

Language Fundamentals

Naming Convention

- Camel Case
    - Every first letter of the word except the first word is kept in capital
    - It is used for
        - local varibles
        - class fields
        - method names
        - method parameters

```
void calculateTotalSalary(double salary , double incentive){
    double totalSalary = salary + incentive;
}
```

- Pascal Case
    - Every first letter of the word is kept capital
    - It is used for
        - class name
        - interface
        - enums

```
class SalesManager
```

- for the packages
    - keep all the words in small case

```
java.lang
java.util
```

- for constants/final fields
    - Every word in Capital

```
final double PI = 3.14;
```

Comments

```
// - single line comment
/* Multi-Line Comment */
/** Documention Comment*/
```

Keywords

- These are the reserved words which have special meaning to it.
- eg -> abstract boolean char double enum final int long public static void

Data types

- Data types defines 3 things
    - 1. Nature
        - What type of data can be stored inside it
    - 2. Memory
        - How much memory is required to store the data
    - 3. Operations
        - What all operations we can perform on that data.

- In java datatypes are divided into two categories
    - 1. Primitive types (Value Types)
        - boolean (1 bit)
        - char (2 bytes)
        - byte (1 byte)
        - short (2 bytes)
        - int (4 bytes)
        - long (8 bytes)

    - float (4 bytes)
    - double (8 bytes)
  - 2. Non Primitive types (Reference Types)
    - class
    - interface
    - enum
    - Array

Variables

- It is a container that can store a value
- The variable can be cretaed of primitive as well as non primitive type.
- The variable created of non-primitive type is called as reference
- The variable cretaed can be assigned with another variables or with the constant values

Literals

- the constant values used to initialize the variables are called as literals
- java have defined below six literals
  - 1. Integral Literals
  - 2. Floating Point Literals
  - 3. String Literals
  - 4. Character Literals
  - 5. Boolean Literals
  - 6. null Literal

```
int num1 = 10; // Intergral Literal
float salary = 1000.123f; // Floating point Literal
String name =  "sunbeam"; // String Literal
char ch = 'a'; // character Literal
boolean status = true; // Boolean Literal
Scannner sc = null; //null Literal
```

Operators

- Java have classified the operators into below categories

- 1. Arithmetic Operators
  - +, - , *, /
- 2. Assignment Operators
  - =,+=,-=, etc..
- 3. Comparaision Operators
  - ==, <,>,<=,>=,etc..
- 4. Logical Operators
  - &&, ||, !
- 5. Bitwise Operators
  - &,|,~, etc

- 6. Misc Operators
  - Ternary Operator (? 😃
  - dot Operator (.)

Narrowing/Widening (Demo01 -> Program01)

- Keeping the narrower type of data into the wider type is called as Widening
- Keeping the wider type of data into narrower type is called as Narrowing
- At the time of narrowing explicit typecasting is mandatory
- Narrowing may cause data loss

```
int num1 = 10;
double num2 = num1; // Widening

double num3 = 123.45;
int num4 = (int)num3; // Narrowing
```

Wrapper classes (Demo02 -> Program02 & Program03)

- All the primitive types are not classes but java have given classes for all such primitive types.
- These classes are called as Wrapper classes
- Use of Wrapper classes
  - 1. For conversion from primitive type to respective reference type
  - 2. To get the SIZE , Range(Max and min) value of a primitive datatype
  - 3. to use helper methods provided by these classes
  - 4. Java collection cannot store data of primitive types it can only store reference types.

Boxing/Unboxing (Demo01 -> Program03 and Program04)

- Converting value type into reference type is called as boxing
- If the boxing is done automatically without any helper methods then it is called auto-boxing

```
int num1 = 20;
Integer i1 = new Integer(num1); // Boxing
Integer i2 = Integer.valueOf(num1); // Boxing

Integer i3 = num1;  // Auto boxing
```

- Converting the reference type into value type is called as unboxing
- If the unboxing is done automatically without any helper methods then it is called as auto unboxing

```
Integer i1 = new Integer(10);
int num1 = i1.intValue();// Unboxing
int num2 = i1; // auto unboxing
```

Command Line Arguments (cmd_line -> Demo01)

- To compile and execute the code use below commands

```
javac Program.java

java Program 10 20
```

Passing Command Line Arguments in STS (Demo02)

- Right click on the program -> select Run as -> Run Configuration
- Inside the Arguments tab provide the Program arguments

Control Structures

- In java all the statemets are executed one after the other
- we can control the flow of statements using control statements
- Types of control statements
    1. Decision Making Statements
        - if statement
        - switch
    2. Loop Statements
        - do..while
        - while
        - for
        - for-each
    3. Jump Statements
        - break
        - continue

Java Method

- In java we cannot define the functions globally
- If we want to define a function that must be defined inside the class only.
- the functions that are defined inside the class are called as methods in java.
- Methods can be defined as satic or non static
- methods can return something or it can return void
- methods can have parameters or it can be parameterless

Class

- It is a logical entity
- It is also called as blueprint of an object
- Class is a non primitive type(Reference type) in java
- Class consists of fields and methods.
- Methods inside the class can be static or nonstatic
- Fileds inside the class can be static or nonstatic

- to define a class use the keyword 'class' and give the name(Identifier) to the class

```
class Employee{

}
```

Reference

- Variable created of a class is called as reference in java.
- reference points to the object of the class.
- local references gets created on java stack
- references declared as fields inside the class gets the memory on the heap section

```
Employee e1; // Reference in java
```

Object

- It is a physical entity
- It is also called as an instance of a class
- Process of creating object of a class is called as inistantation.
- Object defines 3 things
    - State
        - Fields of the class represent state of an object
    - Behaviour
        - Methods of the class represents behaviour of an object
    - Identity
        - Unique fields inside the class represents the identity of an object.
- All the objects in java are created using new operator

Labwork

- Read the slides max 20 mins
- Implement todays classwork if required
- Assignments
- Revise the concepts
    - class,objecct,reference
    - from cpp -> namespace, array(1d,2d,arry of pointers)