Java Fullstack

Course Agenda

* Core Java
* JDBC
* Javascript (ES new features)
* React.js
* Angular Framework
* Spring
* Spring Boot
* Spring Microservices
* DevOps and AWS

Core Java

Java is a platform independent programming language & it is object oriented as well.

Platform Independent: You can run java programs on any platforms without altering

Object Oriented: It allows you to create applications with real world entities which are called as objects, object will have 2 things mainly

1. properties (data/varaibles/fields)
2. behaviour (actions/functions/methods).

Software required

* JDK (Java Development Kit) - 1.8
* JRE (Java Runtime Environment) comes along with JDK
* Eclipse IDE (tools to develop java applications)

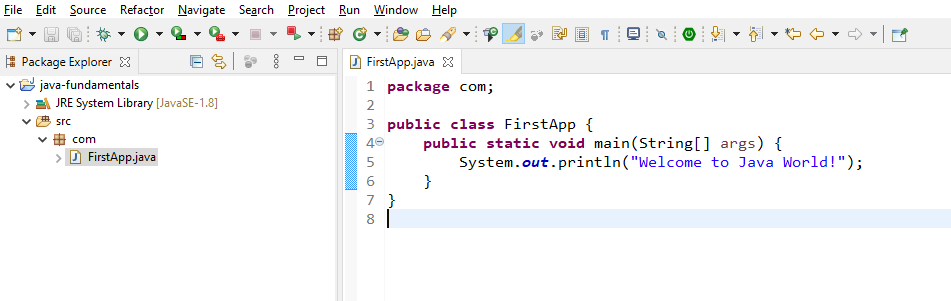
JRE: It is a runtime environment to run java applications, it will have JVM to execute byte codes

Building block of Object Oriented language

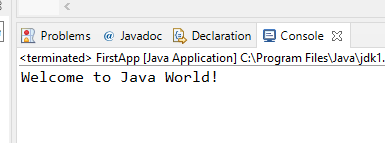
* Class: It is a template of an object, doesn’t take memory in the heap
* Object: It is a real world entity created from the class, takes memory in the heap

How to write our first java program

1. Use Eclipse & Create a java project, if in case you are in JavaEE perspective change it
2. Create a class & name it and also mention the package name
3. Create an entry point method i.e., main method
4. Run the main method.



Output:



Java Features

* Simple to understand, because most of the low level functionalities are abstract
* Object Oriented
* Platform Independent
* Architecture Neutral(x64, X86)
* Distributed applications
* Robust
* Secured
* Large Community
* Open Source

Fundamentals of Java

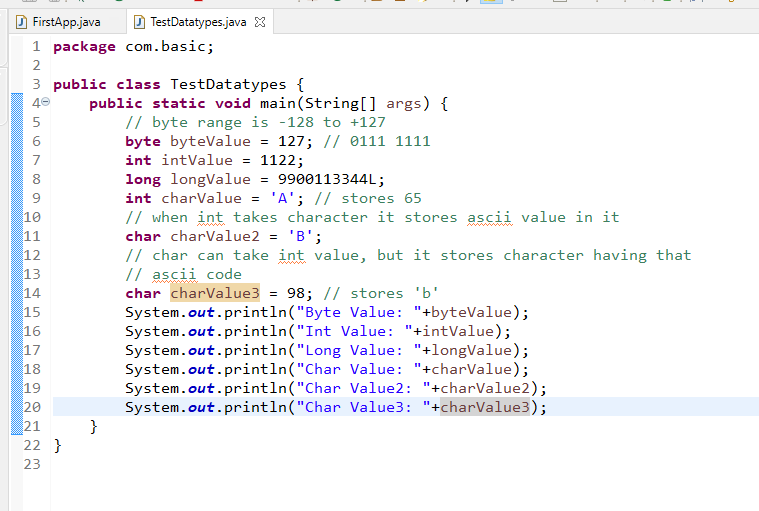
* Datatatypes
* Keywords
* Operators
* Conditional Statements
* Arrays
* Loops
* Variables
* Methods
* Classes
* Objects

Datatypes in Java

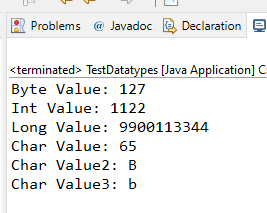
These are set of keywords used to create variables to store some value like numbers, characters, boolean, object and etc.

There are 2 types of datatypes in java

1. Primitive type: These are simple types whose size is defined
   1. byte = 1 byte
   2. short = 2 bytes
   3. int = 4 bytes
   4. long = 8 bytes
   5. float = 4 bytes
   6. double = 8 bytes
   7. char = 2 bytes
   8. boolean = 1 byte
2. Derived type: These are complex types that can store multiple values
   1. array
   2. class
   3. interface
   4. enum



Output:



Keywords in Java

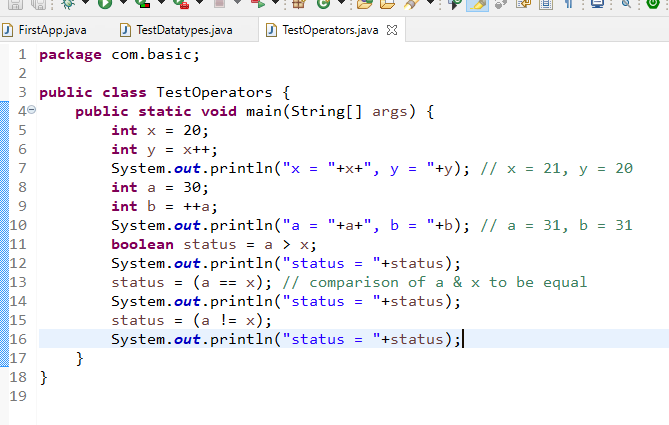
These are reserved words in Java, which must not be used to name the variables, methods or classes, they solve specific purpose

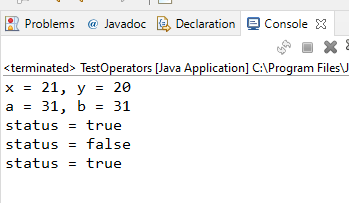
int, char, float, double, boolean, long, final, public, private, class, protected, static, void, return, if, for, do, while, switch, break, continue, abstract, interface, enum, extends, super, this and so on.

Operators in Java

+, -, \*, /, %, ++, --, =, ==, <, >, <=, >=, !=, &&, ||

TestOperators.java

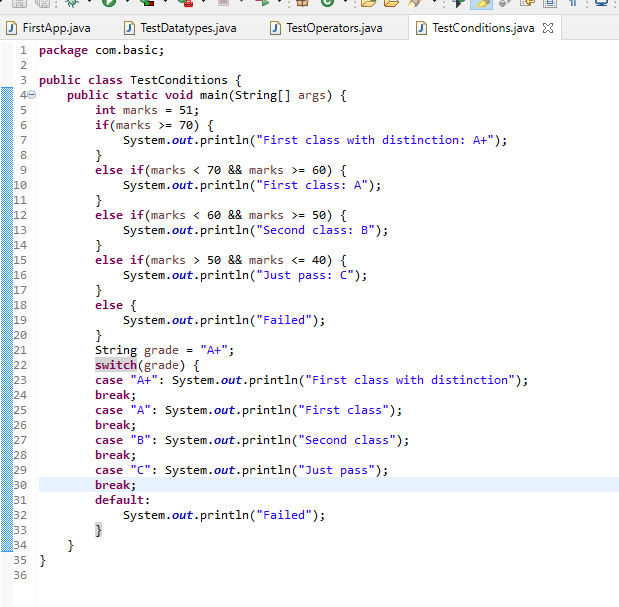




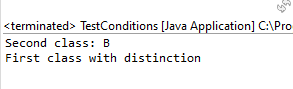
Conditional Statements

These are used when you want to execute the statements based on some conditions, we have following conditional statements

* if
* if else
* if else if .. else
* switch



Output:



Scanner class:

It is a predefined class present in java.util package, which is used to take input from the keyboard, you need to import this class using ‘import java.util.Scanner’

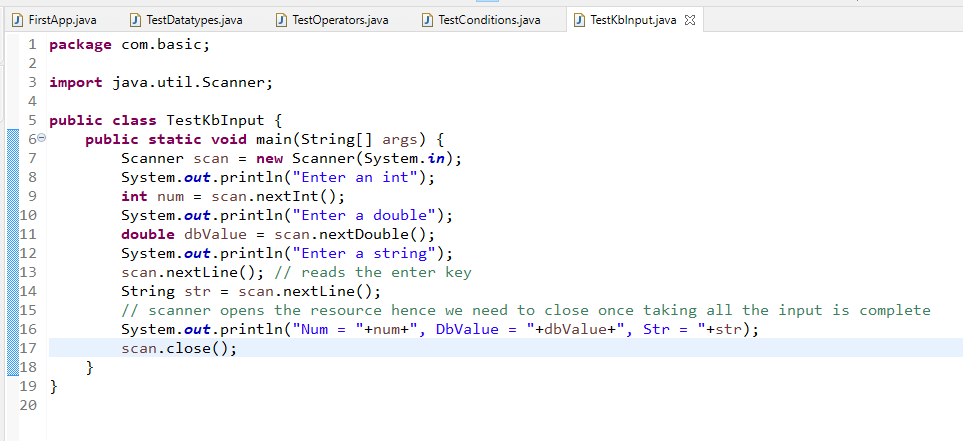
It defines lot of methods to accept different types of value like

nextInt(), nextLong(), next(), nextFloat(), nextDouble() and so on

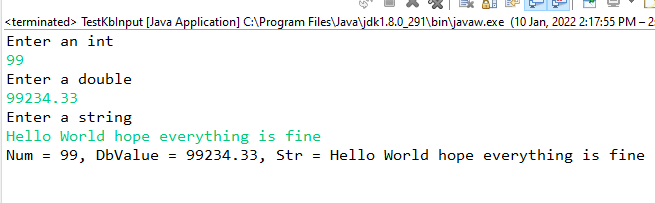
All the above methods you can call by creating scanner object

Scanner scan = new Scanner(System.in);

The above code creates object of scanner, System.in is a reference to the input stream i.e., keyboard



Output:



How to work with arrays

In Java you can create arrays using [], you can create arrays of simple types to complex types

Different ways of initializing array & their values

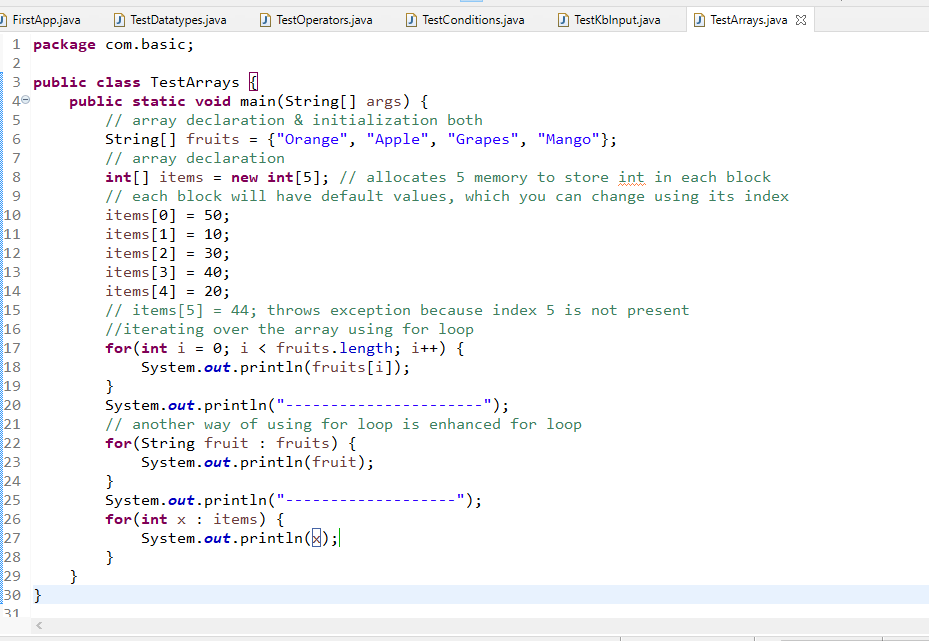
int[] items = {10, 20, 15, 35, 30}; // it is an integer array which is initialized

int[] items = new int[5]; // it is an integer array but not initialized the blocks

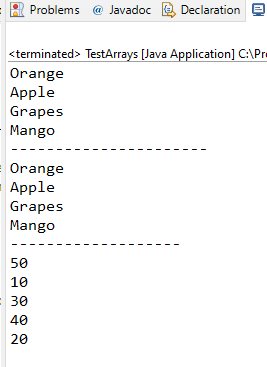
items[0] = 10; items[1] = 30, … items[4] = 50;

String[] fruits = {“Apple”, “Mango”, “Orange”}; // string array which is initialized

You can use loops to iterate over each items of the array i.e., for loop



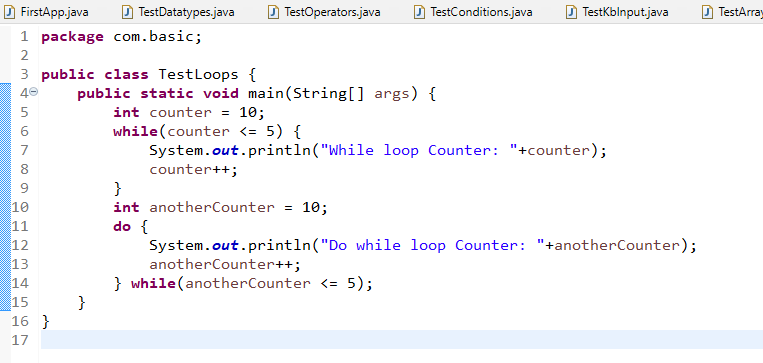
Output:



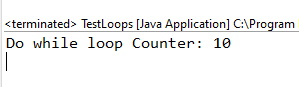
Loops in Java

We have 3 types of loops

1. for: When you want to iterate fixed number of times or you know how many times you want iterate
2. while: When you want to iterate until condition is true
3. do-while: When you want to iterate until condition is true, but in do while the statement is atleast executed once



Output:

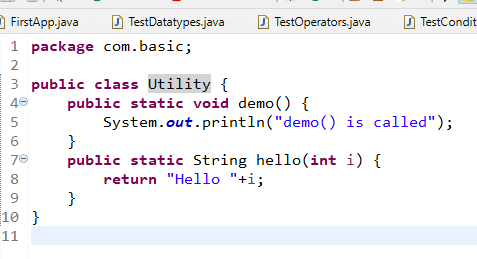


Methods in Java

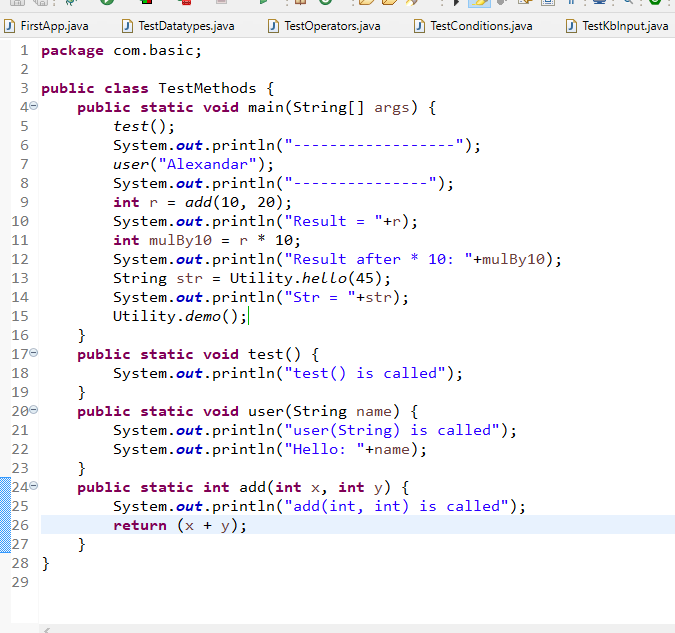
You can create methods in class in 2 ways

* Using static keyword: You can call this method without creating object
* Without static keyword: You can call this method by creating object

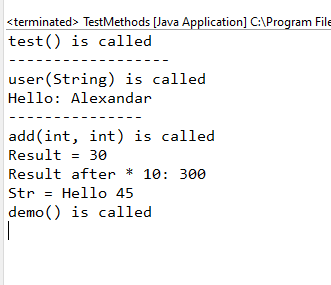
Utility.java



TestMethods.java



Output:



Activities

Create a Exercise folder & day1 sub-folder & keep all the day1 activities here, same way on 2nd day you will create day2 sub-folder and so on.

1. Try all the above examples
2. Create a method that accepts name as a parameter & returns a welcome message, call this method from the main method and display the welcome message

Ex: If you enter Raj the method should return Hello Raj, welcome to java

1. Create a method that accepts 3 numbers as parameters & returns the largest number among them

Ex: If you pass findLargest(3, 9, 1), then it must return 9

1. Create a method to accept 3 numbers & returns the sum of biggest & smallest number.

Ex: If you pass sum(9, 1, 3) then it must return (9 +1), if you pass sum(1, 2, 5) then it must return (5 + 1)

1. Create a method that accepts 3 numbers & prints their digits in words ex: If the input is 754 then the output should be “Seven Five Four”