Java and Microservices

Java is platform independent & object oriented language.

Java applications are compiled to byte code and these bytes codes are run by JVM (Java Virtual Machine)

When you install Java you get 2 things

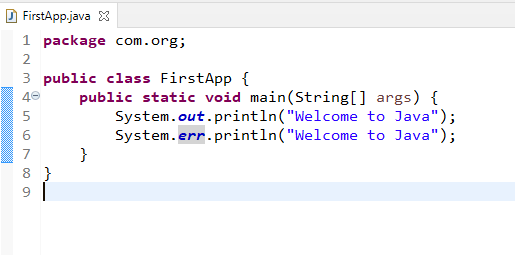
* JDK - Compiler
* JRE - Java Runtime Environment, JVM is part of JRE

Class: Blueprint of an object

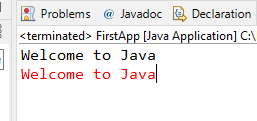
Object: It is an instance of the class

Package: It is a folder that will have all the classes & sub packages, package names follow reverse domain convention i.e., com.org, com.google, com.ibm,

FirstApp.java



Output:



Fundamentals of Java

1. Datatypes
2. Keywords
3. Variables
4. Operators
5. Loops
6. Conditional Statements
7. Classes & Objects

Datatypes: They mention what kind of value a variables stores, there are two types

1. Primitive Datatypes - byte, short, int, long, float, double, char, boolean
2. Non-Primitive Datatypes - Classes, Interfaces, Enums, Arrays

|  |  |
| --- | --- |
| Datatypes | Size in byte |
| byte | 1 |
| short | 2 |
| int | 4 |
| long | 8 |
| float | 4 |
| double | 8 |
| char | 2 |
| boolean | 1 |

By default all the whole numbers are treated as int, but if you want to represent higher precession types you have some suffix

125: It is treated as int

125L: It is tread as long, you can also use lower case ‘l’

20.05: It is treated as double by default

20.05F: It is treated as float, you can also use lower case ‘f’.

byte b1 = 125; // 125 is byte only

short s2 = 350; // 350 is short only

int s3 = 350; // 350 is int

long s4 = 450; // 450 is treated as int

float f = 45.0; // it gives error, because 45.0 is treated as double

float f2 = 45.0f; // 45.0 is treated as float

Keywords: These are reserved words we must not use to create identifiers in that name, some of the keywords are:

int, long, float, short, byte, char, boolean, static, void, public, private, protected, package, class, interface, enum, for, do, while, if, else, return, abstract, default

Operators:

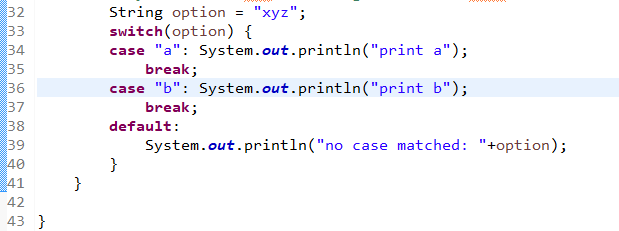
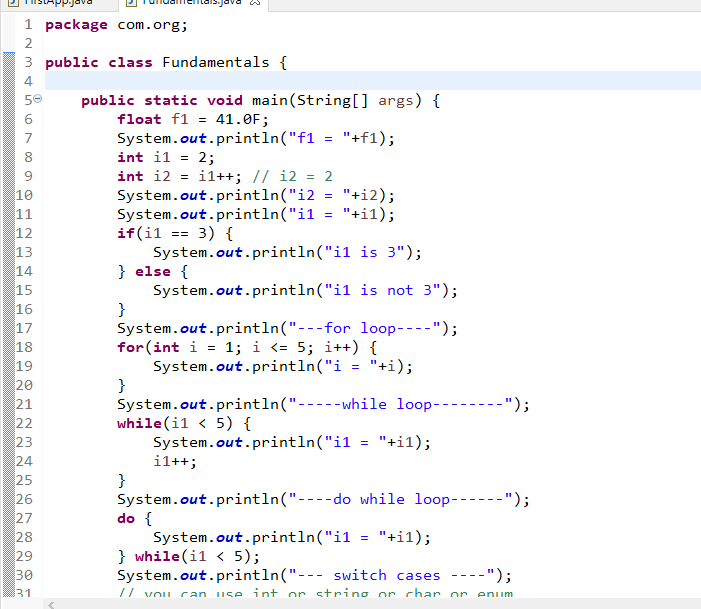
=, ==, !=, <=, >=, ++, --, \*, /, %, &&, ||

Loops:

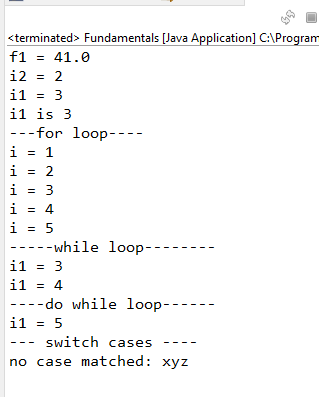
* for
* while
* do while

Conditional statements:

* if
* if else
* if else if else if … else
* switch

Using Fundamentals  


Output:



classes & objects

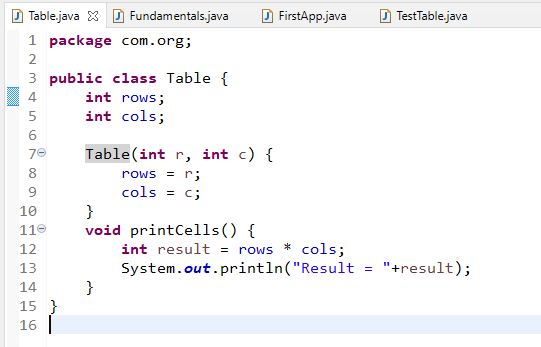
Classes: These represents the structure of the object, it is a blueprint of the object

Object: These are created from the classes, it is an instance of the class

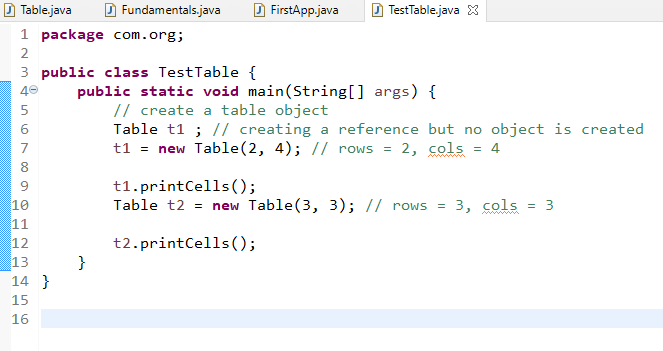
What all the things you can have in a class

1. variables: These store the data
2. methods: These can be any name, they perform actions
3. constructors: same as the class name but used to create the object, initializes the object properties/variables

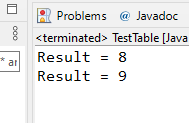
Table.java



TestTable.java



Output:



Note: Java by default creates a constructor i.e., default constructor only in a case where you don’t have any constructors in the class.

Apart from the variables, constructors, methods we can have

static variables & static methods

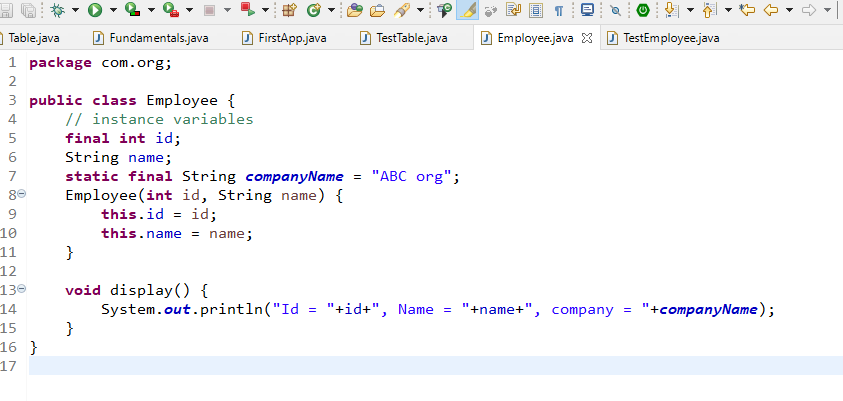
Static: It is part of the class not a part of an object, you can access static members without creating object, you can use class names directly.

Static means common to all the objects, if you modify it is reflected to all the objects, but it can be changed.

How to create constants in Java

final keyword: It can be used to create constants

Employee.java



Final instance variable can be initialized at the time declaration or in constructor,

Static Final variable need to be initialized at the time declaration only,

Final Instance variable is a constant per object

Final static variables are constants for all the objects

OOPs principles

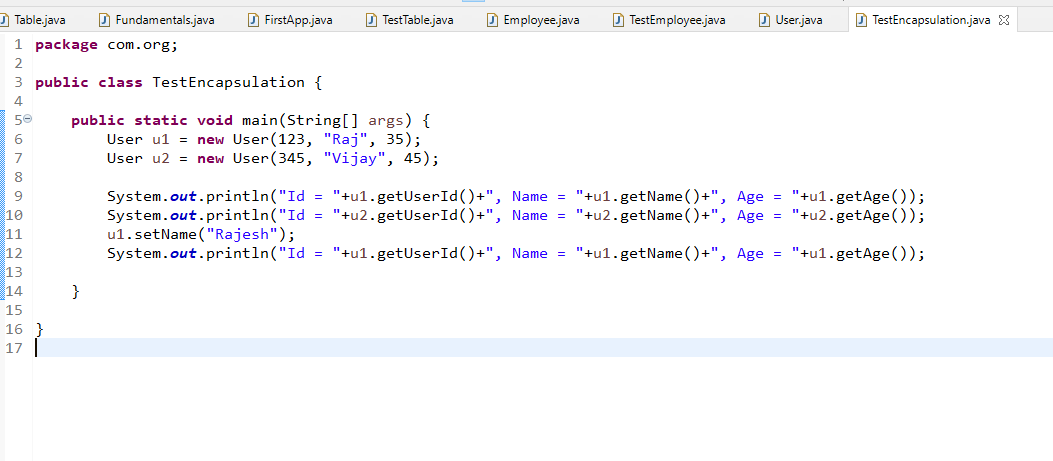
1. Encapsulation
2. Inheritance
3. Polymorphism
4. Abstraction

Encapsulation: Data hiding, where you make variables private and only way can access it is only through the public methods of the enclosing class.

User.java



TestEncapsulation.java



Output:

