

# Day 1

Monday, August 28, 2017 9:49 AM

## Why Java? -

- Pros:
  - JVM
  - Well documented.
  - Lots of built in libraries.
  - Automatic memory management (garbage collection).
  - Open source
  - Large job market
- Cons:
  - Performance

JVM - Java Virtual Machine - Java programs are compiled into bytecode. The JVM executes bytecode which means any platform with a JVM can execute it. Write once, run anywhere.

## OOP

- Encapsulation - Data Hiding. The process of wrapping data and methods that act on that data into a single unit. This is done by using a class and access modifiers. Benefits are that it can be used to protect data from misuse. It is a protective wrapper.
- Abstraction - The process of abstraction in Java is used to hide certain details and only show the essential features of the object. Achieved through the use of Interfaces and Abstract Classes.
- Inheritance - The process where a class acquires methods and fields of a parent class.
- Polymorphism - The ability of something to take on many forms. Objects can be treated like their super classes, method overriding, method overloading.

## Access Modifiers

- private - only accessible within the class.
- (default) - no keyword, accessible within the class and package.
- protected - accessible within the class, package and sub classes.
- public - accessible everywhere

## Constructors

- No return, so no return type
- First line is always `super()` by default, unless we specify `this()` or another `super()`.
- If we do not specify, we will get the default constructor which takes no parameters and just calls `super()`;

## Naming conventions

- Package - `com.reverse.domain.name` (all lower case)
- Class - `PascalCase` (first letter of words capitalized)
- Function - `camelCase`
- Variable - `camelCase`
- Constants - `UPPER_CASE_WITH_UNDERSCORES`

## Types

`//whole number`  
`byte b; //8 bits signed`  
`short s; // 16 bits signed`

char c; // 16 bits unsigned

int i; // 32 bits signed

long l; // 64

//decimal values

float f; // 32 bits 1 bit is for sign, 8 bits are exponent and 23 bits of significance

double d; //64 bits 1 bit sign, 11 bits exponent, 52 bits of significance

boolean bool; //true or false

#### Control Statement

- If - else if - else
- Loop
  - While
  - Do-while
  - For
- Switch - works with byte, short, char, int and as of Java 7 String
- Ternary
- Continue vs break
- labels

#### Scopes of a variable

- Static - sometimes also referred to as class scope, accessed directly from the class no object is required.
- Instance - each object or instance of a class has it's own scope of variables
- Method - each method has it's own scope for creating variables
- Block - every block, denoted by {}, has it's own scope.