Lecture 1 (06/07/2019)

github: <https://github.com/nagabhushanamn/nex-wissen>

Course Outline

L1:

* Java 8
* Java Enterprise Edition

L2:

* Spring

L3:

* Web Services (SOAP & REST)

L4:

* Micro Services

L5:

* HTML & CSS
* Javascript

L6:

* Angular v7

L7:

* Dev-ops Tools

L8:

* Security

L9:

* Cloud (AWS)

L10:

Java Features

* OOP + FP
* Static typed
* Portable
* Concurrency & Parallelism

Java Notes:

* By default has 1 main thread and 21 threads are java side threads
* Use package <package-name>
* Compile using javac –d <bin-dir> filename.java
* Run using java <package-name>.filename (while in the <bin-dir>)

Project Structure

app

|  
|----- package1  
| |  
| |  
| |--------- package1.1  
|  
|  
|------ package2  
|  
…  
|------ resources

Java Structure

File.java

// package declaration (1)

// import statement (n)

// public Type {class | interface | enum | annotation } File

// default Type (n)

Need for packages

* To organize code well
* Avoid type-name collisions

Package Syntax

package root;

package root.sub1.sub2. …;

eg. To create Employee.java in the employee module, employee management project, for company wissen which is of commercial type

package com.wissen.em.employee;

(use atleast 3 levels)

Object

* State (data)
* Behavior (methods)
* Identity (address)

Class

* Describes a type of object that have same State and Behavior
* Can be instantiated to create Objects

OO Concepts

* Abstraction / Interface / Front-End / API
  + Easy to use
  + Loose coupling between dependent and dependency
* Encapsulation / Implementation / Back-End
  + The actual implementation behind the interface
  + Hides complexity of the implementation
* Inheritance
  + Reusability
  + Generalization/Specialization
* Polymorphism