Lesson 2: Programming Fundamentals

Aniruddha Kudalkar

Objective

- VSCode Setup
- Understanding Java Concepts
- Implementing Codes
- Writing Test Cases

VSCode Setup

VSCode

Lets us understand first, very basics of programming

- Installing Extensions
- Checking Setup
- Test Case Setup

Installing Extensions

you can install the Coding Pack for Java, which includes VS Code, the Java Development Kit (JDK), and essential Java extensions. The Coding Pack can be used as a clean installation, or to update or repair an existing development environment.

Checking Setup

Visual Studio Code allows you to debug Java applications through the Debugger for Java extension. It's a lightweight Java debugger based on Java Debug Server, which extends the Language Support for Java by Red Hat.

Test Case Setup

Testing Java in Visual Studio Code is enabled by the Java Test Runner extension. It's a lightweight extension to run and debug Java test cases.

Understanding Java Concepts

Java™

Java Basics

write once, run anywhere (WORA)

- Java keywords
- Variables & Data Types
- Packages
- Access Modifiers
- Classes, Objects and Constructor
- Operators

Java Basics

write once, run anywhere (WORA)

- Arrays
- Conditionals
- Loops
- Enums
- Strings, Numbers, Wrapper Classes
- Exception Handling
- File Handling

Java Basics

write once, run anywhere (WORA)

- Abstract Classes
- Interfaces
- Inheritance
- Polymorphism
- Threading
- Generics
- Collections

Java Keywords

which is a word with special meaning in a particular context

abstract	continue	for	new	switch
assert***	default	goto*	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum****	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp**	volatile
const*	float	native	super	while

* not used

** added in 1.2

*** added in 1.4

**** added in 5.0

Variables and Data Types

Variable value for particular type of data

Variables

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

Data Types

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html

Packages

Grouping of similar type of class, interfaces and enums

Using Packages

https://docs.oracle.com/javase/tutorial/java/package/packages.html

Access Modifies

determine whether other classes can use a particular field or invoke a particular method

Access Modifiers

https://docs.oracle.com/javase/tutorial/java/javaOO/accesscontrol.html

Classes and Objects

Foundation of Object Oriented Principles

Classes

https://docs.oracle.com/javase/tutorial/java/javaOO/classes.html

Objects

https://docs.oracle.com/javase/tutorial/java/javaOO/objects.html

Something more

- https://docs.oracle.com/javase/tutorial/java/javaOO/more.html
- https://docs.oracle.com/javase/tutorial/java/javaOO/nested.html
- https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html

Operators

Operators are special symbols that perform specific operations

Operators

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/operators.html

Arrays

storage/retrieval of elements sequentially

Array

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

Conditionals

These are decision-making statements; if-else, switch-case

if-else

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/if.html

switch-case

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html

Loops

the looping statements; for, while, do-while

While

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/while.html

For

https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html

Enums

special data type that enables for a variable to be a set of predefined constants

Enum

https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html

String, Number, Wrappers

special data type that enables for a variable to be a set of predefined constants

Strings

https://docs.oracle.com/javase/tutorial/java/data/strings.html

Numbers

https://docs.oracle.com/javase/tutorial/java/data/numberclasses.html

Wrapper Classes

https://docs.oracle.com/javase/tutorial/java/data/autoboxing.html

Exception Handling

Java uses exceptions to handle errors and other exceptional events.

Exceptions

https://docs.oracle.com/javase/tutorial/essential/exceptions/

File Handling

Java uses exceptions to handle errors and other exceptional events.

File IO

https://docs.oracle.com/javase/tutorial/essential/io/fileio.html

Abstract Classes

They represents incompleteness

Abstract Class

https://docs.oracle.com/javase/tutorial/java/IandI/abstract.html

Interfaces

They represents incompleteness

Interface

https://docs.oracle.com/javase/tutorial/java/IandI/createinterface.html

Inheritance

Reusability is main topic

Inheritance

https://docs.oracle.com/javase/tutorial/java/IandI/subclasses.html

Polymorphism

Multiple forms of one operation

Polymorphism

https://docs.oracle.com/javase/tutorial/java/IandI/polymorphism.html

Threading

Multiple execution paths

Threads

https://docs.oracle.com/javase/tutorial/essential/concurrency/threads.html

Generics

generics enable types (classes and interfaces) to be parameters when defining

Generics

https://docs.oracle.com/javase/tutorial/java/generics/index.html

Collections

store, retrieve, manipulate, and communicate aggregate data

Collections

https://docs.oracle.com/javase/tutorial/collections/TOC.html

Thanks,

Let's Code Now

Credits

- https://en.wikipedia.org/
- https://docs.oracle.com/