Software Design & Modelling 2

# Introduction to Java:

* General-purpose, object-oriented programming language
* Created in 1995 by Sun Microsystems (acquired by Oracle in 2010)
* Platform independent: write once, run anywhere
* Typed language
* Typically uses JDK (Java Development Kit) and IDEs like Eclipse and IntelliJ
* Uses JVM (Java Virtual Machine) to run compiled bytecode
* Automatic memory management via garbage collection

# Basic Syntax in Java:

JAVA:

A computer screen with text on it

Description automatically generated

C#:



# Data Types:

* Primitive Data Types in Java:
* int , long , short , double , Boolean, char , float , byte
* Objects and arrays are reference types
* C#:
* Similar primitive data types but with nullable types (int?, double?, etc.)
* Bool instead of Boolean

# Console Input/Output:

* Java provides mechanisms to read user input from the console and write output to the console window

A screen shot of a computer program

Description automatically generated6

# Conditional Statements:

* Same if-else syntax in Java and C#
* Ternary operators ( condition ? value1 : value2 ) are present in both

A screen shot of a computer code

Description automatically generated

# Control Structures – Loops:

* Same syntax for ‘for,’ ‘while’ , and ‘do-while’ loops
* C# offers an explicit foreach keyword, but Java uses for

A computer screen with white text

Description automatically generated

# Arrays:

* Same syntax for declaring and using arrays
* Arrays in both Java and C# are fixed in size after initialization

A screenshot of a computer code

Description automatically generated

# Collections:

* Both Java and C# have dynamic array-like collections

A black background with white text

Description automatically generated

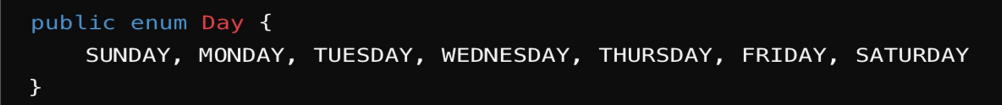
* HashSet (Unordered Collection, No Duplicates)
* HashMap (Key-Value Pair)
* LinkedList (Doubly-Linked List)

# Constants and Enums:

* Constants in Java can be declared using the *final* keywords



* Enum is a special type to define a set of constants (named values)





# Strings:

* String in Java is a class, but in C# is a data type

A computer screen with text

Description automatically generated with medium confidence

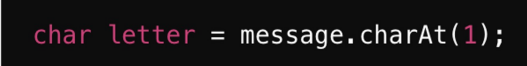
* Length:



* Substring:



* CharAt:



* Contains:



# Classes & Objects:

* A class in Java is a blueprint or encapsulation for defining properties (fields) and behaviors (methods) for objects
* Constructor: a special method (has the class name) with no return

A screen shot of a computer code

Description automatically generated





# Class Inheritance:

* Inheritance is a mechanism where a new class (subclass) derives from an existing class (superclass), using the extends keyword
* C# uses : for inheritance
* The subclass inherits fields and methods from the superclass and can also add its own fields and methods
* Class can inherit only one class (C# allows multi-class inheritance)

A screenshot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

# Abstract Class:

* An abstract class cannot be instantiated (cannot have objects) on its own, and is meant to be subclassed
* An abstract method is s declared without an implementation (no body). It must be implemented by any concrete subclass.

A black background with blue and red text

Description automatically generated

A screen shot of a computer

Description automatically generated

# Final Class:

* A final class cannot be subclassed (no other class can extend it)

A black background with text

Description automatically generated

* A final method cannot be overridden by subclasses, making its implementation unchanged in subclasses

A black screen with white text

Description automatically generated

# Interfaces:

* An interface is a completely "abstract class" that is used to group related methods with empty bodies
* Java allows a class to implement multiple interfaces

A screen shot of a computer code

Description automatically generated

# Packages:

* Used for organizing code, and if used, must be declared at the 1st line

A black background with white text

Description automatically generated

* C# uses namespace:

A black background with red and blue text

Description automatically generated

* Importing classes from existing packages using the import keyword

A black background with white text

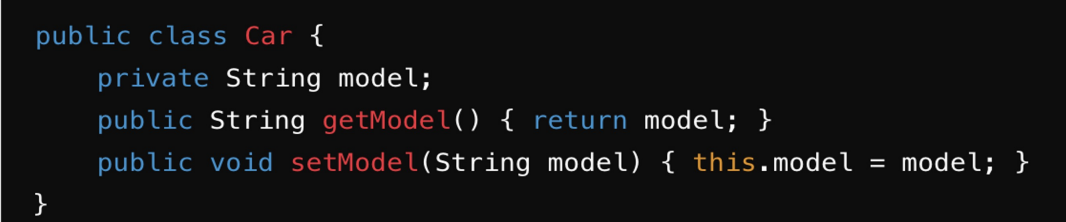
Description automatically generated

# Access Modifiers:

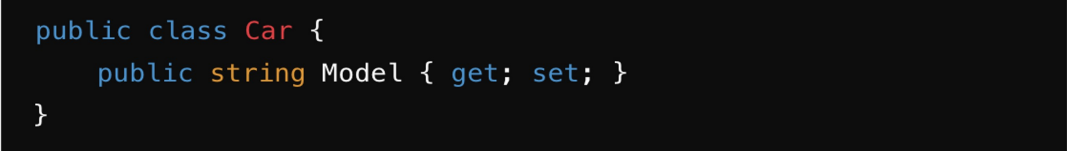
* private (within class)
* public (everywhere)
* protected (within package, or outside package through child class)
* Default (no modifier) (within package, no access outside package)

# Properties (Setters/Getters):

* Java

****

* C#

****

# Static Method & Fields:

* Allow class-level functionality, using the “static” keyword

A computer screen shot of a program code

Description automatically generated

# Documentation and Comments:

* **Single-Line Comments:** for brief explanations in the code
* **Multi-Line Comments:** for longer explanations, or comment out code

A black screen with white text

Description automatically generated

* **Javadoc Comments:** for generating API documentation from source code, providing descriptions for classes, methods, and fields

A computer screen with white text

Description automatically generated

# Exception Handling:

* A mechanism to handle runtime errors and other exceptional conditions, preventing the program from crashing

A screenshot of a computer program

Description automatically generated

# Files and I/O:

* The File class is used to create, delete, and check file properties

A computer screen shot of code

Description automatically generated

* • Files (in the java.nio.file package) provides a simple way

# Threads and Concurrency:

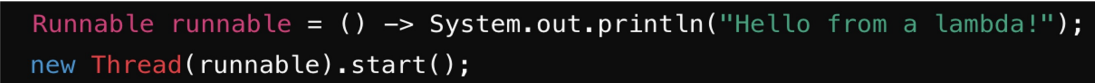
* A thread is a lightweight unit of execution within a process
* Allows programs to perform multiple tasks simultaneously (multithreading)
* Example: Web server handling multiple client requests at the same time



# Lambda Expression:

* A feature introduced in Java 8 that allows writing concise and expressive code
* They enable you to pass behavior as parameters, write inline implementations, and reduce boilerplate code
* Syntax:





# Generics:

* Allow you to define classes, interfaces, and methods with type parameters
* You need a storage container that can hold various types of items, such as documents, tools, toys, etc.

A black screen with white text

Description automatically generated

A black background with white text

Description automatically generated