Introduction to Java

LASSONDE SCHOOL OF ENGINEERING | YORK UNIVERSITÉ UNIVERSITY | EECS ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

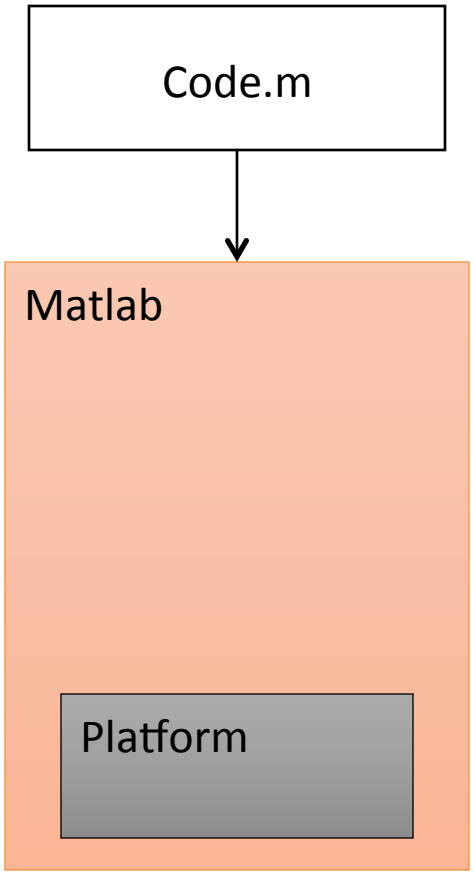
Why Java?

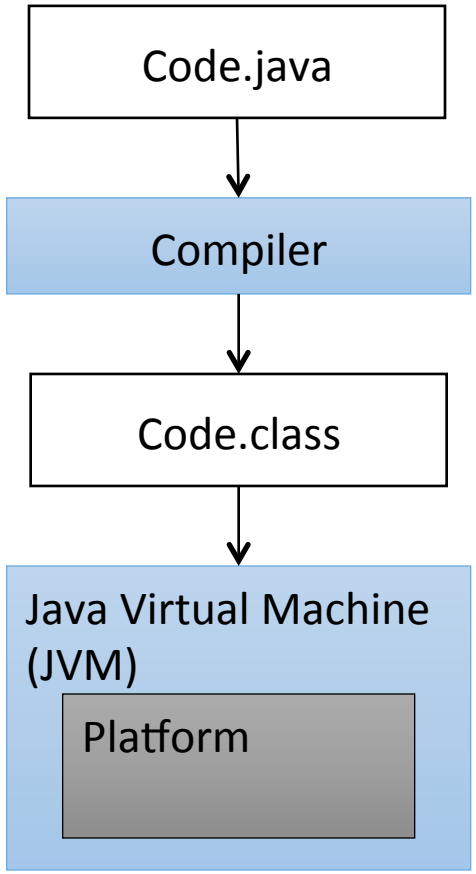
- The most popular programming language in the world (as of January 2018)
  - Python is rising fast though
- Widely used in industry:
  - Financial trading
  - E-Commerce
  - Server applications
  - Software development
  - Entertainment
- Examples:
  - Amazon, eBay, Gmail (backend servers)
  - Android smartphones (Java-like programming)
  - Blu-ray players and set top boxes
  - Matlab and Maple user interfaces
  - Eclipse

How it Differs from Matlab

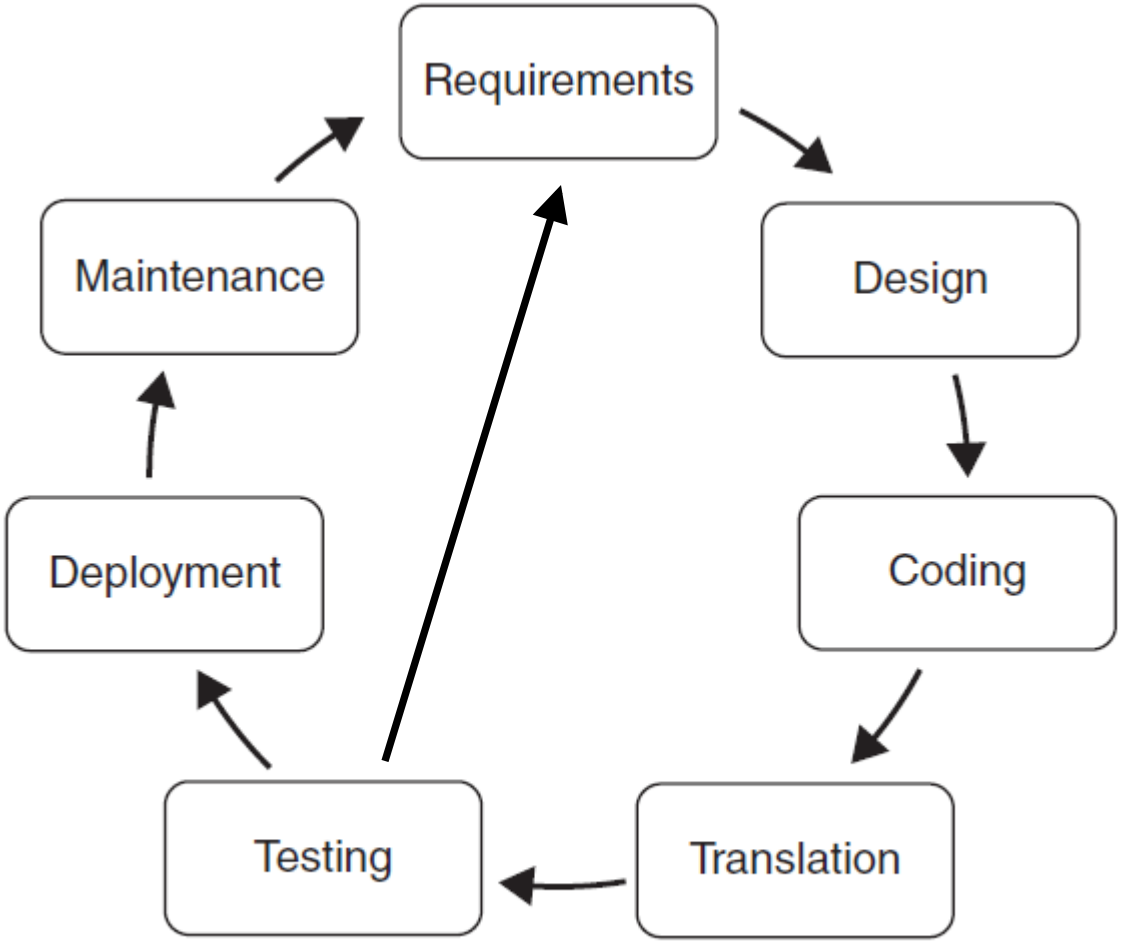
- Java is a general-purpose language
- Java does not need to run in an application
  - Java development kits are freely (legally) available
- Java is “statically typed”, meaning
  - Variables must have their data types specified
    e.g., int, float, boolean, String
  - Variables cannot change their data types
- Java code must be compiled before it is run

How it Differs from Matlab (con't)





The Programming Process



Example: BMI Calculator

- Create variables to store a person's height (in metres) and weight (in kilograms)
- Calculate the corresponding Body Mass Index (BMI) using the following formula:
  - BMI = weight / height^2
- Output the BMI, rounded to one decimal place
- Oracle: online BMI calculator

BMI Calculator in Matlab

```
% This script calculates the BMI corresponding to
% the values defined for height and weight.

weight = 60;    % weight in kg
height = 1.7;   % height in m

bmi = weight / (height * height);
display(sprintf('\nYour BMI is %.1f.\n', bmi))
```

BMI Calculator in Java

```
/*    This class calculates the BMI corresponding to
      the values defined for height and weight. */


public class BMICalculator
{
  public static void main(String[] args)
  {
    double weight = 60;  // weight in kg
    double height = 1.7; // height in m
    double bmi = weight / (height * height);
    System.out.printf("\nYour BMI is %.1f.\n", bmi);
  }
}
```

Similarities in Syntax

- Assignment uses the "=" operator
- Rules for identifiers (i.e. variable names)
- Basic arithmetic operators
  - Other operators are not, more on this later
- Output formatting specifications

Differences in Syntax

- Comment delimiters // and /* */
- Class and method declaration
- **Must** declare variable type
- Every statement **must** end with a ";"
- Strings enclosed in double quotes, not single

Java Language Structure - Classes

- Containers for code
  - Code enclosed in braces {}
- Define how real-world objects and/or concepts are represented and behave
- Factories for objects

Java Language Structure - Variables

- Hold data
- Must be declared before use
- Must define a data type
  - The data type cannot change during the program

Java Language Structure - Method

- Define object behaviour
  - Code enclosed in braces
- Access specifier (e.g., “public”) define who can use it and/or whether it needs an object (e.g., “static”)
- Returns a value
  - Specify data type or “void”
- Required arguments

Java Language Structure - Keywords

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

Java Language Structure - Identifiers

- Name of a language element (i.e., class, variable, method)
- Cannot begin with a digit or be a keyword
- No additional rules for identifiers, but general conventions:

IDENTIFIER	CONVENTION	GOOD EXAMPLES	BAD EXAMPLES

Class	UpperCamelCase: The first letter of each word is capitalized.	`BMICalculator`
		`Student`
		`MyProgram`
			`bmiCalculator`
			`STUDENT`
			`myProgram`

Variable	lowerCamelCase: The first letter is lowercase and the first letters of all following words are capitalized.	`myHeight;`
		`myWeight;`
		`height;`
		`weight;`
			`MyHeight;`
			`myheight;`
			`Height;`
			`WEIGHT;`

Method	lowerCamelCase: The first letter is lowercase and the first letters of all following words are capitalized.	`main` `calculateMyBMI`	`Main` `CalculateBMI`

Java Data Types

TYPE NAME	KIND OF VALUE	MEMORY USED	SIZE RANGE
`boolean`	`true` or `false`	1 byte	Not applicable
`char`	Single character (Unicode)	2 bytes	Common Unicode characters
`byte`	Integer	1 byte	−128 to 127
`short`	Integer	2 bytes	−32768 to 32767
`int`	Integer	4 bytes	−2147483648 to 2147483647
`long`	Integer	8 bytes	−9223372036854775808 to 9223372036854775807
`float`	Floating-point number	4 bytes	$\pm 3.40282347 \times 10^{+38}$ to $\pm 1.40239846 \times 10^{-45}$
`double`	Floating-point number	8 bytes	$\pm 1.76769313486231570 \times 10^{+308}$ to $\pm 4.94065645841246544 \times 10^{-324}$

Java Data Types - Casting

- Can force a data type to act (temporarily) as another

- E.g. int a = (int) 23.7; // a is assigned 23

Java Operators

- Arithmetic:
  - + - * / % (modulo, remainder after division)
- Assignment:
  - = += -= ++ --
- Logical
  - && || ^ !
- Relational
  - < > <= >= == !=

Eclipse IDE

- *Demo given in-lecture*