Workshop1

COMP90041 Programming and software development

Prepared By: Chenyang Lu

Presenter

Chenyang Lu(Luke)

Email: chenyang.lu@unimelb.edu.au

If you have any questions about lecture or workshop, ask me directly during the workshop or put it on Discussion Board.

Day	Time	Duration	Location
Thu	17:15:00	1:00:00	PAR-Elec. Engineering-121
Thu	10:00:00	01:00:00	PAR-207 Bouverie St-B116

Before the workshop

- You are expected to finish all the lab exercise before coming to the lab.

- The lab exercise will not be marked.

- The first 10-15 minutes will be a quick review of what we learnt last week.

- Ask questions and do lab exercise for the rest of time

Review

What is Java?

- Java is an object oriented programming (OOP) language

    - A java program is made up of one or more classes
    - A class contains one or more methods
    - The "main" method is entry point of java program.

println Method

System.out.println("Hello world");

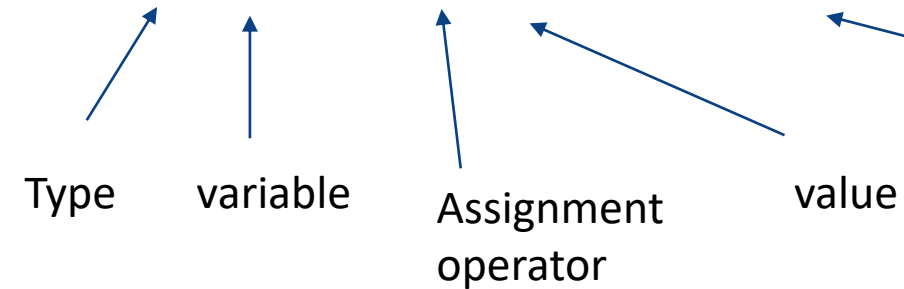System.out: an object

println: a method belong to system.out object

Method is invoked by "."

```java
public class HelloWorld {

    /*
     * This is the main method
     */
    public static void main(String[] args) {
        System.out.println("Hello World!");
        aMethod();



    }


    /*
     * This is the aMethod
     */

    public static void aMethod(){
        int count = 0;
        System.out.println("The value of count is " + count);


    }


    /*
     * This is the bMethod
     */


    public void bMethod(){


    }
}
```

Variable Declaration

int count = 0;

Type   variable   Assignment operator   value

```java
public class HelloWorld {

    /*
     * This is the main method
     */
    public static void main(String[] args) {
        System.out.println("Hello World!");
        aMethod();



    }


    /*
     * This is the aMethod
     */

    public static void aMethod(){
        int count = 0;
        System.out.println("The value of count is " + count);


    }


    /*
     * This is the bMethod
     */

    public void bMethod(){


    }
}
```
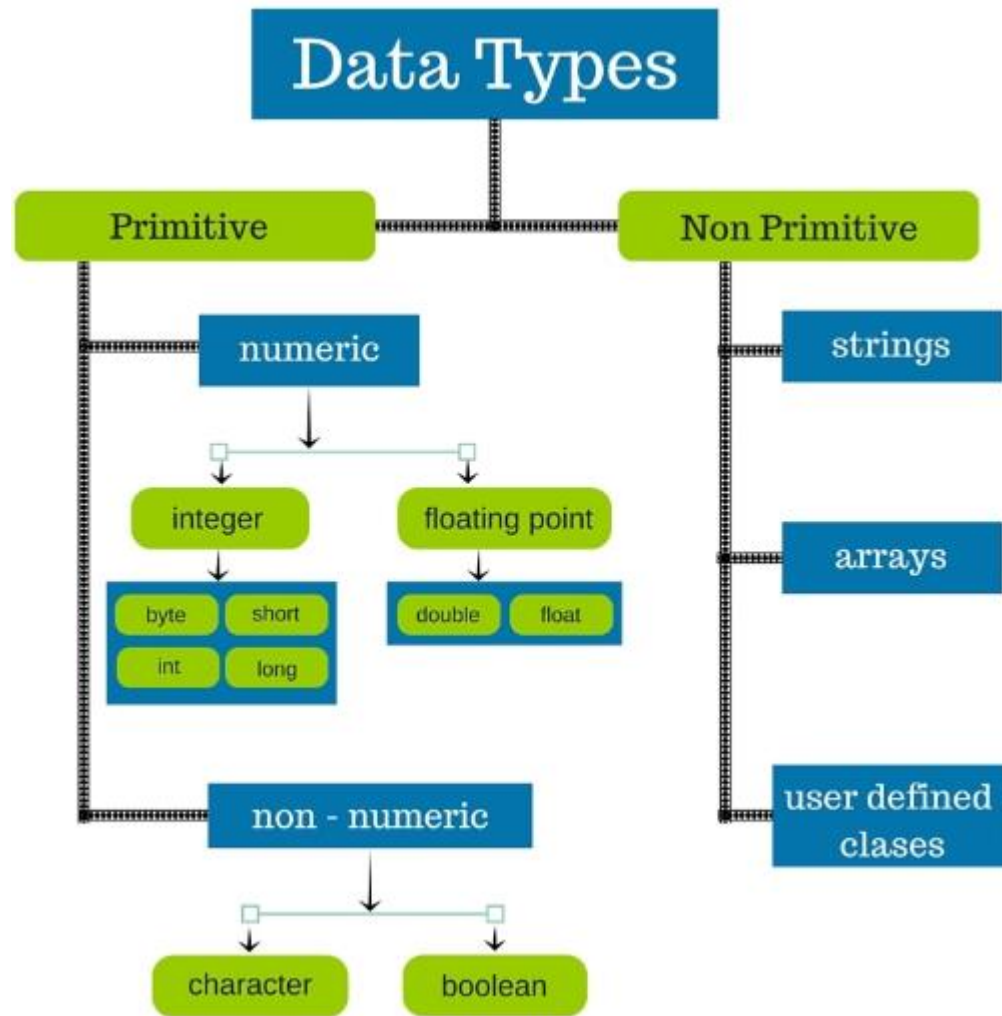
Variable Declaration

- Every variable in a Java program
  must be declared before it is used

- A variable declaration tells compiler
  what kind of data will be stored in
  variables

```java
int a;   // a is an integer
double b;  // b is a double-precision float point number
char c = 'a';   // c is a character and is given a initial value
a = 5;   // you can also assign value in this way
d = 6;   // ERROR! d undefined
```

Data types

```java
public class HelloWorld {

    /*
     * This is the main method
     */
    public static void main(String[] args) {
        System.out.println("Hello World!");
        aMethod();



    }


    /*
     * This is the aMethod
     */

    public static void aMethod(){
        int count = 0;
        System.out.println("The value of count is " + count);


    }


    /*
     * This is the bMethod
     */


    public void bMethod(){


    }
}
```
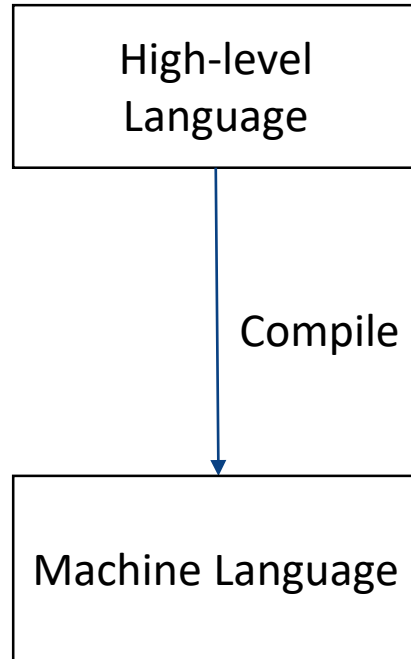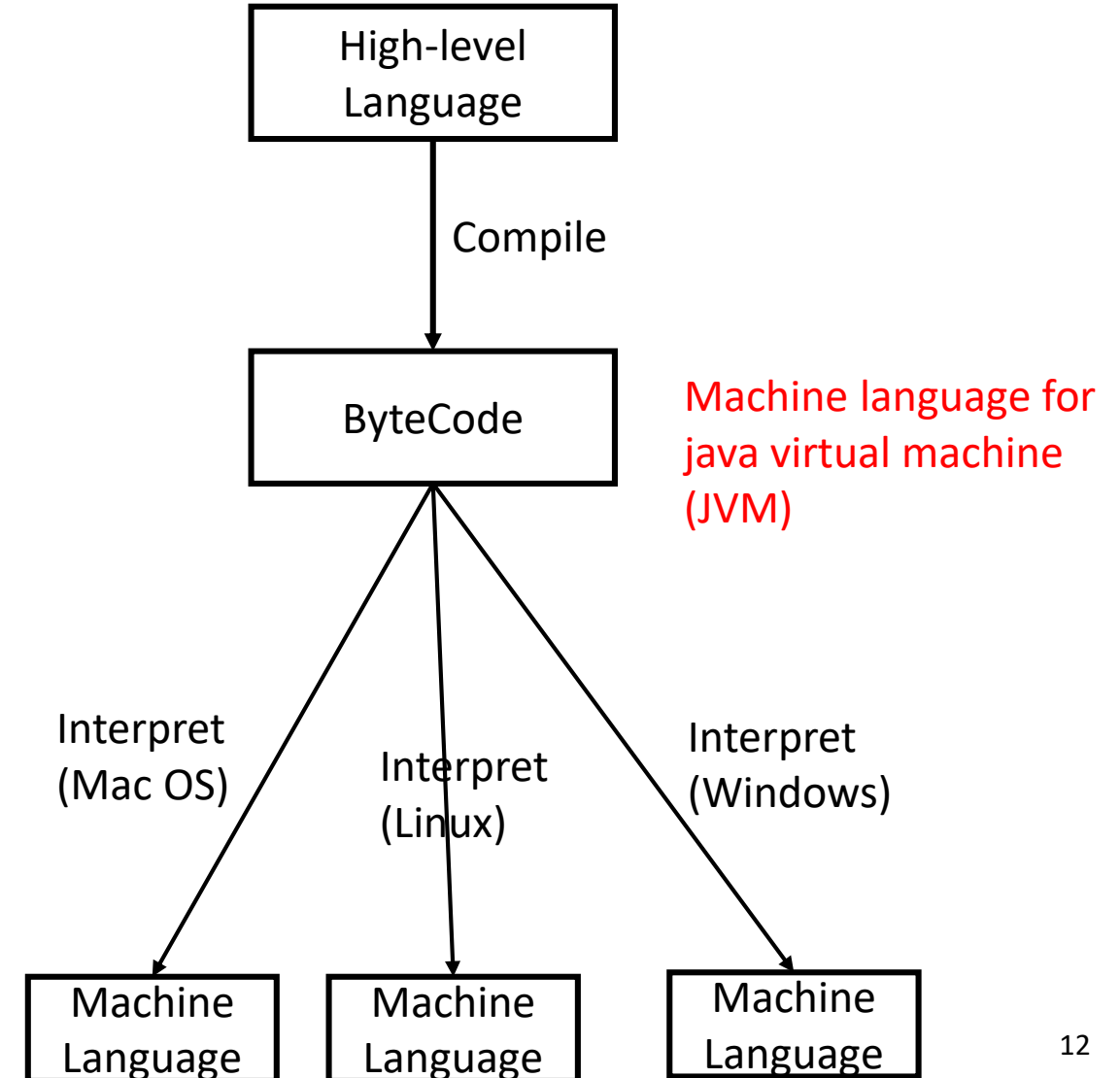
Any question?

Computer Language Level

- High Level Language

- Machine Language

- Low level Language

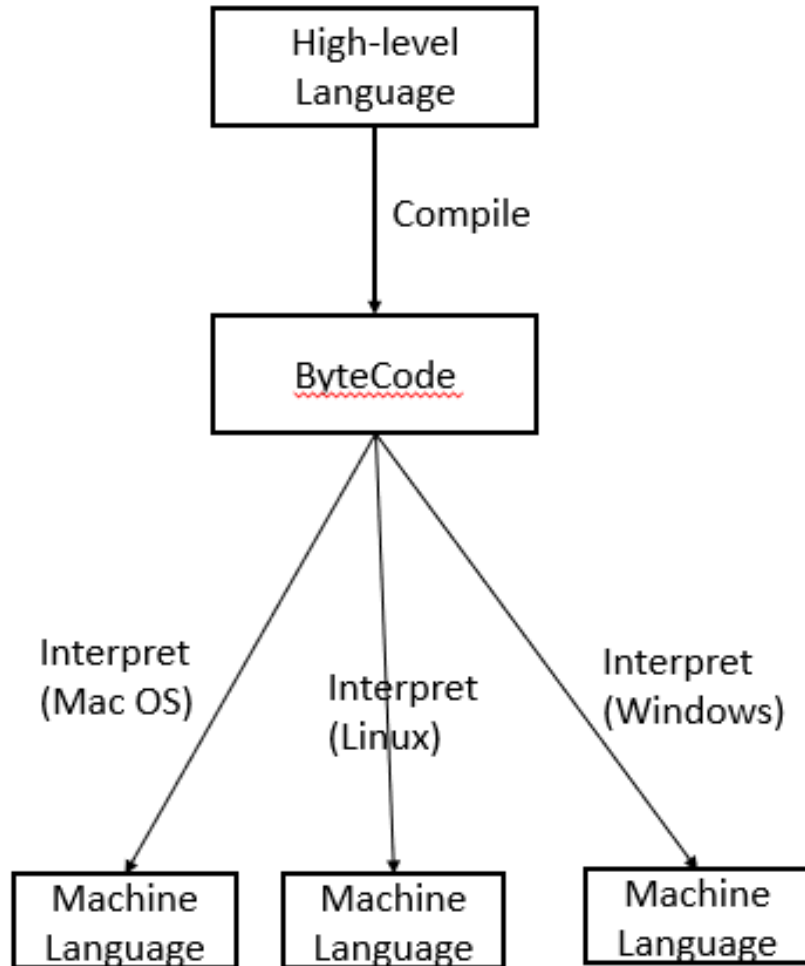# Convert between high-level language and Machine Language

Traditional way (C, C++)

```
┌─────────────────┐
│   High-level    │
│    Language     │
└─────────────────┘
         │
         │ Compile
         ▼
┌─────────────────┐
│ Machine Language│
└─────────────────┘
```

Java

```
┌─────────────────┐
│   High-level    │
│    Language     │
└─────────────────┘
         │
         │ Compile
         ▼
┌─────────────────┐
│    ByteCode     │      Machine language for
└─────────────────┘      java virtual machine
    ╱     │     ╲        (JVM)
Interpret │  Interpret
(Mac OS) Interpret (Windows)
         (Linux)
  ╱       │        ╲
┌────────┐ ┌────────┐ ┌────────┐
│Machine │ │Machine │ │Machine │
│Language│ │Language│ │Language│
└────────┘ └────────┘ └────────┘
```

Convert between high-level language and Machine Language (Java)

1. **Step one: from high-level language to byte-code**
Javac FirstProgram.java =》 FirstProgram.class

2. **Step two: from bytecode to machine Language**
Java FirstProgram

Lab Exercise

Objective

- Follow "Lab2-step-by-step.pdf" to finish lab exercise.

- Set up Java Development Environment
    1. Install JDK and JRE
    2. Install IDE (e.g. Eclipes, IntelliJ)
    3. Try to start a new project in your IDE!