Java: Beginning

Computer Engineering Department

Java Programming Course

Asst. Prof. Dr. Ahmet Sayar

Kocaeli University - Fall 2014

JAVA

- Platform independent
- Compile once run everywhere
- JVM - Java Virtual Machine

Applications and Applets

- two kinds of Java programs: *applications* and *applets*
- applications
  - regular programs
  - meant to be run on your computer
- applets
  - little applications
  - meant to be sent to another location on the Internet and run there

Class Structure

```
package myprojects.javaprojects.prj1;
import java.util.*;

public class test {
    int x;
    int y;

    public static void main(){
        int a;
        int b;
    }
}
```

Program Structure

```
class CLASSNAME {
    public static void main(String[] arguments) {
        STATEMENTS
    }
}
```

First Program

```
class Hello {
    public static void main(String[] arguments) {
        // Program execution begins here
        System.out.println("Hello world.");
    }
}
```

Some Terminology

- The person who writes a program is called the *programmer.*
- The person who interacts with the program is called the *user.*
- A *package* is a library of classes that have been defined already.
  - `import java.util.*`

Some Terminology, cont.

- The item(s) inside parentheses are called *argument(s)* and provide the information needed by methods.
- A *variable* is something that can store data.
- an instruction to the computer is called a *statement*; it ends with a semicolon.
- The grammar rules for a programming language are called the *syntax* of the language.

Printing to the Screen

```
System.out.println ("Whatever you want to print");
```

- `System.out` is an object for sending output to the screen.
- `println` is a method to print whatever is in parentheses to the screen.
- How about System.out.print

- Int var=3;
- System.out.println(var + " time A"); ?

Printing to the Screen

```
class Hello2 {
    public static void main(String[] arguments) {
        System.out.println("Hello world.");  // Print once
        System.out.println("Line number 2");  // Again!
    }
}
```

Simple Input

- Sometimes the data needed for a computation are obtained from the user at run time.
- Keyboard input requires

  ```
  import java.util.*
  ```

  at the beginning of the file.

Simple Input, cont.

- Data can be entered from the keyboard using

```
Scanner keyboard =
        new Scanner(System.in);
```

followed, for example, by

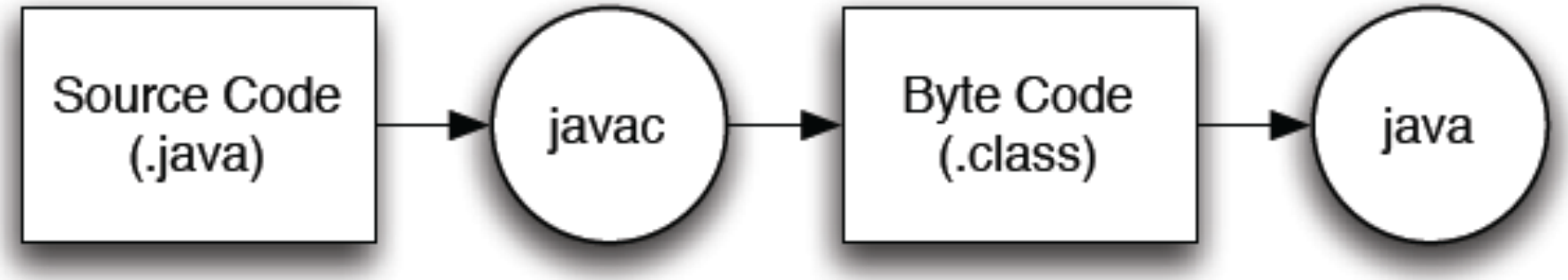
```
int eggsPerBasket = keyboard.nextInt();
double d1 = keyboard.nextDouble();
```

which reads one `int` value from the keyboard and assigns it to `eggsPerBasket`.

Compiling a Java Program or Class

- A Java program consists of one or more classes, which must be compiled before running the program.
- You need not compile classes that accompany Java (e.g. `System` and `Scanner`).
- Each class should be in a separate file.
- The name of the file should be the same as the name of the class.

Compiling



Compiling and Running

- Use an *IDE* (integrated development environment) which combines a text editor with commands for compiling and running Java programs.
- When a Java program is compiled, the byte-code version of the program has the same name, but the ending is changed from `.java` to `.class`.

Compiling and Running, cont.

- A Java program can involve any number of classes.
- The class to run will contain the words

```
public static void main(String[] args)
```

  near the beginning of the file.

Input from Keyboard - 1

```
import java.util.*;

public class FirstProgram
{
    public static void main(String[] args)
    {
        System.out.println("Hello out there.");
        System.out.println("I will add two numbers for you.");
        System.out.println("Enter two whole numbers on a line:");

        int n1, n2;

        Scanner keyboard = new Scanner(System.in);
        n1 = keyboard.nextInt();
        n2 = keyboard.nextInt();

        System.out.println("The sum of those two numbers is");
        System.out.println(n1 + n2);
    }
}
```

Name of the package (library) that includes the `Scanner` *class.*

Name of the program.

Sends output to screen.

Says that `n1` *and* `n2` *are variables that hold integers (whole numbers)*

Sets up things so the program can have keyboard input.

Reads one whole number from the keyboard

Sample Screen Dialog

```
Hello out there.
I will add two numbers for you.
Enter two whole numbers on a line:
12 30
The sum of those two numbers is
42
```

Display 1.5

A Sample Java Program

Input from Keyboard - 2

```
import java.util.*;
```

Name of the package (library) that includes the Scanner *class.*

```
public class EggBasket2
{
    public static void main(String[] args)
    {
        int numberOfBaskets, eggsPerBasket, totalEggs;

        Scanner keyboard = new Scanner(System.in);
        System.out.println("Enter the number of eggs in each basket:");
        eggsPerBasket = keyboard.nextInt();
        System.out.println("Enter the number of baskets:");
        numberOfBaskets = keyboard.nextInt();

        totalEggs = numberOfBaskets * eggsPerBasket;

        System.out.println("If you have");
        System.out.println(eggsPerBasket + " eggs per basket and");
        System.out.println(numberOfBaskets + " baskets, then");
        System.out.println("the total number of eggs is " + totalEggs);

        System.out.println("Now we take two eggs out of each basket.");

        eggsPerBasket = eggsPerBasket - 2;
        totalEggs = numberOfBaskets * eggsPerBasket;

        System.out.println("You now have");
        System.out.println(eggsPerBasket + " eggs per basket and");
        System.out.println(numberOfBaskets + " baskets.");
        System.out.println("The new total number of eggs is "
                                                + totalEggs);
    }
}
```

Sets up things so the program can have keyboard input.

Reads one whole number from the keyboard

Sample Screen Dialog

```
Enter the number of eggs in each basket:
6
Enter the number of baskets:
10
If you have
6 eggs per basket and
10 baskets, then
the total number of eggs is 60
Now we take two eggs out of each basket.
You now have
4 eggs per basket and
10 baskets.
The new total number of eggs is 40
```

Display 2.3

Program with Keyboard Input

Input from Command Line

```
package Lab_1;

public class HelloWorld {

	public static void main (String args[]) {
		System.out.println("Hello World!");
		System.out.println(args[0]);
	}
}
```

Since it is in package Lab_1, it is expected to be located in [project_path]/src/Lab_1/HelloWorld.java

Sample Command Lines Compiling and Running

- [project_path]>javac Lab_1/HelloWorld.java
  - HelloWorld.class is created
  - This is bytecode class or also called target class
- [project_path]>java Lab_1/HelloWorld Ahmet
  - Output: ???
  - Hello World! Ahmet
- [project_path]>java Lab_1/HelloWorld Ahmet Sayar
  - Output: ???
  - How about output of System.out.println(args[2]); ???

Questions?

- What is JVM?
- What is byte code, source code, target code
- Is java portable how?
- What is java SDK?
- What is java JRE?
- Not specifically for java, in general
  - What is linking?
  - What is loading?
- What is compiling (derleyici)?
- What is interpreting (yorumlayici)?
- Is JAVA compiler based or interpreter based programming language? How?

Questions?

- What is the extension of sourcecode java class
- What is the extension of bytecode (targetcode) java class
- What is the command to compile a java class?
- What is the command to run a java class?
- Can java run on every platform?
- What is the name of small java codes embedded in html and run on web environment?
- What is encapsulation?