Advanced Programming

Lab 1

JDK, JRE

- **JDK** = Java Development Kit
  - “tools” for developing, debugging, and monitoring Java applications
  - Oracle JDK, OpenJDK
  - Download
- **JRE** = Java Runtime Environment
  - “tools” required for running a Java application
  - Java Virtual Machine

Integrated development environment (IDE)

- Syntax highlighting
- Code completion
- Refactoring
- Debugging, etc.

- Choices: **NetBeans, Eclipse, IntelliJ**

First Project

- New Project
  - Java with Maven
  - Java with Gradle
  - **Java with Ant** (choose this for now)

- **Ant, Maven, Gradle** are used for automating the creation of a software build and the associated processes: *downloading dependencies, compiling , packaging, testing, deployment*

First Program

HelloWorld.java

```
package lab1;
/**
 *
 * @author Your Name
 */
public class HelloWorld {

    public static void main(String args[]) {
        System.out.println("Hello World!");
        // just write sout and then press TAB
    }
}
```

Compile and Run

- Compiling: **javac**
  - `javac` **`HelloWorld.java`**
  - Produces: **`HelloWorld.class`**

- Virtual Machine: **java**
  - `java` **`HelloWorld`**
  - It runs your application

- ...or press the “Big Green Button”

Creating other methods

Procedural approach

```
public class Lab1 {

    public static void main(String args[]) {
      someMethod();
    }

    public static void someMethod() {
      //Do stuff
    }
}
```

Creating other methods

Object oriented approach

```
public class Lab1 {
    public static void main(String args[]) {
        Lab1 lab1 = new Lab1();
        lab1.compulsory();
    }
    void compulsory() {
        //Do stuff
    }
    void homework() {
        //Do stuff
    }
    void bonus() {
        //Do stuff
    }
}
```

Arrays

- Declaration

```
int[] a; byte b[];
```

100 elements of type char

- Instantiation

```
a = new int[10]; char c[] = new char[100];
```

- Initialization

```
String colors[] = {"Red", "Yellow"};
someMethod( new String[] {"Red", "Yellow"} );
```

- The size of an array

`a.length` **and not** ~~`a.length()`~~

Multi-dimensional Arrays

- Arrays of arrays

```
int[][] m2d = new int[10][20];

int[][][] m3d = new int[10][20][30];
```

- Copying arrays

```
System.arrayCopy

int a[]; int b[]; …  What about a = b?;
```

- Utility methods for arrays

```
java.util.Arrays

   - binarySearch, equals, fill, ...
```

Strings

- char[]

  ```
  char data[] = {'a', 'b', 'c'};
  ```

- String Immutable Object

  ```
  String s = "abc";  String s = "a" + "b" + "c";
  String s = new String("abc");
  String s = new String(data);
  ```

- StringBuilder, StringBuffer

  ```
  StringBuilder sb = new StringBuilder("a");
  sb.append("b").append("c");
  ```

Example

```
public class Problem2 {

    public static void main(String args[]) {
        Problem2 app = new Problem2();
        char alphabet[] = {'A', 'C', 'G', 'T'};
        String word = app.createRandomWord(7, alphabet);
        System.out.println(word);
    }

    private String createRandomWord(int len, char[] alphabet) {
        StringBuilder word = new StringBuilder();
        Random rand = new Random();
        for (int i = 0; i < len; i++) {
            int k = rand.nextInt(alphabet.length);
            word.append(alphabet[k]);
        }
        return word.toString();
    }
}
```

Command Line Arguments

```
public class Problem2 {

    public static void main(String args[]) {
        if (args.length < 3) {
            System.out.println(
              "Usage: number, number, one or more characters");
            System.exit(-1);
        }

        int n = Integer.parseInt(args[0]);
        int p = Integer.parseInt(args[1]);
        int m = args.length - 2;
        char alphabet[] = new char[m];
        for(int i=0; i<m; i++) {
            alphabet[i] = args[i+2].charAt(0);
        }
}
```

Your Project → Properties → Run → Arguments

Notes

- String to int conversion
  - `int n = Integer.parseInt("123");`
- Binary: **0b**1001, Hexa: **0x**FF
- Unicode: **\u**
  - `System.out.println("\u0394\u03B4");` Δδ
- How much memory is needed?
  - `int a[][] = new int[30_000][30_000];`