

## Week 1 Command line and Basic Java Syntax

资源：

1. (从此以后就靠它了) <http://stackoverflow.com>
2. leetcode, lintcode会是我们今后做练习的地方
3. oracle API

## 第一部分 Command Line

这一部分介绍基本的command line操作。我们刚刚接触电脑的时候都会用鼠标在窗口之间拖拽文件，添加移除文件、文件夹。Command Line的一些基本操作也可以完成同样的任务（其实command line先于界面窗口出现）。

Commands: ls pwd cd rm mkdir cat echo \*grep \*cp \*mv \*pushd \*popd

\*这些command有兴趣可以看看

### ls(LiSt)

列举当前文件夹下的文件

ls -l （包含全部信息，e.g.权限...）

ls -a （包含隐藏文件）

### pwd(Produce Working Directory)

输出当前的路径... 什么是路径？

### cd(Change Directory)

切换当前路径

### rm(ReMove)

删除文件 / 文件夹

rm -r

rm -rf (可怕的命令，最好别用)

### mkdir(Make Directory)

生成新文件夹

### cat(conCatenate)

浏览单个 / 多个文件

### echo

逐字输出

## 第二部分

### Basic Java Syntax

#### 0.1 Code Arrangement

层次是, project -> package -> class, 缺一不可 (this is strict).

#### 0.2 Compilation

Different from python, where mostly code is executed by interpretation.  
HelloWorld.java -> “compiler” -> HelloWorld.class -> “JVM” -> execution!!!  
demo见Week1视频

### 1. Variables (变量)

变量有诸多type, 主要分为两类:

#### a. primitive

byte, char, short, int, long  
float, double  
boolean

#### b. class

builtin: 这里强调String和Array。因为String不是primitive, 比较的时候需用 “.equals”  
customised (自定义的)

使用变量之前都要声明它的type (specify the types before using variables).

e.g. int x;  
double x = 10.5;  
String s = “123”;

Scope among classes

private  
public  
protected  
package-private

### Scope among code blocks

### 2. Control flow (控制语句)

If... else

template:

```
if (<condition1>) {  
} else if (<condition2>) {  
} else {  
} // all conditions evaluate to booleans
```

loops

FOR

template 1:

```
for (i = 0; i < someValue; i++) {  
    // do something according to index  
}
```

template 2 (for Arrays and ArrayLists):

```
for (int x : array) {  
    // do something  
}
```

WHILE

template:

```
int var = 0;  
while (var < someValue) {  
    // do something  
    var++;  
}
```

### 3. About OOP (Object Oriented Programming)

\*Encapsulation

A language mechanism for restricting access to some of the [object's](#) components.

说白了：外面不能修改里边的东西，但是可以通过method接触。这样增强了每个class的独立性，这是OOP很重要的一点。

Fields & Methods

Field是object的variable

method是object的function

Constructor

What is a constructor?

相当于python里面的init

Constructor 名字和class name完全一致

一个class可以有多个constructor，但是每个的function signature都不同(也叫overloading)

没有人为constructor的时候，默认有一个清零所有variable的

有认为添加constructor的时候，默认constructor不存在！

equals, hashCode method

## 4. Data structures

### a. Arrays

这里强调，每一个array其实都是一个object。初始化的时候需要说明长度（或者说元素都是什么）。

Syntax:

```
int[] array0 = new int[]{2, 3, 3, 3};
int[] array1 = new int[length1];
String[] array2 = new String[length2];
boolean[] array3 = new boolean[length3];
array3[0] = true;
```

### b. ArrayList

ArrayList不同于array，初始化的时候不需要说明长度。

Syntax:

```
ArrayList list0 = new ArrayList();
ArrayList<Integer> list1 = new ArrayList<>();
list1.add(10);
```

### c. HashMap

Map就相当于一个映射，或者组合，简单理解为pair，类比python的dictionary。

Syntax:

```
HashMap map0 = new HashMap();
HashMap<Integer, String> map1 = new HashMap<>();
map1.put(1, "one");
map2.put(2, "two");
```

### d. \*HashSet

## 5. Special java keywords

### STATIC

- a. 这里的static和“java is statically typed”不是一个意思
- b. 这里讲的static是一个修饰词，专门修饰method和field
- c. static表示，相应的method或者field不依附于object。上升到了class的层次
- d. 为什么main method是static的？  
main method是一切程序运行的起点，它不依附于object
- e. static method里面只能调用static类的东西(variable, class, method...), 而不能调用instance类的

Syntax:

```
class Example {  
    static int attr1;  
    public Example() {  
    }  
    public static int getAttr1() {  
        return attr1;  
    }  
    public static void main(String[] args) {  
        System.out.println(Example.getAttr1());  
    }  
}
```

这个例子说明什么？

课堂演练：

写一个car class。要求：能通过调用一个叫howMany的方法知道曾经有几个object被initialise。

作业：

分别用Python, Java完成leetcode 88题。