Another Program

```
public class Welcome
{
   public static void main(String[] args)
   {
     String[] greeting = new String[3];
     greeting[0] = "Welcome to ‘‘;
     greeting[1]  = "the second class of";
     greeting[3]  = "CS520";

     for (int i = 0; i < greeting.length; i++)
         System.out.println(greeing[i]);
     }
}
```

Setting Execution Path

```
/usr/local2/bin/java
```

Bourne Shell

```
export PATH=/usr/local2/bin:$path
```

Cshell

```
export path=(/usr/local2/bin $path)
```

Setting Class Path

Bourne Shell

```
export CLASSPATH=.
```

Cshell

```
setenv CLASSPATH .
```

Primitive Data Types

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

Basic Programming Structures

Same as C++

- Statements
- Assignments
- Blocks
- Conditional Statements
- While Loops
- Do Loops
- For Loops
- Switch Statement

Classes and Objects

- An Object is defined by its class.
- Classes define methods.
- Each object is an instance of a class.
- A method is invoked on an object.

Strings

```
class StringsDemo {
    public static void main(String[] args) {
      String myName = "Petronius";

      myName = myName + " Arbiter";
      System.out.println("Name = " + myName);
     }
}
```

A Simple Class: Celestial Bodies

```
class Body {
    public long idNum;
    public String name;
    public Body orbits;

    public static long nextID = 0;
}
```

```
Body mercury
```

Class Members

- Fields
- Methods

Creating Objects

```
Body sun = new Body();
sun.idNum = Body.nextID++;
sun.name = "Sol";
sun.orbits = null;// in solar system,
       //sun is middle

Body earth = new Body();
earth.idNum = Body.nextID++;
earth.name = "Earth";
earth.orbits = sun;
```

Constructors

```
class Body {
    public long idNum;
    public String name = "<unnamed>";
    public Body orbits = null;

    public static long nextID = 0;

    Body(){
        idnum = nextID++;
}
```

Constructors (cont)

```
Body sun = new Body();  // idNum is 0
sun.name = "Sol";

Body earth = new Body(); //idNum is 1
earth.name = "Earth";
earth.orbits = sun;
```

Constructors (cont)

```
Body(String bodyName, Body orbitsAround) {
   this();
   name = bodyName;
   orbits = orbitsAround;
 }
```

```
Body(Body other {
    idNum = other.idNum;
    name = other.name;
    orbits = other.orbits;
}
```

Continued

```
class Body {
    public long idNum;
    public String name = "<unnamed>";
    public Body orbits = null;
    private static long nextID = 0;
    {
     idNum = nextID++;}
    public Body(String bodyName,
                Body orbitsAround)
     {name = bodyName;
      orbits = orbitsAround;}
}
```