Scope Rules

- Scope is the term used to describe the visibility of a method or variable.
- Examples
  - Private methods have class scope (they may only be used within the class they are created in)
  - A private instance variable also has class scope and can be used only within the class it's declared in

Local Variable Scope

- The scope of a local variable declared in a method is the method.
  - The variable does not exist outside of the method.

Shadowing

- What will this code print?
  ```java
  class Test
  {
      static int x = 1;
      public static void main(String[] args) {
          int x = 0;
          System.out.println("x=" + x);
          System.out.println("Test.x=", Test.x=" + Test.x);
      }
  }
  ```
  - x = 0, Test.x = 1

Forward Instance Variable Declaration

- Bad code:
  ```java
  class Test
  {
      public static void main(String[] args) {
          System.out.print("x=");
          System.out.println("", Test.x=" + Test.x);
          int x = 0;
      }
  }
  ```
  - Prints out: x=1, Test.x=1

What's wrong here?

```java
class TestVar
{
    int x=0;
    public static void main(String[] args) {
        System.out.print("x=");
        System.out.println("", TestVar.x=" + this.x);
    }
}
```  
- Can instance variable be used in class methods?
  - NO!!!

Arrays

- A variable can hold exactly one value
- An array allows us to group a number of variables, of the same type, under a single name
- Each location in the array is identified by its position, or index, in the array
  - Array indices start at 0 and run to one less than the number of locations in the array
1-D Arrays are like a table with one row

Array of 4 integers

Indices

Ringo  John  George  Paul

Draw It

Draw an array and give the appropriate indices for the list of ages given below:
1. 20
2. 35
3. 5
4. 67
5. 21

The Answer

20 35 5 67 21

Arrays

Arrays are objects but there is no class that array objects are instances of
- This means that array variables are reference types
- Like any reference type, before using an array you must do two things
  1. Declare the identifier that will be used to refer to the array
  2. Create the array

Arrays

Variables of array type are declared using bracket ([ ]) notation:

typename[] identifier;
typename identifier[];

Array Creation

It is possible to directly initialize/create the values of the array elements using an initializer list:

int[] n = {1, 2, 3, 4, 5};
float[] m = {1.3f, 2.4f, 3.5f};
String[] w = {"Tom", "Scary", "Moo Cow"};

Or the array can be created with the new command:

int[] n = new int[5];
float[] m = new float[5];
String[] w = new String[3];
If the array is created with the new command, it should be initialized to some sort of values:

```java
int[] n = new int[5];
for (int i=0; i<5; i++)
    n[i] = i;
```

Create an array of integers for the five ages given in the previous question and then set the values in the array to be the values of the five ages:

```java
int ages[] = {20, 35, 5, 67, 21};
OR
int ages[] = new int[5];
ages[0] = 20;
ages[1] = 35;
ages[2] = 5;
ages[3] = 67;
ages[4] = 21;
```

An array object has an instance variable, `length`, that gives the number of locations in the array:

```java
int anArray[] = new int[10];
System.out.println( anArray.length );
```

Once created, the length of an array is fixed.

Write a class to find the maximum number of an integer array of 5 ages entered at the command line. You must create an integer array to hold the numbers. In order to convert strings to integers you may use the following method:

```java
int Integer.parseInt( String str )
```
class SelfTestArray {
    public static void main(String[] args) {
        int numInArray = args.length;
        int[] numbers = new int[numInArray];
        int maximum;

        for (int i=0; i<numInArray; i++)
            numbers[i] = Integer.parseInt(args[i]);

        maximum = numbers[0];
        for (int i=0; i<numInArray; i++)
            if (maximum < numbers[i])
                maximum = numbers[i];

        System.out.println("The maximum value is: " + maximum);
    }
}

RIT Computer Science Dept.