



THE UNIVERSITY
of NORTH CAROLINA
at CHAPEL HILL

COMP 110

Introduction to Programming

Tuesday November 4, 2014

Jay Aikat

Fall 2014

TR 9:30 - 10:45, GS-G100



Previous Class

- What did we discuss?



Announcements

- **Lab 6 – Due Tue, Nov 11**
- **Reading: 6.1, 6.2, 6.3**

COMP 110 - Fall 2014

3



Pre- and Postcondition Comments

- Precondition comment
 - States conditions that must be true before method is invoked
- Example

```
/**  
Precondition: The instance variables of the calling  
object have values.  
Postcondition: The data stored in (the instance variables  
of) the receiving object have been written to the screen.  
*/  
public void writeOutput()
```

COMP 110 - Spring 2014

4



Pre- and Postcondition Comments

- Postcondition comment
 - Tells what will be true after method executed
- Example

```
/**
 * Precondition: years is a nonnegative number.
 * Postcondition: Returns the projected population of the
 * receiving object after the specified number of years.
 */
public int predictPopulation(int years)
```

COMP 110 - Spring 2014

5



Calling a Method from main

```
public class Exercise1 {
    public static void main(String[] args){
        display();
    }

    static void display() {
        System.out.println("This is an exercise to work with
        methods");
    }
}
```

COMP 110 - Fall 2014

6



Calling a Method that takes input

```
public class Exercise1 {
    public static void main(String[] args) {
        int num1 = 5;
        int num2 = 10;
        int num3 = 15;
        addNumbers(num1, num2, num3);
    }

    static void addNumbers(int n1, int n2, int n3) {
        int result = n1+n2+n3;
        System.out.println("The result is " + result);
    }
}
```

COMP 110 - Fall 2014

7



Calling a Method that returns a string

```
import java.util.*;
public class Exercise1 {
    public static void main(String[] args){
        String s1 = getUserInput();
        System.out.println("Welcome to COMP110, " + s1);
    }

    static String getUserInput() {
        Scanner keybrd = new Scanner(System.in);
        System.out.println("Please input your full name");
        String user1 = keybrd.nextLine();
        return user1;
    }
}
```

COMP 110 - Fall 2014

8



Calling a Method that takes input and returns a value

```
public class Exercise1 {
    public static void main(String[] args) {
        int num1 = 5;
        int num2 = 10;
        int num3 = 15;

        int sum = addNumbers(num1, num2, num3);
        System.out.println("The result is " + sum);
    }

    static int addNumbers(int n1, int n2, int n3) {
        int result = n1+n2+n3;
        return result;
    }
}
```

COMP 110 - Fall 2014

9



Calling a Method from a method

```
import java.util.*;
public class Exercise1 {
    public static void main(String[] args){
        int num1 = 5, num2 = 10, num3 = 15;
        int sum = addNumbers(num1, num2, num3);
        System.out.println("The result is " + sum);
    }
    static String getUserInput() {
        Scanner keybrd = new Scanner(System.in);
        System.out.println("Please input your full name");
        String user1 = keybrd.nextLine();
        return user1;
    }
    static int addNumbers(int n1, int n2, int n3){
        String s1 = getUserInput();
        System.out.println("Welcome to COMP110, " + s1);
        int result = n1+n2+n3;
        return result;
    }
}
```

COMP 110 - Fall 2014

10



Next class (Thu, Nov 6)

- More on methods and classes...