Extra Credit Assignment

• Posted on homework section of web site
• Due February 24
  – You must send me email by Friday, Feb 7 if you are going to work on this assignment
• Everything that you need to know should be on the web site
• Let me know if you have any questions

Javadoc

• Javadoc is a tool for generating web page documentation of Java programs
• Javadoc looks for special comments and tags to produce the documentation
• Special comment /** ... */

Javadoc

• Recommended practice
  – Put Javadoc comment at beginning of class
  – Put javadoc comment at beginning of every method
• Useful tags for methods
  – @param describes a method parameter
  – @return describes the method return value
Javadoc Example

- From Interest.java:

```java
/**
 * Determine the number of years it takes to reach the
 * target amount and print out intermediate results.
 *
 * @param start  the number of dollars that you start with
 * @param target the number of dollars that you would like
 * to have
 * @param rate   the annual interest rate, as a decimal
 *               value. For example, pass 0.075 for 7.5%.
 * @return       the number of years it takes to reach the
 *               target amount at the given interest rate
 */
static int numYearsToTarget(int start,
    int target,
    double rate) {
```

Javadoc Example

Javadoc Example

Running Javadoc

- Javadoc Interest.java
  - Produces web pages describing the class Interest
  - Only includes methods with the `public` keyword
- Javadoc -private Interest.java
  - Includes all methods
- See Interest.java and javadoc example on course web site
Quiz on Friday

- Covers chapter 4 topics, including
  - Methods
    - definition
    - invocation
  - Parameter passing
    - pass-by-value
      - using expressions and method calls as arguments
  - Return types
    - void
    - return statement
  - Variable scope

Quiz on Friday

- Covers chapter 4 topics, including
  - Method overloading
    - signature matching
  - Recursion
  - Math.random()
    - conversion to integer range
- Make sure you can answer the review questions, starting on page 133.

Example Questions

- Write a method isOdd that takes an integer parameter and returns true if the parameter value is odd, false if it is even.
Example Questions

• Using line numbers, what is the scope of a, i, j, m, and n in the following program?

```java
1  class Scope2 {
2    public static void main( String[] args ) {
3      int a = 5;
4      int j;
5      j = mystery(a);
6      System.out.println("The mystery value is "+j);
7    }
8
9    static int mystery( int n ) {
10      int m = 0;
11      for (int i = 0; i <= n; i++ )
12        m = m + n;
13      return m;
14    }
15  }
```

Example Questions

• What is printed by the following program?

```java
class Mystery {
  public static void main( String[] args ) {
    int m;
    m = mystery( 10 );
    System.out.println("m = "+m);
  }

  static int mystery( int x ) {
    if ( x <= 1 )
      return 1;
    else {
      return ( x + mystery( x/2 ) );
    }
  }
}
```

Example Questions

• What is printed by the following program?

```java
class WhoKnows {
  public static void main( String[] args ) {
    int m;
    m = foo( foo( 3, 4 ) );
    System.out.println("m = " + m);
    m = foo( foo( 5, foo( 2 ) ) );
    System.out.println("m = " + m);
  }

  static int foo( int x ) {
    return x * x;
  }

  static int foo( int x, int y ) {
    return x * y;
  }
}
```
Example Questions

• What is printed by the following program?

class WhatIsIt {
    public static void main(String[] args) {
        int a = 3;
        double b = 2.5;
        byte d = 2;
        long e = 5L;
        System.out.println( x(a) );
        System.out.println( x(b) );
        System.out.println( x(a + d) );
        System.out.println( x(e) );
    }

    static int x(int a) {
        return a * 2;
    }

    static double x(double a) {
        return a * 3.0;
    }
}

Example Questions

• What is printed by the following program?

class Hmmm {
    public static void main(String[] args) {
        double x = 2.0;
        double y = 1.0;
        System.out.println( doIt(y, x) );
    }

    static double doIt(double x, double y) {
        return x / y;
    }
}

Example Questions

• Recall that Math.random() returns a double with a value ranging from 0.0 upto but not including 1.0. Complete the following fragment so that i has a value ranging from 1 to 20, such that all values of i are equally likely.

double r;
int i;
r = Math.random();
i =
Any Questions

• Any more questions about chapter 4?

Chapter 5: Arrays

• What if you had a bunch of quiz scores that you wanted to store and print?
  – You could have a variable for each one
    int quizScoreCrystal;
    int quizScoreWilliam;
    int quizScoreEric;
    int quizScoreMallory;
  – This is not very simple
    • What if you wanted to sort them?
    • What if you wanted to print them?

Array

• An array allows you to store many values using a single identifier
  int[] quizScore;
• Each element of the array can be accessed individually using its index or subscript.
  quizScore[0] - the first element
  quizScore[1] - the second element
  quizScore[i] - the i+1th element
Arrays in memory

```
int x;

int x;
```

Array Declaration

- Use [] to declare array variables
  ```
  int[] quizScore;  // declares an array of ints
  double[] temperature;  // declares an array of doubles
  String[] lastName;  // declares an array of Strings
  ```
- The declaration does not create or initialize the array.
  ```
  quizScore[0]  // the first element
  quizScore[1]  // the second element
  quizScore[i]  // the i+1\textsuperscript{st} element
  ```

Create the array

- After you have declared an array variable, you need to create the array
  ```
  arrayvariable = new type [ length ];
  ```
- This creates an array of the given type with length elements.
- arrayvariable is assigned to reference the newly created array.
Examples

• Create an array reference and an array of 65 ints.
  ```java
  int[] quizScore;
  quizScore = new int[65];
  ```
  – This elements of this array will be indexed from 0 to 64.
  – The elements will be initialized with the value 0.

• Create an array reference and an array of count doubles
  ```java
  double[] temperature;
  temperature = new double[count];
  ```

Examples

• You can create the array reference and the array in one statement:
  ```java
  int[] quizScore = new int[65];
  double[] temperature = new double[count];
  boolean[] isFemale = new boolean[100];
  ```