

Last Time



- ❖ We covered:
 - Basic GUI components
 - Layout managers
 - Methods
 - Random numbers
 - Constants

2/11/04

CS360 Client/Server
Programming Using Java

1

Multiplication Program



- ❖ Write a program that will assist children in learning multiplication of single digit numbers. Your program should display a text label asking the children the question, a text field for the user to enter the result, a button to submit the answer and a label displaying if the answer was correct or incorrect
- ❖ If the answer was incorrect, the user should be asked the same question again, otherwise the numbers change

2/11/04

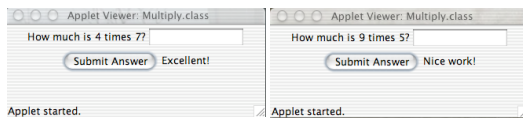
CS360 Client/Server
Programming Using Java

2

Multiplication Program



- ❖ The message displayed after the user has submitted an answer should randomly choose from a list of different messages, so that the user doesn't get bored



2/11/04

CS360 Client/Server
Programming Using Java

3

Fields



- ❖ Fields are variables that have been declared inside a class but outside of any method
- ❖ These fields are accessible to all methods
- ❖ Called *instance variables* in other programming languages

2/11/04

CS360 Client/Server
Programming Using Java

4

Scope of Declarations



- ❖ The scope of a parameter declaration is the body of the method in which the declaration appears
- ❖ The scope of a local variable declaration is from the point at which the declaration appears in the block to the end of that block
- ❖ The scope of a local variable declaration that appears in the initialisation of a **for** statement's header is the body of the **for** statement

2/11/04

CS360 Client/Server
Programming Using Java

5

Scope of Declarations



- ❖ The scope of a method or field of a class is the entire body of the class
- ❖ If a local variable or parameter in a method has the same name as a field, the field is "hidden" until the block terminates executing
- ❖ This is called shadowing

2/11/04

CS360 Client/Server
Programming Using Java

6

What is the Output?



2/11/04

CS360 Client/Server
Programming Using Java

7

Method Overloading



❖ Several methods of the same name can be declared in the same class as long as they have different parameters

❖ This is called method overloading

❖ Example:

```
> Public int square( int x ) { }  
> Public double square( double x ) { }
```

2/11/04

CS360 Client/Server
Programming Using Java

8

Object-Based Programming



- ❖ Classes are used to encapsulate data and methods
- ❖ Encapsulation enables information hiding although objects can communicate across interfaces
- ❖ Procedural programming is action oriented and the unit of programming is the function
- ❖ In Java, the unit of programming is the class

2/11/04

CS360 Client/Server
Programming Using Java

9

Object-Based Programming



- ❖ Procedural programmers concentrate on the verbs
- ❖ Object oriented programmers concentrate on the nouns

2/11/04

CS360 Client/Server
Programming Using Java

10

Abstract Data Types (ADT)



- ❖ ADTs hide their implementation from their clients
- ❖ Write a class time that will store the time in hours, minutes and seconds. The class will contain a constructor that sets all the data to zero, a setTime method to allow the client to set the time, and two methods to output the time in universal and decimal time.

2/11/04

CS360 Client/Server
Programming Using Java

11

```
public class Time1 extends Object {  
    private int hour;    // 0 - 23  
    private int minute; // 0 - 59  
    private int second; // 0 - 59  
  
    public Time1()  
    {  
        setTime( 0, 0, 0 );  
    }  
}
```

2/11/04

CS360 Client/Server
Programming Using Java

12



```
public void setTime( int h, int m, int s )
{
    hour = ( ( h >= 0 && h < 24 ) ? h : 0 );
    minute = ( ( m >= 0 && m < 60 ) ? m : 0 );
    second = ( ( s >= 0 && s < 60 ) ? s : 0 );
}
```

2/11/04

CS360 Client/Server
Programming Using Java

13



```
public String toUniversalString()
{
    DecimalFormat twoDigits =
        new DecimalFormat( "00" );

    return twoDigits.format( hour ) + ":"
        + twoDigits.format( minute ) +
        ":" + twoDigits.format( second );
}
```

2/11/04

CS360 Client/Server
Programming Using Java

14



```
public String toStandardString()
{
    DecimalFormat twoDigits =
        new DecimalFormat( "00" );

    return ( hour == 12 || hour == 0 ) ? 12 : hour % 12 )
        + ":" + twoDigits.format( minute ) + ":" +
        twoDigits.format( second ) +
        ( hour < 12 ? " AM" : " PM" );
}
```

2/11/04

CS360 Client/Server
Programming Using Java

15

Testing the Class



- ❖ To test the class that we just created, we need to create a new class that contains the main method.
- ❖ How are objects created from a class?
- ❖ How are methods called using the objects?

2/11/04

CS360 Client/Server
Programming Using Java

16



```
public class TimeTest1 {
    public static void main( String args[] )
    {
        Time1 time = new Time1();

        String output = "The initial universal time is: "
            + time.toUniversalString() + "\nThe initial
            standard time is: " + time.toStandardString();

        time.setTime( 13, 27, 6 );
        output += "\n\nUniversal time after setTime is: "
            + time.toUniversalString() + "\nStandard time
            after setTime is: " + time.toStandardString();
    }
}
```

2/11/04

CS360 Client/Server
Programming Using Java

17



```
time.setTime( 99, 99, 99 );
output += "\n\nAfter attempting invalid settings: "
    + "\nUniversal time: " + time.toUniversalString()
    + "\nStandard time: " + time.toStandardString();

JOptionPane.showMessageDialog( null, output,
    "Testing Class Time1",
    JOptionPane.INFORMATION_MESSAGE );

System.exit( 0 );
} // end main
```

2/11/04

CS360 Client/Server
Programming Using Java

18

Information Hiding



- ❖ Information hiding is accomplished in object-oriented programming through the use of the `private` keyword
- ❖ Variables (fields or data members) and methods that have been declared as being private are not accessible to the clients outside of the class

2/11/04

CS360 Client/Server
Programming Using Java

19

Information Hiding



```
Time1 time = new Time1( );  
  
time.hour = 9;  
time.minute = 25;  
time.second = 0;
```

- ❖ These declarations are incorrect and the error will be caught by the compiler

2/11/04

CS360 Client/Server
Programming Using Java

20

The *this* Reference



- ❖ Every object can access a reference to itself with the keyword `this`
- ❖ The `this` reference can be used implicitly and explicitly to refer to the instance variables and methods of the object on which the method was called

2/11/04

CS360 Client/Server
Programming Using Java

21

The *this* Reference



- ❖ Can you think of an example where the `this` reference could be used?

2/11/04

CS360 Client/Server
Programming Using Java

22

Constructors



- ❖ What is a constructor?
- ❖ What is the syntax of constructors?

2/11/04

CS360 Client/Server
Programming Using Java

23

Constructors



- ❖ Recall from CS250 that classes can have many constructors
- ❖ In Java these are called overloaded constructors
- ❖ These constructors are differentiated by the number of arguments

2/11/04

CS360 Client/Server
Programming Using Java

24

Program



- ❖ Let's review some of the GUI components that we talked about on Tuesday
- ❖ Write a Java applet that will use the time class that we have created in class. The applet should include text fields to allow the user to enter in the hour, minute and second, and a button that will move the time forward by one second

2/11/04

CS360 Client/Server
Programming Using Java

25

Time Program



- ❖ What GUI widgets do we need?
- ❖ Which of these widgets need to handle events from the user?
- ❖ What methods will we need in the program?

2/11/04

CS360 Client/Server
Programming Using Java

26

Summary



- ❖ Today we covered:
 - Scope of variables and methods
 - Classes
 - Constructors
 - Information hiding
 - The this reference
- ❖ From the book:
 - We have completed chapters 1 through 7
 - We are half way through chapter 8, up to section 8.9
- ❖ The assignment is due on Tuesday

2/11/04

CS360 Client/Server
Programming Using Java

27