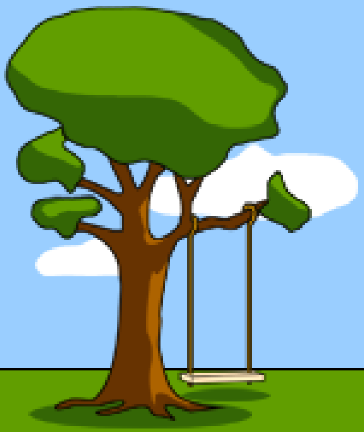
Digital Skills Academy

FUNDAMENTALS OF PROGRAMMING

LAB 4 – PROBLEM SHEET



http://www.projectcartoon.com/cartoon/2

Conor O'Reilly

Development Approach

- **STAGE 1: Before writing code your code design should be documented in the eLabbook as follows:**
- Problem Definition
  - What is the objective
  - What is the program to do
- Design
  - Draw a picture of the execution steps
  - Write down in words the execution steps
  - Draw the design Object or Class diagrams
- Test Cases (how will you test it)
  - Write what you will use for testing that it runs and creates the right answer

• Test Case 10 : 1 + 1 = 2	Simple Case
• Test Case 20: 5 + 9 = 14	Normal Case
• Test Case 30: 0 + 9 = 9	Edge condition
• Test Case 40 : 5 + 0 = 5	Edge condition

- *If this was division we could have division by zero issues and very small answers*

• Test Case 50 : 166666666666 + 788777777777777 = 788944444444443	test the very big

- **STAGE 2: Once you have documented your approach you should proceed to do the following:**
- Write Code
  - Step by step, on piece of functionality at a time, get it working, save a copy of that working version ***addTwoNumbers_v1.code*** in your _Attic directory, add the next bit of functionality
- Test Code with test cases
  - Debug the code, change ONLY ONE thing at a time, KEEP SAVING VERSIONS

- **STAGE 3: The code once written should be documented in the eLabBook under the following headings:**
- Code
  - Insert the code into the eLabBook
- Screens
  - Take snapshots of the program screens and copy them into the eLabBook
- Test Records
  - Records of the tests you performed and the results
- documentation
  - How to use the program documentation
  - Object and class diagrams showing the implementation
- References
  - Any websites or code you looked up or used in the creation of your program

PROBLEM 4.1

Develop a program that calculates the area of a Rectangle

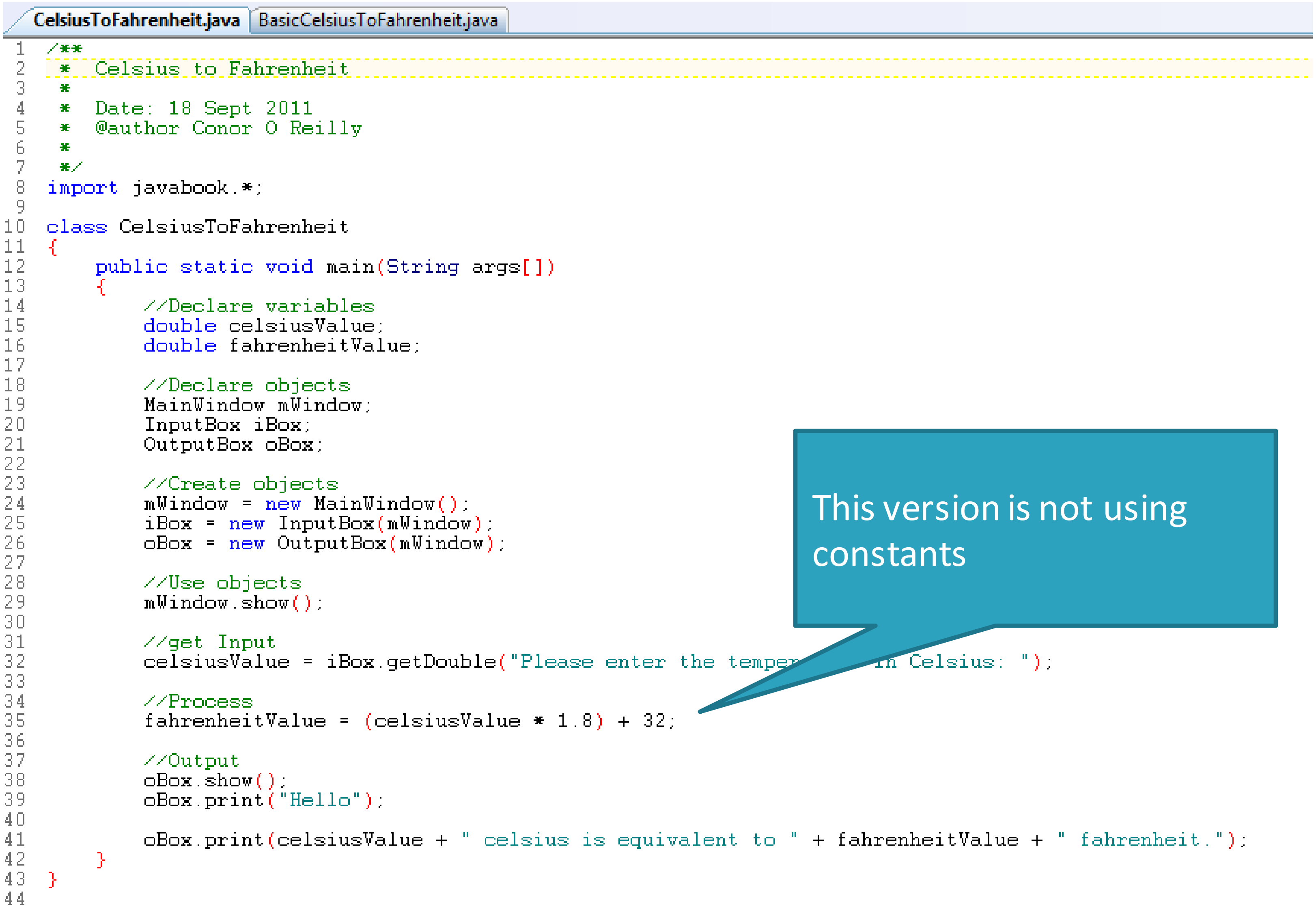
area = length * breadth

Use the javabook class library InputBox class to ask the user for the length and breath. Display the results using the OutputBox class.

Objectives

- Understand and use numerical data types and the associated operators.
- Be able to write arithmetic expressions in Java.
- Be able to write programs that input and output data using the Javabook InputBox and OuputBox classes.

Problem 4.1 Notes



```java
/**
 *  Celsius to Fahrenheit
 *
 *  Date: 18 Sept 2011
 *  @author Conor O Reilly
 *
 */
import javabook.*;

class CelsiusToFahrenheit
{
    public static void main(String args[])
    {
        //Declare variables
        double celsiusValue;
        double fahrenheitValue;

        //Declare objects
        MainWindow mWindow;
        InputBox iBox;
        OutputBox oBox;

        //Create objects
        mWindow = new MainWindow();
        iBox = new InputBox(mWindow);
        oBox = new OutputBox(mWindow);

        //Use objects
        mWindow.show();

        //get Input
        celsiusValue = iBox.getDouble("Please enter the temper       in Celsius: ");

        //Process
        fahrenheitValue = (celsiusValue * 1.8) + 32;

        //Output
        oBox.show();
        oBox.print("Hello");

        oBox.print(celsiusValue + " celsius is equivalent to " + fahrenheitValue + " fahrenheit.");
    }
}
```

This version is not using constants

Conor O'Reilly

PROBLEM 4.2

Develop a program that converts temperatures from Celsius to Fahrenheit

Fahrenheit = 1.8 * Celsius + 32

Where 1.8 and 32 are defined as constants

Use the javabook class library InputBox class to ask the user for the Celsius value to be converted. Display the results using the OutputBox class.

Objectives

- Be able to implement constants
- Be able to write arithmetic expressions in Java with constants.
- Understand the input, processing and output process

Conor O'Reilly

PROBLEM 4.3

Develop a program that calculates the area of a Circle

area = Πr^2

Use the Maths library

- To obtain a value for Π
- To calculate the power of r^2 using the pow() function
- Use the javabook class library InputBox class to ask the user for the radius of the circle. Display the results using the OutputBox class.

Objectives

- Understand the use of the Maths library
- Lookup online the functionality provided by the Maths class
- http://docs.oracle.com/javase/6/docs/api/java/lang/Math.html
- http://www.functionx.com/java/Lesson17.htm

Conor O'Reilly

PROBLEM 4.4

Develop a program that asks the user to input 3 numbers and then returns the average of the 3 numbers

- **Use the javabook class library InputBox class to ask the user for each number.**
- **Display the results using the OutputBox class.**

Objectives

- Understand the nature of objects of a class

PROBLEM 4.5

Develop a program that when run simulates the throwing of a dice

- **Hints**
  - int diceA;
  - final int NUMBER_OF_SIDES = 6;
  - diceA = 1 + (int) (Math.random() * NUMBER_OF_SIDES);
- **Display the results using the OutputBox class.**

Objectives

- Understand random number generation
- Use the Maths.random function and understand how to scope the range of numbers generated by the random function
- Implement explicit casting