Lab 2 | Introduction to Computer Science ICSI201 Fall 2012

Preliminary--Get Help in Lab

Get your lab and project directory set up:

1. You MUST have done the expanding of the BookClasses archive and the setting of DrJava's EXTRA CLASSPATH (directed in Lab 01) for your new work to be successful. If you did not, consult a TA or neighbor for help as soon as possible.
2. Make an new, empty directory (or folder) just for CSI201/Project2.

   Eventually, your project work MUST be uploaded to the Project 2 item of Course Content on Blackboard **in the form of exactly one (1) archive file** (a .zip or equivalent) that expands to one (1) directory (or folder) that contains all the files from your lab work. They must include **`Turtle.java`** which is your final version for the TAs to read, run and base the your program functionality part of the grade. Pros call it a "release".
3. Copy G&E's **`Turtle.java`** file into your new, empty Project2 directory (or folder).

   (An easy way to do this within DrJava is to invoke **Open**, navigate to your **`BookClasses`** directory, find and open **`Turtle.java`**, and **IMMEDIATELY do Save as ... INTO your Project2 directory**.)
4. ***Compile*** it, ***run*** it, and verify that G&E's **`main( )`** method makes one **`Turtle`** draw one vertical line of default length. (You might have to bring forward the window hidden behind DrJava to see that.)
5. Modify the **`static main( )`** method in your copy of **`Turtle.java`** so it draws something different, then compile and run to verify the effects of your modifications.

Lab Activity

Summary: Reproduce golf club drawing examples like those from from lecture AND THEN do golf club drawing WITH METHODS YOU ADD to the `Turtle`. How to make your own methods and use them is introduced below!

1) Straight line code.

Here is the straight line code for drawing a golf club, with head weight 5, and leaving the Turtle in the same **state** as it started: MAKE SURE IT ACTUALLY DRAWS ONE CLUB; show your TA!

```
tu.forward( 100 );
tu.turn(30);
tu.setPenWidth( 5 );
tu.forward( 10 );
tu.forward( -10 );
tu.setPenWidth( 1 );
tu.turn( -30 );
tu.forward( -100 );
```

Below is the one line of code to make the **`Turtle`** referred to by **`tu`** turn 45 degrees, to prepare for the next club:

```
tu.turn( 45 );
```

2) Code with a loop on a counting variable.

Here is a very clear way to program a loop. Variable **NumToDraw** keeps a count of how many more clubs to draw. The loop to draw 7 clubs is:

```
//All this is WITHIN the { ... } of the main method.
int NumToDraw;
NumToDraw = 7;
//Here is BEFORE THE LOOP.
while(NumToDraw > 0)
{
 //Here is INSIDE THE LOOP'S BODY.
 //Put the code to draw one club here.
 //Put the code to turn the Turtle here, below it.
 NumToDraw = NumToDraw - 1; //Count off the last club drawn!
}
```

Reproduce drawing 7 golf clubs.

3) Make the head weight variable, by making a variable named weight that stores the weight.

```
   int weight; //declare BEFORE the loop.

   ...

while(NumToDraw > 0)
  {
    weight = 5;

    tu.forward( 100 );
   tu.turn(30);
    tu.setPenWidth( weight );
   tu.forward( 10 );
   tu.forward( -10 );
   tu.setPenWidth( 1 );
   tu.turn( -30 );
   tu.forward( -100 );

   tu.turn( 45 );

   NumToDraw = NumToDraw - 1;

}
```

First, reproduce drawing the 7 clubs as before, with the **weight** a variable whose value stays at **5**.

Second, make the value of **weight CHANGE** instead of setting it to 5 over and over. The java statement that makes the value stored in the variable **weight** change has the form:

weight = expression for new value, like **weight + 2 ;**

Make it draw 7 clubs with 7 different weights! If you get an undefined variable error message, make sure to set weight to an initial value ABOVE the looping code.

4) Make and use a club drawing method.

For a little while, you will edit the Turtle.java file contents that are OUTSIDE the main method you had been working in.

Write the code below ***in the space*** just above **`public static void main`** , to add to the **`Turtle`** class the method to give every **`Turtle`** the new potential behavior to draw a golf club:

```
public void drawClub()

{

    this.forward( 100 );  //Think: Which Turtle is commanded to draw? THIS one!
    this.turn(30);        //Just copy the code from main, and then replace tu
    this.setPenWidth( 5 );//with this.  Restore 5 for the width instead of the
    this.forward( 10 );   //the variable width.
    this.forward( -10 );
    this.setPenWidth( 1 );
    this.turn( -30 );
    this.forward( -100 );

    return ;

}
```

Be smart and fast: Copy and paste the code from inside the loop, then modify it.

Now, instead of the straight line code to draw a club inside the `{ ... }` of the loop, just write in a method call into the body of your loop. Remember to remove the old drawing operations, which will now be taken from your new method.

```
while(NumToDraw > 0)
  {
     weight = 5;

     tu.drawClub( );  //ONLY 1 line instead of 8!

    tu.turn( 45 );

    NumToDraw = NumToDraw - 1;

}
```

Make it work: You should see 7 clubs, all the same weight.

5) Make and use a second club drawing method with a head weight parameter.

Type in another method definition as follows, on the next page:

```
public void drawClubP(int weight)

{

    this.forward( 100 );
    this.turn(30);
    this.setPenWidth( weight ); //Instead of 5!
    this.forward( 10 );
    this.forward( -10 );
    this.setPenWidth( 1 );
    this.turn( -30 );
    this.forward( -100 );

    return ;

}
```

And inside the loop, code

`tu.drawClubP(` *your expression for different weights* **`);`**

instead of **`tu.drawClub();`** It's a method call with a parameter!

Make it work: Get help from the TAs and neighbors as needed.

When it works to draw 7 clubs with different weights, you are ready to upload the lab work for full credit!

Outside of lab, find out how to make the archive file of your Project2 directory. Then attach it to upload it to Lab02 under Course Material on Blackboard. Your archive must contain a .class file plus , plus the **`Turtle.java~`** backup file that DrJava makes automatically plus a good **`Turtle.java`** file for full credit!

This assignment is due uploaded to Blackboard by Wednesday, September 19, at 11:59PM. But it is smart to get it done much sooner!

AFTER LAB: FIND AND READ WHAT'S WRITTEN IN THE TEXTBOOK ABOUT METHODS. THE NEW CONCEPTS WILL BE EXPLAINED, ILLUSTRATED AND DISCUSSED A LITTLE AT A TIME, AS YOU NEED THEM.