Lab 3 – While Loops, Strings, the Scanner, and Methods

Aim

This lab aims to introduce you to While loops, the Scanner object, and using methods. This material builds on material from weeks 1 and 2, so it is useful to have completed these weeks first. This practical will assume you have more knowledge of Java, and therefore, you will need to use your weeks 1 and 2 material

Tips:

1. If you have not completed the checkpoint from last week, get this checked off first

2. Some of these tasks may be easier if you work with a classmate.

3. Ensure you are saving on your network folder!

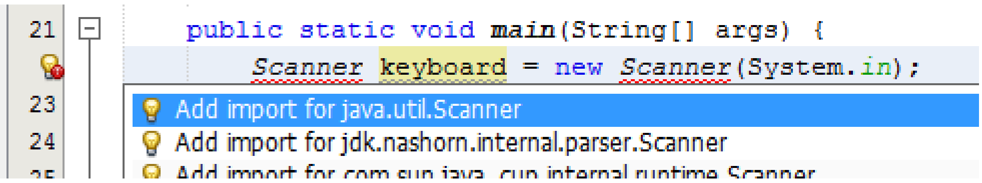
Introduction to the Scanner

The Scanner object is a useful way to access characters that you type.

- Create a new Java Project, call it lab3-scanner, and create a new java class ScannerInput, with a main method.

- Using your lecture notes, add a new Scanner object, named keyboard

- This may cause an error in Netbeans, click on the small lightbulb to the left of the line to find out more about the error:



```
    public static void main(String[] args) {
        Scanner keyboard = new Scanner(System.in);
```

Add import for java.util.Scanner
Add import for jdk.nashorn.internal.parser.Scanner
Add import for com.sun.java_cup.internal.runtime.Scanner

- The error here is because the Scanner object requires an external library to be used, i.e. something outside the standard Java code. Click the “Add import for java.util.Scanner” option.

- Check the top of your Java class. You should see that the line:

  ```
  import java.util.Scanner;
  ```

  has been added to your code. Every time you need to use a Java library, you will need to import it before you can use it

- To use the Scanner successfully, it is first useful to add a prompt to display a message. So below your Scanner creation, add a System.out, asking “What is your name?”

- To then receive the input from the scanner, you need to receive the input. Using your lecture notes, create a String variable called "input", and initialise it to `keyboard.next()`. Here, the the output of the scanner is being assigned to a variable.

- When you run the program, you should be able to click on the console window and then type your name.

- Add a System.out to then display your name, so the final console output should look like:

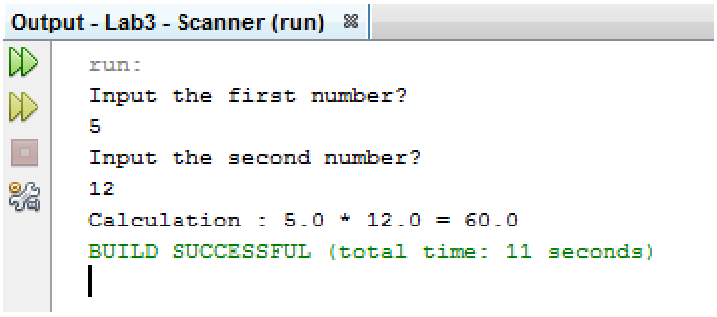
```
What is your name?
Yang
Your input was :Yang
```

- Try to input your full name (given and family name – i.e. Andrew Abel). Does it display correctly? If not, what is happening?

- `keyboard.next()` only reads up to the first space (it tokenizes the data, which will be covered in more depth later in the semester). Try replacing the `keyboard.next()` with `keyboard.nextLine()`. Test it. Do you understand the difference?

Further Use of the Scanner

- Sometimes, you may want to user a numeric value than a String value. Comment out **but do not delete your code**, and create a scanner input that will receive 2 double values, multiply them, and display the result.
    - o **HINT use .nextDouble() or .nextInt()**
- Your console result should look like this:



```
run:
Input the first number?
5
Input the second number?
12
Calculation : 5.0 * 12.0 = 60.0
BUILD SUCCESSFUL (total time: 11 seconds)
```

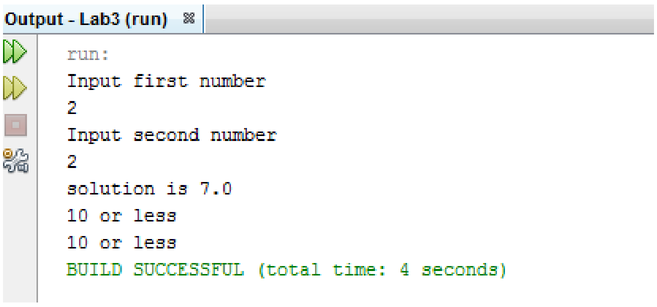
More Calculations

This semester, you have been creating a "Calculations" class, including showing different solutions, generating random numbers, thresholds, and areas of a circle. You should have a copy of this, with some of the code commented out! We will modify some of this

- Copy your completed week 2 calculations class into your netbeans project. You may need to change the main class (see week 1) configuration to run it successfully

- In week 2, for the checkpoint, you had to generate 2 random numbers and check they were over or under a threshold. Change your program so that you use keyboard input

- Add a System.out to then display prompts so the final console output should look like:



```
run:
Input first number
2
Input second number
2
solution is 7.0
10 or less
10 or less
BUILD SUCCESSFUL (total time: 4 seconds)
```

- Do the same for your area of a circle code. Ask the user to input a radius. You should now have 2 calculations with keyboard input

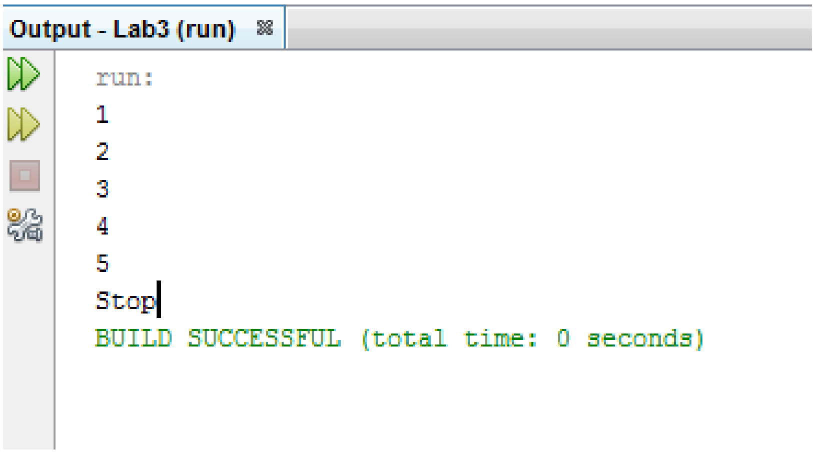
While Loops - Introduction

If we want to repeat something a number of times, a loop is very useful. This week, we will introduce the while loop.

- Create a new class called WhileClass. Create a main method, and change the configuration so it is the default class

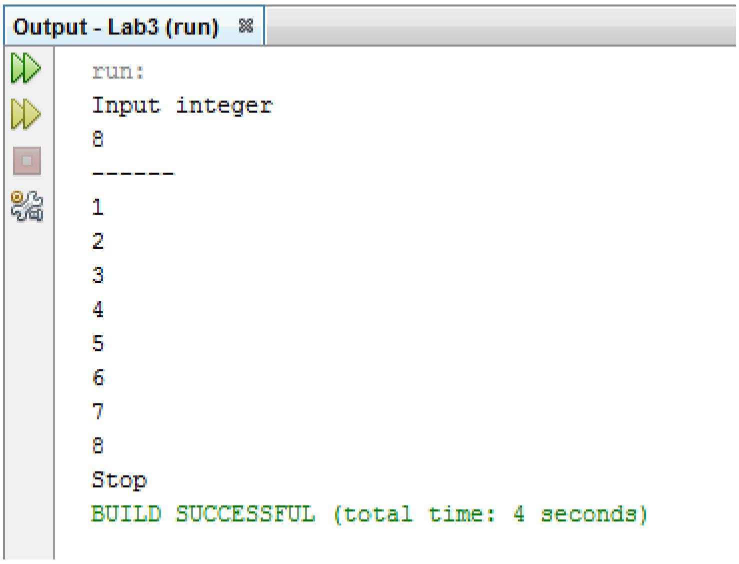
- In the lecture notes there is an example of a "countdown" while loop, copy this and run it

- It should count down from 10 to 0. Study the code. If you do not understand it, ask a TA. Change the code so it counts up from 1 to 5, and then displays "Stop"



```
run:
1
2
3
4
5
Stop
BUILD SUCCESSFUL (total time: 0 seconds)
```

- Now add a Scanner input to accept an integer, and count up from 1 to this integer



```
run:
Input integer
8
------
1
2
3
4
5
6
7
8
Stop
BUILD SUCCESSFUL (total time: 4 seconds)
```

- You have now created a while loop, and used keyboard input to define the conditions!

While Loops – Variable handling

A while loop is very useful for a number of things, not just counting

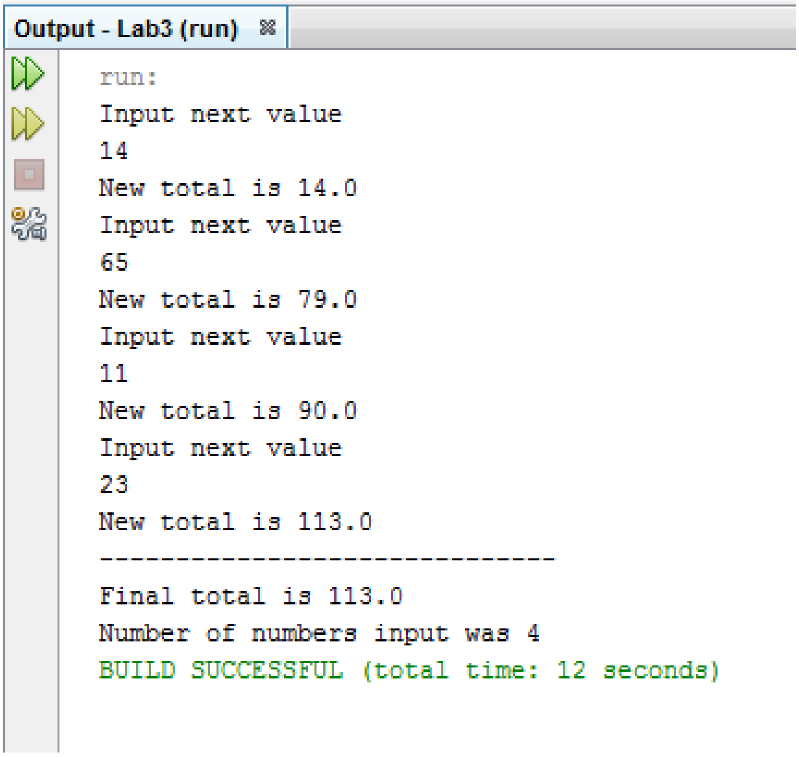
- Comment out your code in the WhileClass class, and create a new while loop

- This will take an integer input from the keyboard.We want to add each integer to a total, and display both an ongoing total, and then the final total

- To do this, we need an initial integer total variable set to 0, and then a loop where user input is repeatedly asked. This input is added to the total. The loop only stops when the total is greater than 100

- Using the code you have written in this lab so far, and your lecture notes, try to design and write a program that will solve this problem

- After the loop finishes, you should display both the total, and the number of times the program looped. An example of the console output is shown below



```
run:
Input next value
14
New total is 14.0
Input next value
65
New total is 79.0
Input next value
11
New total is 90.0
Input next value
23
New total is 113.0
------------------------------
Final total is 113.0
Number of numbers input was 4
BUILD SUCCESSFUL (total time: 12 seconds)
```

Methods

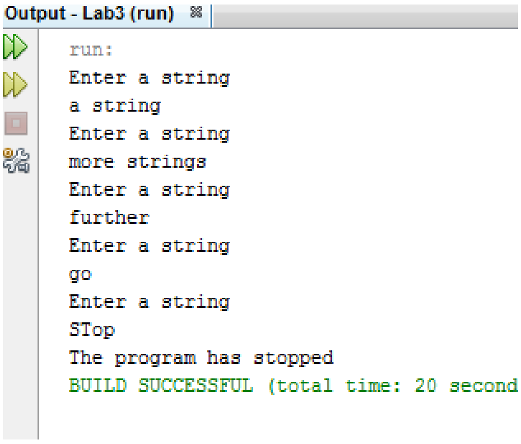
So far, everything you have done has been in the main method, which is what is called when a Java program is run. Good practice is that a method should do one thing, and methods can prevent repetition of code.

- Using the lecture notes, create a method in your ScannerInput class that will take a string from the scanner, and call it from your main class, with no output

- You will need to set your Scanner variable to be a global variable

- Once this is functioning, create a while loop in your method. The method will loop until the String is equal to "stop"

- Finally, we want to stop even if upper case letters like StOP or STOP are used. Check the lecture notes to find a suitable method to use



```
run:
Enter a string
a string
Enter a string
more strings
Enter a string
further
Enter a string
go
Enter a string
STop
The program has stopped
BUILD SUCCESSFUL (total time: 20 second
```

Methods – Passwords

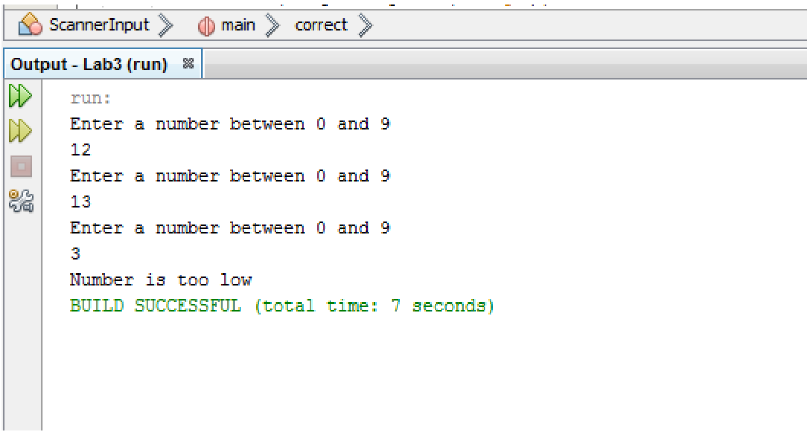
Create a second method, this is similar to the method you created previously, but this time, the method will receive a secret password as an argument, and then will ask for repeated inputs until the user input **contains** that password.

- Create a method that receives a string, and returns nothing. This will require a parameter
- Take user input from the keyboard. Set it to loop while the user input does not contain the password you have passed in (check your lecture notes for an appropriate String method)
- Change the method type to return that string to the main method, and store it as a variable
- Now, IF the length of the string is over 5 characters, then return a substring of the LAST five characters
- ELSE (length is 5 or less), return the entire string
- Change your method type to return that string to the main method

Methods – Secret Code

As well as passwords, we can also check digits. Create a method that receives one integer as an argument, and will loop until a digit between 0 and 9 has been entered. It will then return true if the final integer matches the parameter, and false otherwise.

- Create a method that receives an integer and returns a boolean
- Take user input from the keyboard, and loop while the input is not in the range [0-9]
- When a valid integer has been entered, after the loop, but in the method, check if the input matches the parameter passed in, and return a Boolean true or false as appropriate
- As an additional challenge, you should also report if the integer is too high, or too low
- The Boolean should be assigned to a variable in your main method



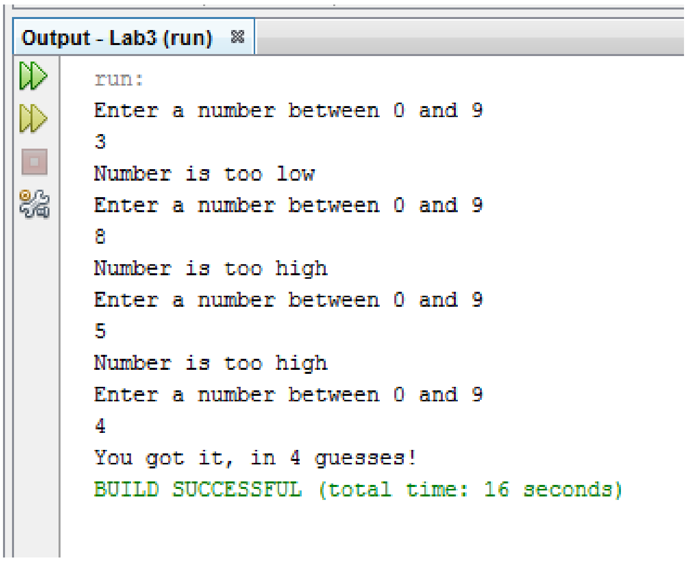
```
run:
Enter a number between 0 and 9
12
Enter a number between 0 and 9
13
Enter a number between 0 and 9
3
Number is too low
BUILD SUCCESSFUL (total time: 7 seconds)
```

Methods – Using Your Method!

In the previous section, you created a method that returns a Boolean variable. Now you want to loop this method call until the user guesses the correct number

- Loop your main method until the Boolean value is true

- Count how many guesses it took to get the correct number!



```
run:
Enter a number between 0 and 9
3
Number is too low
Enter a number between 0 and 9
8
Number is too high
Enter a number between 0 and 9
5
Number is too high
Enter a number between 0 and 9
4
You got it, in 4 guesses!
BUILD SUCCESSFUL (total time: 16 seconds)
```

- You have now made a game! Get your classmate to try it, and see how long it takes to guess your secret code!

- [Extra] Can you extend it to a 2 digit code?

Next Steps – Homework 3

When you have completed this sheet, move on to homework sheet 2 on ICE!