|  |
| --- |
| Logbook for ISD |
| Your name and student number |
|  |
|  |
|  |

Logbook for ISD

Your name and student number

Contents

[Introduction 2](#_Toc431296390)

[Week 1 3](#_Toc431296391)

[Exercises 1 3](#_Toc431296392)

[Exercises 2 3](#_Toc431296393)

[Exercises … 3](#_Toc431296394)

[Week 2 4](#_Toc431296395)

[Exercises 1 4](#_Toc431296396)

[Exercises 2 4](#_Toc431296397)

[Exercises … 4](#_Toc431296398)

[Week 3 5](#_Toc431296399)

[Exercises 1 5](#_Toc431296400)

[Exercises 2 5](#_Toc431296401)

[Exercises … 5](#_Toc431296402)

[Week…13 6](#_Toc431296403)

[Exercises 1 6](#_Toc431296404)

[Exercises 2 6](#_Toc431296405)

[Exercises … 6](#_Toc431296406)

# Introduction

A brief introduction to what you have done within the module and how your experience was with the exercises and the overall module. Probably up to half a page.

# Week 1

Some overview of the topics covered by the lecture and the exercises. Not too much, may be a paragraph.

## Exercises 1

Provide the exercises description and your answers. Where applicable use source code excerpts, explanations of these, represent your results, for example by showing screenshots of your program and, where applicable, display the use of your code repository (github) either by screenshots or by providing log data from your code repository.

## Exercises 2

Provide the exercises description and your answers. Where applicable use source code excerpts, explanations of these, represent your results, for example by showing screenshots of your program and, where applicable, display the use of your code repository (github) either by screenshots or by providing log data from your code repository.

## Exercises …

# Week 2

Some overview of the topics covered by the lecture and the exercises. Not too much, may be a paragraph.

## Exercises 1

(Assume 1 egg for each person)

Number Of People

1. Buy eggs equal to NumberOfPeople.
2. Put pan on stove.
3. Turn on gas to 3.
4. Add 1 \* NumberOfPeople gram of oil, and wait for (15 \* NumberOfPeople) in seconds.
5. Break eggs and pour them into the pan.
6. Get rid of egg shells.
7. Add salt (1 \* NumberOfPeople) in grams.
8. Start to stir at 80 rpm for (1 \* NumperOfPeople) in minutes.
9. Turn off the Stove.
10. Serve.

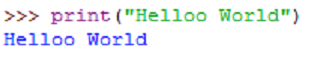
## Exercises 2

Python Idle (the Python language shell) is an interpreter. Because it translates the program one statement at a time. It is also easier to debug, because it translates one statement at a time, and there for making it simpler to pinpoint where the problem lies.

Python and Ruby use interpreter.

## Exercises 3

In order to put text, or more correctly in computer language known as “String”, to appear in Python programming language; you write the key word, **print,** followed by parenthesis(), like so, **print**( ), then inside the parenthesis you put 2 double quotation marks, like so **print**(“ ”), and between the double quotation mark you write any text, or better known as, **String,** like so, **print**(“Helloo world”), and then hit enter to start the show!(: And here is how the Magic looks when you put it all together.



## Exercises 4

So in this part everything works in the same way as Exercise 3 above; the only additional difference is the back slash followed by the letter n, like so: \n. If you put this in you text, String, then you’ll have the privilege of starting a new line, another magical code!



## Exercises 5

So in Python IDLE, create a new file to be able to write lines of programs.

For now please ignore the, **userName =**, in the image below, and just follow from the **input** written in red,in the image below.

So in order to make the computer to allow you to type in your name, or anything else, you need to use the command word, **input**. This will prepare the computer to take in your input!

However the **input** function is special and very clever! The input function is kinda like ordering an IceCream, and getting fresh red Strawberries with it too! Or whatever fruit you prefer.

So in practical term, the code, **input**(“Hi, there what is your name? ”), not only allows you to print your sentence**,** it also readies the computer to accept your response when you hit enter.

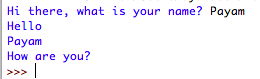
An example of this is shown in the first line of the second image with blue lines, and the name Payam, is my response.

Now you might still be wondering about the , userName = , code? Which you should be! Simply with ,userName = , the computer is just making a box or a folder, or any container that you like to think of to save, and hold the input you are going to type in. So in my case my response, Payam, is now safely resting or being held in a container called **userName.** You can call your container anything you like. Also, in computer world containers are referred to as **variables**. The name is probably due to the container's versatile nature; cause you can put anything in them, like text, numbers and …

So anyway cut a long story short once you named your container, userName = input(“Whatever...? ”), your response is saved, and then you can just use that response where ever and when ever you want. You just say the word baby, and I’ll come running...lol

As you can see I’ve used it in my second line of code print(“Hello,\n” + userName+”\nHow are you?” )

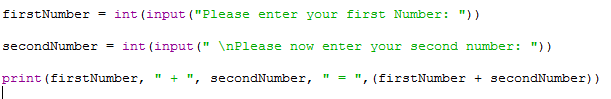


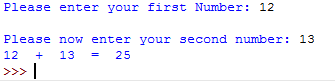


# Week 3

Some overview of the topics covered by the lecture and the exercises. Not too much, may be a paragraph.

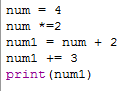
## Exercise 1





## Exercise 2

Part (a)

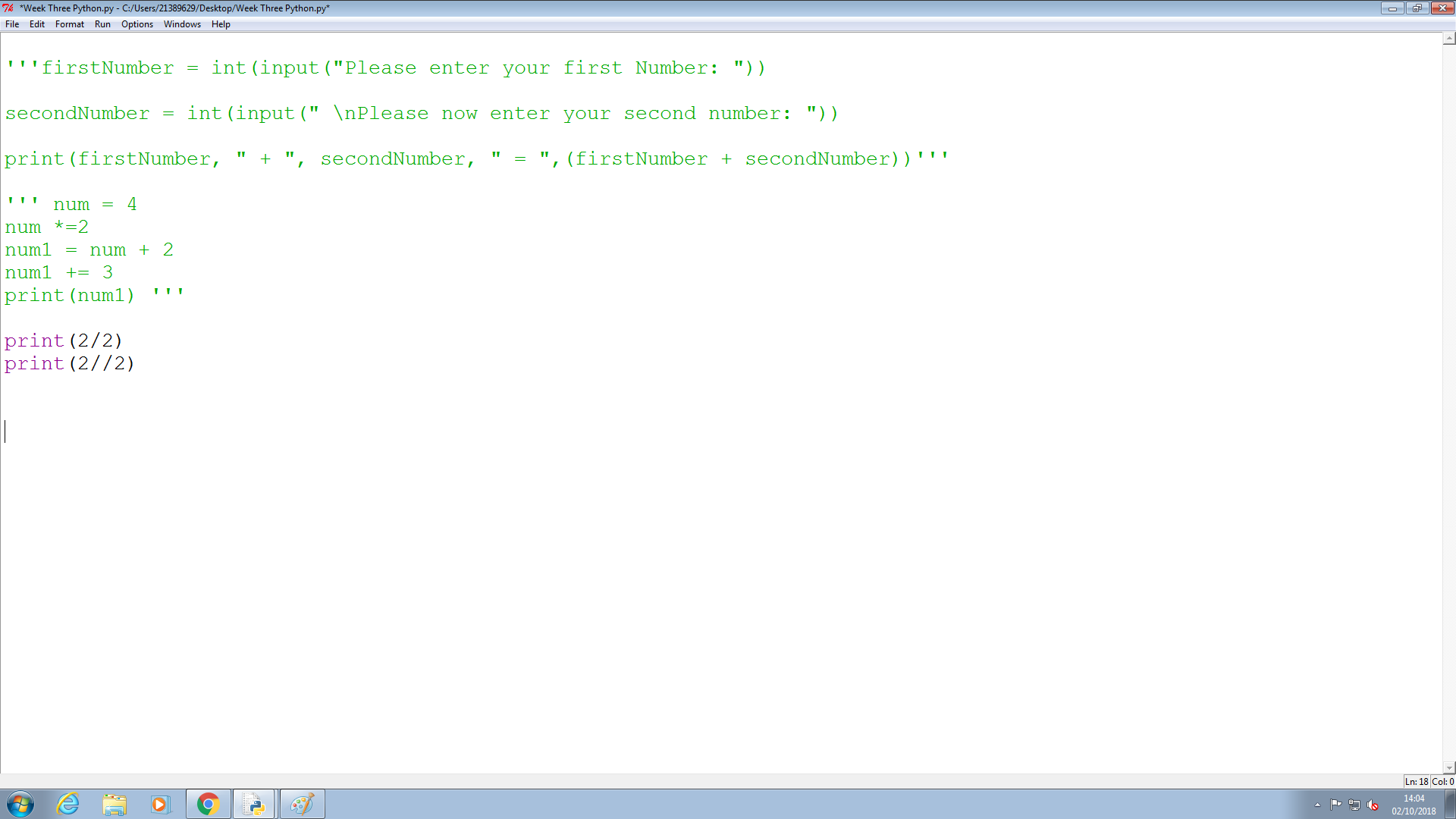




Part (b)

Single slash /, will return float type.

Double slash //, will specifically return an int type, there for saving memory, and serve for any other purpose, where whole numbers are needed.





## Exercise 3

areaOfRectangle

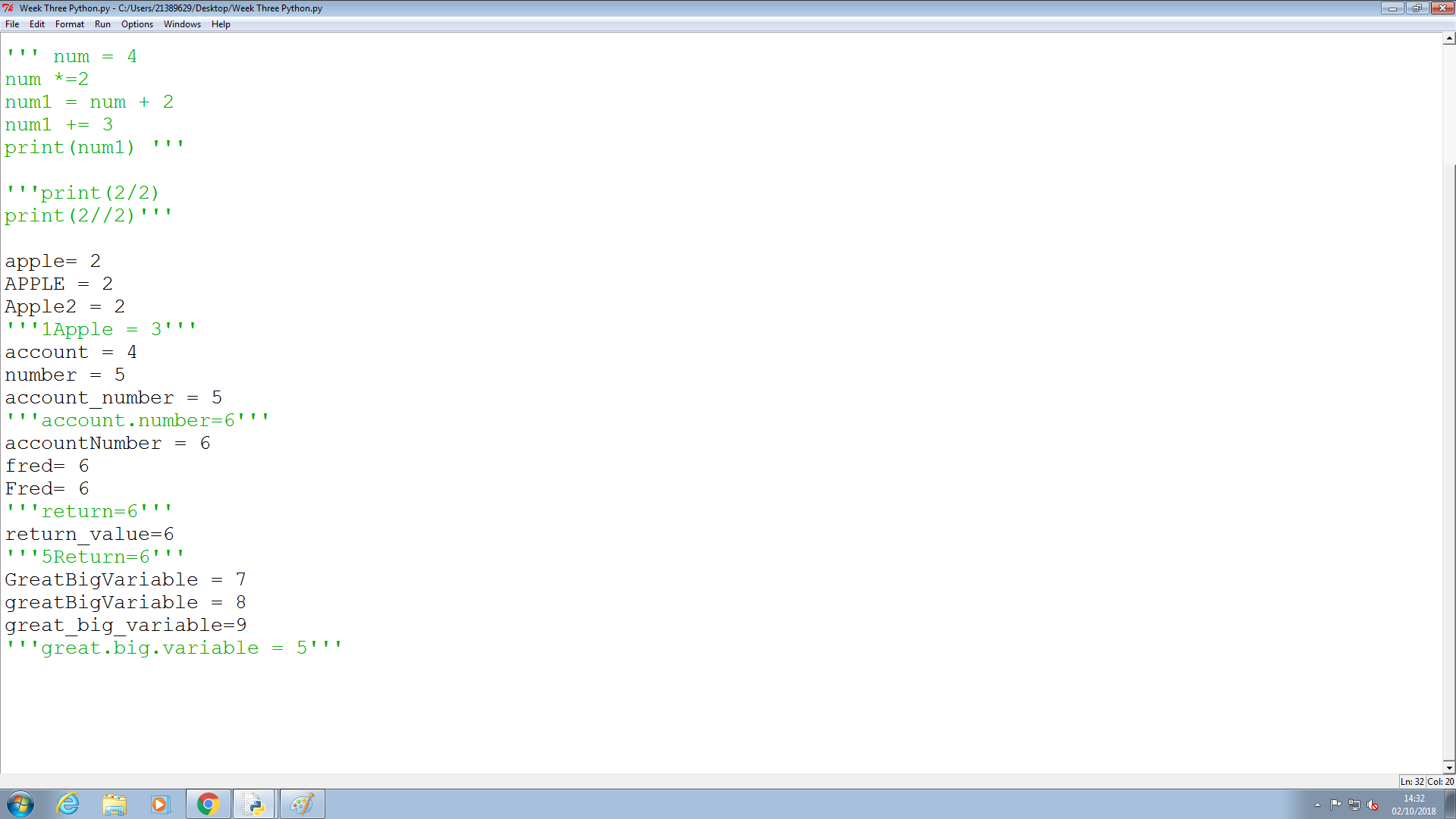
This name is better to use, because it is more informative as it indicates the area as well as the shape.

Also it is camel Casing, which distinguishes between each separate word, making it easier to read.

## Exercise 4

The texts highlighted in green, and wrapped by multiline comment single quotation mark are not allowed. One reason is that the first character variable name can’t be a number. Another reason is that variable names can’t contain a dot notation,”.”. The third reason is that the key word “return”, can’t be placed at the beginning of a variable name.

When ran the program those names with issues came up with problems so I commented them out.



## 

# Week…13

Continue this structure for the remaining weeks up until week 13

Some overview of the topics covered by the lecture and the exercises. Not too much, may be a paragraph.

## Exercises 1

Provide the exercises description and your answers. Where applicable use source code excerpts, explanations of these, represent your results, for example by showing screenshots of your program and, where applicable, display the use of your code repository (github) either by screenshots or by providing log data from your code repository.

## Exercises 2

Provide the exercises description and your answers. Where applicable use source code excerpts, explanations of these, represent your results, for example by showing screenshots of your program and, where applicable, display the use of your code repository (github) either by screenshots or by providing log data from your code repository.

## Exercises …

Example description of an exercise:

