Introduction

(Estimated time: 20 minutes)

This document is meant to help you familiarize yourself with the basics of coding. It is supposed to be fun and inspiring exercises that can show you the potential of coding. This project is going to be over the course of two weeks and contains introductory videos created by Code of the Future as well as exercises related to these videos. This project consists of an introduction and 3 sessions estimated to take 80 minutes to complete each. The programming language used is Python.

Introduction to python:

Watch: https://www.youtube.com/watch?v=1tOvHftYUpU (4:26 min.)

Question: Did you find this video encouraging or discouraging?

Introduction to coding:

The first three things to know about coding is this:

1) Whenever you meet a problem in your everyday life, then write it down. Coding is very versatile, and as the public is getting better at solving everyday problems themselves, so should you. After a couple of months you might return to the problem, and realize that you can solve it yourself using code. This method of writing down problems is also a way to keep up your interest in coding. The easiest way to learn coding is by working on a project. Why not have it be a fun project relevant to you?
2) Everyone can learn how to code, and that includes you. Do not get discouraged when getting error messages, data that is corrupted, or code that cannot run. Having problems with the code is something everybody that ever tried coding has experienced. Read the error message, search for help on the internet, ask a friend, and then try again with your newly acquired knowledge.
3) The internet is your best friend when coding. Almost every problem you can imagine with coding, someone else has had it before you. Using the documentation of functions and packages, or searching forums such as Stackoverflow is one of your strongest tools when coding.

Question: Do you agree with the three statements above?

Session 0

(Estimated time: 30-60 minutes)

Downloading Python using Anaconda:

When using Python you will need an interpreter. This will allow you to write scripts and run your code. For this introductory course, you will use either Anaconda Spyder or Anaconda PyCharm. The videos from Code of the Future use PyCharm, but you are just as welcome to use Spyder if that is prefered.

Task: Download Anaconda at Anaconda.com/downloads and complete installation. (Help installing PyCharm watch: https://www.youtube.com/watch?v=9HPf0UE1s2U (9:37 min.))

Now to create a nice setup in your chosen interpreter:

Spyder: *watch: https://www.youtube.com/watch?v=zcWUSKPNBNA (8:55 min.)*

PyCharm: *watch: https://www.youtube.com/watch?v=5LZdoDDDqmQ (10:14 min.)*

Task: Setup your chosen interpreter to your favorite theme.

Question: What theme did you pick?

CONGRATULATIONS: You are now ready to code!

Task: Write “print(‘Hello world!’)”

Question 4: What happens if you write “print(Hello world!)”?

As previously described Google is your friend when coding.

Task: Use Google to find out what the difference between “print(‘Hello world’)” and “print(Hello world)” is and why one gives an error.

Task: What happens when you write “print(2+2)”?

Session 1

(Estimated time: 80 minutes)

Your very first coding project:

Watch: https://www.youtube.com/watch?v=61Ps-Ykv3nU (27:29 min.)

Task: Calculate the following math using python and the print function:

a) $\frac{1}{2} \cdot 7$ *b) sin(2)* *3)* $3\sqrt{}$

Task: Write a comment in your code.

Question: Do you think PEP 8 warnings are relavant?

Task: Assign variables x and y as numbers you choose, now calculate:

a) x + 2y b) x^(2y) c) 2x/(2y) d) 2x/2y What is the difference between c) and d)?

Task: Create your own calculator code using variable x, y, a, my_number and the print function. Try to import some packages and see what they do.

Session 2

(Estimated time: 80 minutes)

Lists:

Watch: https://www.youtube.com/watch?v=iEuZsH2y51U *(29:00 min.)*

Task: Create a list of the numbers 4, 6, 8, 10, and assign it to a variable called "a".

Task: Create a list of the numbers 5, 9, 13, 17 and assign it to a variable called "b".

Task: Create a list of the numbers 4 till 19 and assign it to a variable called "c".

Question: Does "c[1]" equal 4 or 5?

Task: Find the third element of a, b, and c.

Task: Find the second last element of a, b, and c.

Task: Find the length of the list "d=[2,4,6,82,10,120,14,5,9,1398,174,6,8,10]".

Task: Create a list of 24 elements. Use a, b, a[1:3], and "+" operator to do so.

Task: Try to combine lists, find elements in lists, or assign new lists.

Session 3

(Estimated time: 80 minutes)

Loops:

Watch: https://www.youtube.com/watch?v=mZuY46sNLu4 *(14:56 min.)*

Task: Create a list with the elements 4 till 19 and assign it “a”

Task: Add 3 to every element of “a” using a “for” loop.

Question: Is 3854957234789 divisible by 17?

Task: Assign a variable “b” with the value 3854957234789. If it is divisible by 17 print “It works”, if not print “It doesn’t work”

Task: Create a list with elements 3854957234000 till 3854957234789 called “c”.

Question: How many of the numbers in “c” are divisible by 17?

Summary Quiz:

In the following video only do question 1-4 but watch the whole video

Watch: https://www.youtube.com/watch?v=4RzqiV7_Szg