Capgemini

Academy

Introduction round

Academy

Name

Job & background

Ambitions

What do you hope to take away from this session?

Goals of the session

Academy

- Understand what Python can be used for
- Be familiar with Python development fundamentals
- Getting some hands-on experience in programming with Python

Overview + planning

Academy

9.30-10.00:

- Overview of Python as a programming language
- Python vs other tools (including excel macros)
- Popular use cases for Python + some examples

10.00-12.30:

- Python fundamentals (using a development environment, writing and running Python scripts, variables and different basic data types, if statements)
- Exercises after every topic & coffee break somewhere in the middle

13.15-15.00:

- More Python fundamentals: writing functions and lists and dictionaries, maybe we can do basic loops here as well

15.15-16.30:

- Slightly more advanced: reading and writing files, libraries, csv library, demo of how python can automate excel tasks
- Small exercises about each topic
- If time allows, a slightly bigger project to end the day with

What can Python be used for?

Academy

Application development

Data analysis / machine learning

(Automation) Scripting

Internet of things

Glue language

Examples of things you can build

Academy

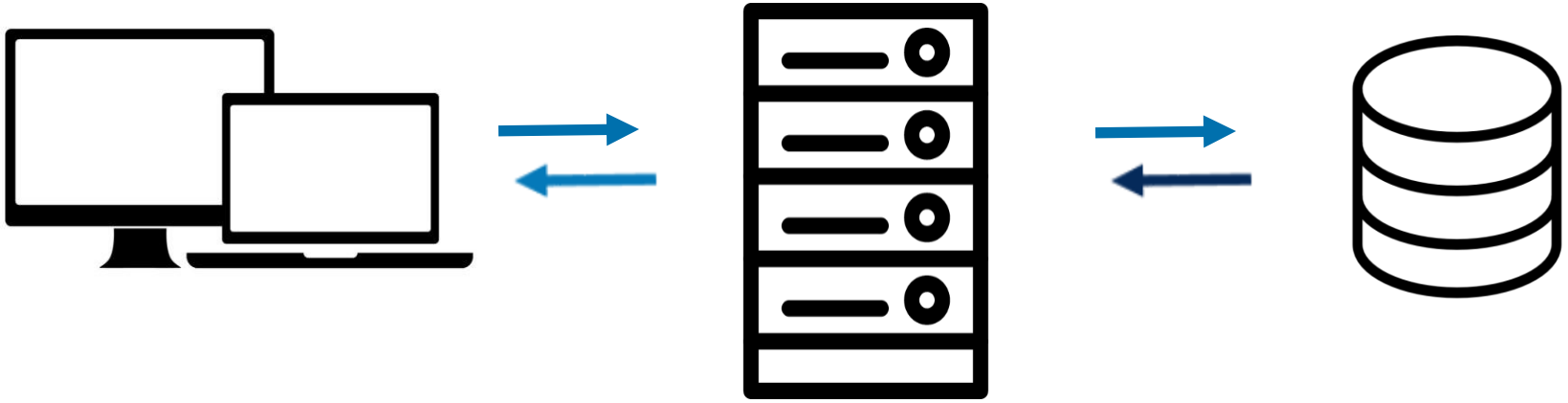
Smart script to determine what's the best route to pick up garbage for a municipality

Website to inform users about possible side effects of medication

Portals where citizens can arrange all sorts of matters with the government

Can you think of additional examples?

Overview of web applications



Python alternatives

Academy

- Excel Macros / VBA
- R
- MATLAB
- SAS
- Julia
- C# with .NET
- Java
- Ruby

Today’s menu: python

Academy

Python is a language created by Guido van Rossum at the start of the 1990’s. It’s Dutch!

Why Python? – it’s awesome, easy language to get started with, many many possibilities

What can you build with languages like Python: application development, data analysis / machine learning, (automation) scripting, internet of things, glue language

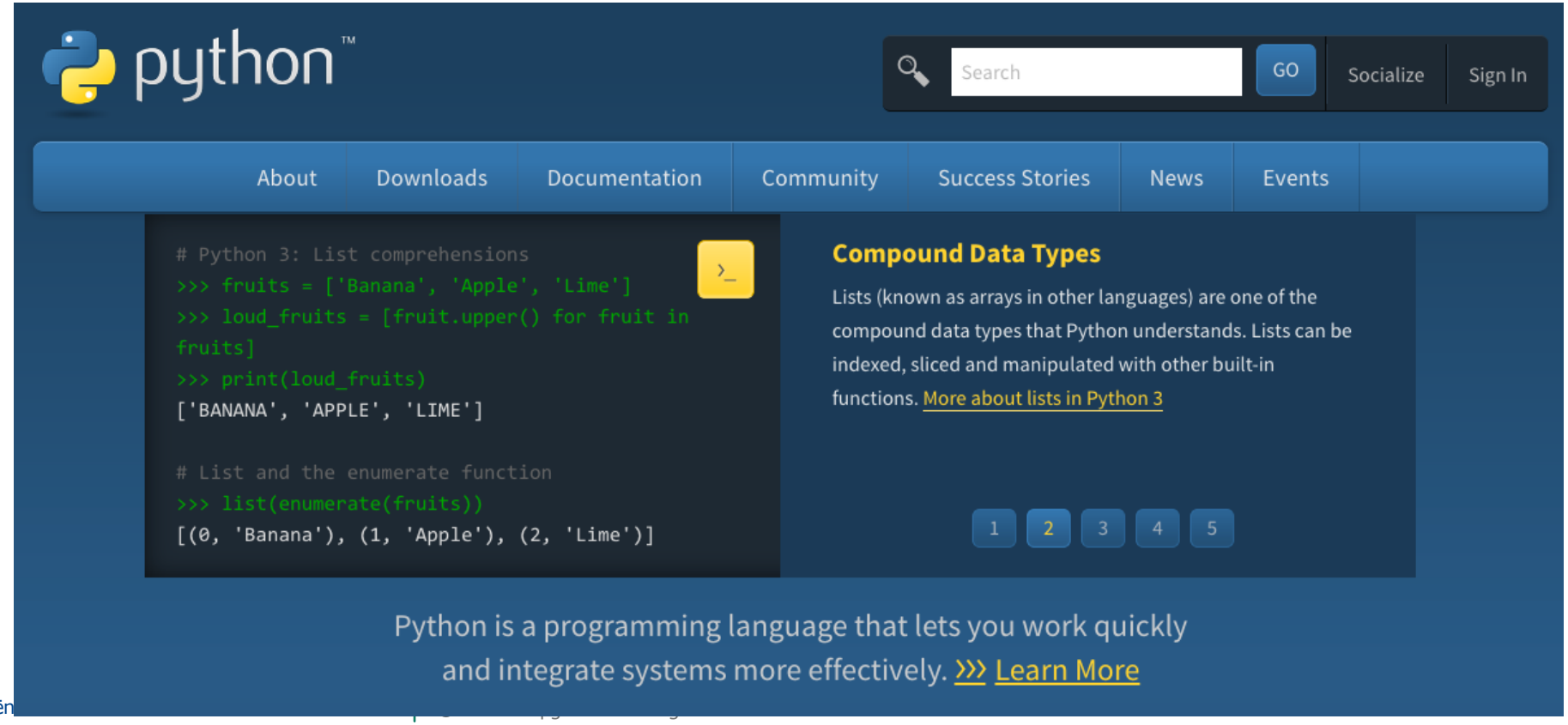
But why Python? – Because Gemeente Amsterdam uses it a lot

Why hands-on? So that you can get a good grasp of whether programming is something you’d love :) (Since it’s the main task of the application developer)

Python community

Academy

One of the strong points of Python is it's community of developers. The main website is Python.org



Python community

Academy

A very important man in this community is Guido van Rossum himself.

His title is BDFL (Benevolent Dictator For Life).

- in the end he has to decide about changes and additions.

Changes are proposed in the Python Enhancement Proposals (PEP).

A very well known PEP is PEP8 "The style guide for Python code".

Also PEP20, The Zen of Python, is famous.

- use import this

Documentation

Academy

On Python.org extensive documentation about Python can be found.

What's new in Python 3.6?
or all "What's new" documents since 2.0

Tutorial
start here

Library Reference
keep this under your pillow

Language Reference
describes syntax and language elements

Python Setup and Usage
how to use Python on different platforms

Python HOWTOs
in-depth documents on specific topics

Installing Python Modules
installing from the Python Package Index & other sources

Distributing Python Modules
publishing modules for installation by others

Extending and Embedding
tutorial for C/C++ programmers

Python/C API
reference for C/C++ programmers

FAQs
frequently asked questions (with answers!)

Academy

Python versions

Not all language constructs in the original Python setup were correct and efficient.

Changing them would mean incompatibility with earlier versions.

While Python was in it's 2 series a decision was made to change a number of language constructs and put this changes in the 3 series.

Part of the developer base still use version 2.x because existing tools and platforms are written in 2.x.

Version 2.x is still supported but this will stop in 2020.

In this Python course we are using 3.x.

Academy

Set up development environment

Set up development environment

Academy

Step 1: https://www.python.org/downloads/ to get latest version

Step 2: run .exe you just downloaded

Step 3: https://www.jetbrains.com/pycharm/download/ get latest version of Pycharm

Run installation

Open Pycharm

First program!

Academy

Open a new file and write: print(“Hello world”)

Run

Check console: does it say “Hello world”?

Academy

Writing and running Python scripts

Python syntax

Academy

Statements are terminated by a line feed.

Multiple lines can be seen as one line by Python with the \ character.

```
a = b + \
    c
```

A colon : and indentation are used to denote different blocks of code.

```
if a:
    statement1
    statement2
else:
    statement3
```

Hands on – code along

Academy

No better way to know whether this is something for you than to just try and see whether you enjoy doing it!

Building blocks of theory

Share solutions

- After point 4, who wants to work on their own pace?

Academy

Variables and data types

Variables

Academy

Any element or factor that is liable to change (variation)

For example: 2x
x is different in 2x = 10 than in 2x = 12

So if I say: greeting = x
x can be “hello” the one time, and “bye” the other time, depending on the value of x

Define variable in python

Academy

```
naam = 8
naam2 = “Hoi”
naam3 = 8.5
naam4 = True
```

Variables

Python is a dynamically typed language where variable names are bound to different values, possibly of varying types, during program execution.

test.py - test.py - [/private/var/folders/6y/t6yz1vgj0lg3kmkqq9dpddph0000gn/T/test.py]

test.py

Project

test.py

External Libraries

< Python 3.6.0 (/Library/Frar

test.py

```
a = 10
print(a)
a =10.0
print(a)
a='Hello'
print(a)
```

Run test

```
/Library/Frameworks/Python.framework/Versions/3.6/bin/python3.6 /Users/andrevanwieringen/temp/test.py
10
10.0
Hello

Process finished with exit code 0
```

Data types

Academy

String

Integer

Float

Boolean

Many more… But not for now ☺

Boolean

Academy

True
False

and
or
not
chaining

Capgemini

Academy

Mini exercises

Operations on data types

Academy

+

-

/ and //

*

**

%

Exercise

Academy

Write a program that takes the lengths of the two sides of a right triangle and calculates the length of the hypotenuse.

Exercise: operation with data types

Academy

Challenge!

I want to do calculations with a time
So if it’s 10.00 and I add 61 minutes, the result should be 11.01
Don’t use if yet, we didn’t discuss it and you don’t need it

If you’re stuck, chat me for a hint

Casting

Converting from one data type, to another

str()
int()
float()

And more, but not for now ☺

Exercise:

Ask the user for two numbers
Add these numbers
Print the calculation you did and the result

Ask the user for two numbers
Divide these numbers
Print the calculation you did and the result

Ask the user for name and highest age (s)he'll have this year
Calculate the year of birth
Print to the console: *username* is born in year *year*

Academy

If statements

Making decisions

If elif else

Academy

If is used for making decisions

If a certain condition is met, a block of code will be executed

If not, it won’t

If no condition is met and there is an else block, the else block will be executed

example

```
x = int(input('Geef een nr: '))
if x < 10:
    print(x, "kleiner dan 10")
elif x < 20:
    print(x, "kleiner dan 20, groter of gelijk 10")
else:
    print(x, "gelijk of groter 20")
```

Exercise:

Express each of the following statements using an if block:

- a. If x divided by y is 5, then set x to 100.
- b. If x times y is 5, then set x to 1.
- c. If x is less than y, then double the value of x.
- d. If x is greater than y, then increment the value of x by 1.

Exercise:

Write a program that prompts the user to enter five test scores between 1 and 10. The program will then count the number of scores that are greater than 7.

Express each of the following statements using an if/else block:

- a. If x times y is 8, then set x to 1; otherwise, set x to 2.
- b. If x is less than y, then double the value of x; otherwise, increment x by 1.
- c. If x is greater than y, then increment both by 1; otherwise, decrement both by 1.

Capgemini

Academy

Mini exercises

Academy

Academy

Functions

Python built-in functions

Academy

What are functions?

Difference between variable and function

Examples:
print()
input()

Exercise:

Write a program that asks that user to enter an article, a noun, and a verb.
The program then creates a sentence of the form: *article* *noun* *verb*.

Exercise:

Ask the user for his/her favorite color

Save the result in a variable

Ask the user for the first tool that comes to mind

Save the result in a variable

Print to the console: “combining your input I’m getting a *color* *tool*.”

Write your own functions

Academy

Arguments
Return

def

```
def goodmorning(name):
        print("Goodmorning, " + name + "!")

goodmorning("Maaike")
```

Exercise

Academy

Split the game time exercise up in separate functions

Call these functions

Types of arguments in functions

Academy

```
def goodmorning(name, day_name):
        print("Goodmorning, " + name + " on " + dayname + "!")
```

Positional:

```
goodmorning("Maaike", "Wednesday")
```

Keyword arguments:

```
goodmorning(day_name="Wednesday", name="Maaike")
```

optional arguments in functions

Academy

```
def goodmorning(name, day_name="this day"):
        print("Goodmorning, " + name + " on " + dayname + "!")
```

goodmorning("Maaike") → will print: goodmorning, Maaike on this day!

return

```
def goodmorning(name, day_name="this day"):
        return "Goodmorning, " + name + " on " + dayname + "!"
```

goodmorning("Maaike") → won't show anything, but returns: goodmorning, Maaike on this day!

Exercise

Academy

Write a function that returns the sum of letters in a name, in which each letter corresponds with its position in the alphabet

Let’s explore

Academy

What happens if you send a list to a function, and change the list in the function, but return nothing? Is the list changed everywhere?

What if you want to have a variable number of arguments?

What if you want to have a variable number of key value pairs as arguments?

Exercise

Academy

Change your lyric function to accept many pieces of lyrics

Capgemini

Academy

Mini exercises

Academy

Lists, dictionaries and loops

List

Academy

[“list”, “with”, “different”, “elements”, “could”, “be”, 5, “too”]

Empty list: []

Listname[0] >> gets first element from list
Listname[-1] >> gets last element
Listname[-2] >> gets second last element, etc

Built-in list functions

Academy

Length of list = len(*listname*)

Add to list: listname.append(“new item”)

Insert at an index: listname.insert(1, “second item”)

Remove item at index: listname.pop(1)

Remove from end: listname.pop()

Remove item with value: listname.remove(“second item”)

Sort the list: listname.sort()

Reverse the items in a list: listname.reverse()

Exercise:

Create a list with your favorite movies

Create a list with your favorite friends

Create a list with your favorite numbers

Exercise:

Create a list with your favorite movies
Create a list with your favorite friends
Create a list with your favorite numbers

Get the length of each list
Sort the lists
Remove your least favorite favorite one from each list
Append an alternative
Make sure that the lists are eventually in naturally reversed order (so Z to A and 9 to 0)

Other functions on lists

Academy

Make a new list object of a list that’s sorted: sorted(listname)

Get the max value: max(listname)

Get the min value: min(listname)

Get the sum: sum(listname)

Exercise:

From your number list, get the min, max, length and sum

Print all the lists sorted, without actually sorting them

To prove, also print the original list afterwards

Exercise:

We have a shop with 4 products and their prices:

```
articles = ["Cheese", "Bread", "Coffee", "Wine"]
prices = [8, 2, 4, 3]
```

Create a program that:

- prints a list of products and asks the user to choose one
- when the user makes a choice, he can enter an amount
- calculate and print the total price for his order

Bonus exercise:

https://www.codingame.com/home choose temperatures

While

Academy

While repeats a code as long as a certain condition is met

Break

Continue

```
while i < 20:
        i+=1
        if i == 13:
                continue
        print(i)
```

Exercise

Write a program that asks the user to enter three numbers. The program will then determine and print the largest of the three numbers.

Create an input validation loop that only accepts numbers in the range of 1 through 10.

Write a script that asks the user, “Are you sure you want to quit [Y, N]?” The script then checks the user’s input and only accepts the letters Y and N as valid answers.

Exercise:

Create a while loop for the following:

- Ask the user for a word
- Store this word in a list (use append)
- Print the list
- When the user enters nothing, stop the program

Deduplication

Change the program so that, every time the user gives a word, you test whether it is already in the list. If it is, you don't add it.

Lengths

Ask the user to enter 4 words. Put those in a list with append.

After all words have been entered, print:

- The shortest and longest word
- The average length of all words

For

Loop over elements of an iterable.

```
x = [1, 2, 3]

for y in x:
    print(y)
```

Exercise

Academy

Loop over the letters of a string and print them to the console

Create a list that contains the days of the week
Create a list containing the working days of the week
Loop over the elements of the first list and print days that are not in the working days

We haven’t seen this yet, but how can you look over the first 5 elements in a list?
And the last five?

Range()

Academy

Function that returns a sequence of numbers
range(start, end, step)

range(5) → return numbers 0, 1, 2, 3, 4
range(2, 5) → returns numbers 2, 3, 4
range(2, 5, 2) → returns numbers 2, 4

You can use it directly in a for loop

- for x in range(0, 100, 5):

Exercise

Academy

Make a list with the numbers 1 to a milion

Sum them and see how fast python can do this

Use range() to make a list of 0 to 1000 and use this to create a new list with all these numbers squared

Dictionary

Academy

{}

Key value pairs

```
thisdict = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
```

```
thisdict["year"] = 2018
```

Loop through Dictionary

Academy

```
for x, y in thisdict.items():
  print(x, y)

for x in thisdict.values():
  print(x)

for x in thisdict.keys():
  print(x)
```

Dictionary functions

Academy

.items
.keys
.values
.popitem
.pop(key)
.get(key)

Exercise

Create three dictionaries of your favorite sport players, store some important properties of a sport player, like what sport, name, year (s)he was born, description

Loop over the items in each dictionary, write the content to a file to create mini biographies about each

Create a list with these dictionaries

Loop over the list, and in each iteration, loop over the items in the dictionary and print these to the console

Game time! – mini project

Academy

Write a game that takes a random integer
And makes the user guess that integer until he guess correctly
Show higher or lower to help the user get it right

Bonus:

- Keep the nr of guesses
- Collect all the guessed numbers and print them later
- If someone uses less than 5 guesses, print that (s)he's awesome
- Make the program safe for non-numeric input
- Save this data to a file: guesses, nr of guesses, correct number

Bonus bonus:

- Write a script that does the guessing, and log the guesses it did
- Run it 10000 times to get the average nr of guesses
- See if you can optimize the script even further

Capgemini

Academy

Mini exercises

Academy

File handling and using libraries

Modules

Academy

A module is just file with Python functions and properties.

You can import a module using the “import” keyword on top of the class

There are many useful standard modules, e.g. math

A module name is the file name without the .py extension

The module name is the value of the global variable __name__

Exercise modules:

Academy

Find a module to generate a random integer

Find a module to get the ceiling and floor of x

Play around with turtle module!

Export your functions in your own modules

Academy

We've been writing modules al along! You save them with the extension .py

After that you can import them using the import statement

Exercise:

Create a separate .py file with a function to print information about a user to a file / console

Import this function to your current file

Call the function

Bonus: how to import not an entire module but only one function?

Open/read/write files

Academy

```
open(name)
read() or readlines()
close()

open(name, “w+”)
write(“blabla”)
close()
```

Exercise

Create a file

Write a little story about the new things you've learnt today

Read from the file and print it to the console

Csv reading

Academy

https://docs.python.org/3/library/csv.html on CSV support

Read CSV files with the CSV reader object

Example CSV format:

Name, city, job

CSV reading Example

Academy

```
import csv

with open("data.csv", 'r') as csvfile:
    # Create a reader object
    reader = csv.reader(csvfile)

    # Let’s loop over de lines in the file
    for row in reader:
        # row is a list with the fields from a line
        # print city
        print(row[1])
```

CSV Writing – use CSV writer obj

Academy

```
import csv

data = [[“animals", “legs", “wings"],
        [“dog", 4, 0],
        [“bird", 2, 2]]

with open(‘animals.csv', 'w') as f:
    writer = csv.writer(f)

    for l in data:
        # write a row
        # please note: the argument for writerow must be a list
        writer.writerow(data)
```

Exercise

Academy

Imagine we own a cable shop

Create a csv file with our products, prices, and number in stock

You can write to the csv file using a list of our products, list of prices and list of stocks

Try to import your csv in excel

Read the values of your csv and print them to the console per row

Xml

Academy

Xml: import xml.dom.minidom

In order to parse XML you'll need to have the entire XML object in memory

Parsing and writing xml

Academy

```
import xml.dom.minidom

def main():
    # use the parse() function to load and parse an XML file
    doc = xml.dom.minidom.parse("Myxml.xml");

    # print out the document node and the name of the first child tag
    print (doc.nodeName)
    print (doc.firstChild.tagName)
    # get a list of XML tags from the document and print each one
    expertise = doc.getElementsByTagName("expertise")
    print ("%d expertise:" % expertise.length)
    for skill in expertise:
        print (skill.getAttribute("name"))

    # create a new XML tag and add it into the document
    newexpertise = doc.createElement("expertise")
    newexpertise.setAttribute("name", "BigData")
    doc.firstChild.appendChild(newexpertise)
    print (" ")

    expertise = doc.getElementsByTagName("expertise")
    print ("%d expertise:" % expertise.length)
    for skill in expertise:
        print (skill.getAttribute("name"))

if __name__ == "__main__":
    main();
```

```
<?xml version="1.0" encoding="UTF-8"?>
<employee>
    <fname>Krishna</fname>
    <lname>Rungta</lname>
    <home>London</home>
    <expertise name="SQL"/>
    <expertise name="Python"/>
    <expertise name="Testing"/>
     <expertise name="Business"/>
</employee>
```

Source: https://www.guru99.com/manipulating-xml-with-python.html

json

Academy

```
import json

# Opening JSON file
f = open(‘example.json', )

# returns JSON object as
# a dictionary
data = json.load(f)

# Iterating through the json
# list
print(data[‘some'][‘keys'])

# Closing file
f.close()
```

How to figure out new stuff?

Academy

Gef files

How to find libraries for these purposes?

Go to https://pypi.org and search for gef

GETTING VALUES OUT OF GEF FILE

Academy

Pip install pygef

Check documentation: https://github.com/ritchie46/pygef

Exercise: Read gef

Academy

Read the GEF

Convert it to CSV

Plot the GEF

Capgemini

Academy

Mini exercises

Academy

Excel demo

Python and excel

Academy

Multiple libraries

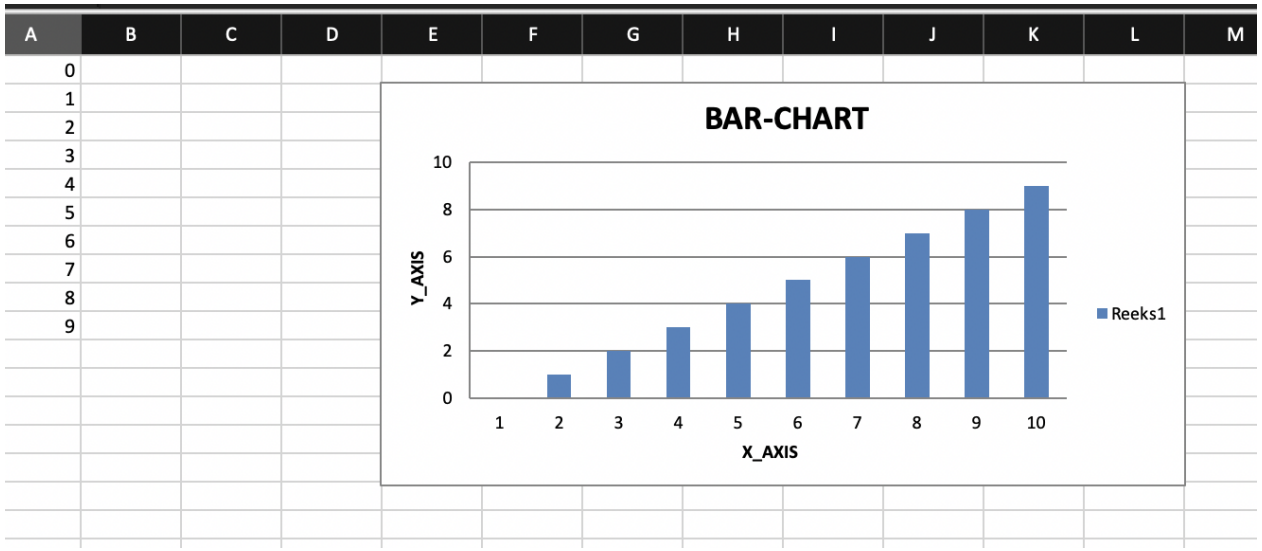
Look through the docs to find details

```
from openpyxl import Workbook

workbook = Workbook()
sheet = workbook.active

sheet['A1'] = 'Using'
sheet['B1'] = 'Excel!'

workbook.save(filename='test.xlsx')
workbook.close()
```

Create graph in excel

Academy

```
import openpyxl
from openpyxl.chart import BarChart, Reference

wb = openpyxl.Workbook()
sheet = wb.active
for i in range(10):
    sheet.append([i])

values = Reference(sheet, min_col=1, min_row=1,
                   max_col=1, max_row=10)

chart = BarChart()

chart.add_data(values)

chart.title = " BAR-CHART "

chart.x_axis.title = " X_AXIS "

chart.y_axis.title = " Y_AXIS "

sheet.add_chart(chart, "E2")

wb.save("examplegraph.xlsx")
```



https://www.geeksforgeeks.org/python-plotting-charts-in-excel-sheet-using-openpyxl-r

pandas

Data analysis and processing tool / library

Docs: https://pandas.pydata.org/docs/

Tables / tabular data are called DataFrame

Often combined with numpy in order to support large multi dimensional arrays and provides many mathematical functions to work with these multi dimensional arrays

Let’s build an app!

Academy

Simple calculator

But first.. What blocks of logic do we need to build a calculator? https://www.mentimeter.com/features/word-cloud.

How to become proficient in Python

Academy

Many ways……

- You could do it all by yourself
- But as a beginner it will speed up the process to start with guided education such as training and finally, you'll have solid basic knowledge, and you can continue with self-directed education
- Practice practice practice…
- Ensure that you can apply your knowledge in real life, so get a project or work on one of your own app ideas

De professionals van Capgemini Academy bieden IT'ers wat ze nodig hebben. Onze mensen hebben een scherp oog voor drijfveren, aandacht voor talent en besef van specifieke omstandigheden. Ze bewegen tot beweging. Programma's die hun oorsprong vinden in het dagelijks werk van onze zowel didactisch als inhoudelijk onderlegde trainers wakkeren het vuur aan. Praktijkverhalen die vertellen hoe je problemen met IT en de mensen eromheen nou écht oplost doen de rest.

Een instituut als het onze helpt mensen en organisaties iedere dag weer het beste uit zichzelf en elkaar te halen. Bereidt hen voor op het zelfbewust aangaan van de uitdagingen van morgen. Stimuleert leer- en nieuwsgierigheid. Opdat IT'ers en hun werkgever beter, langer en intensiever met elkaar vooruit kunnen. Tot wederzijds genoegen.

Capgemini Academy's professionals offer what people in IT need. Our professionals have a keen eye for motivation, talent and are aware of specific contexts and circumstances. They move people to move. Programmes and courses that originate from daily experience of our both didactical and substantively strong trainers, light a fire within the individual IT professionals. Real life stories of our professionals' experience that tell how to solve problems and work with the people around it, do the rest.

An organization, like ours, helps people and their organizations day by day to get the best out of themselves and each other. We prepare them to defy tomorrow's challenges. We stimulate learning and curiosity. In order for individual IT professionals and their employers, to build better, longer and more intensive relationships. For mutual benefit.

Capgemini Academy. We transform IT professionals.

Academy

academy.capgemini.nl

Capgemini

People matter, results count.