**Agenda:**

- Install Python
- Install PyCharm
- virtualenv
- Debugger Showcase
- Python syntax

**Resources**:

- PyCharm CE: https://www.jetbrains.com/pycharm/download/#section=windows
- Python 3.8: https://www.python.org/downloads/windows/

## Install Python

Python can be downloaded from the official website:

- Python 3.8
- Link: https://www.python.org/downloads/windows/

For Python please follow the default installer. The Python interpreter should be at this location

```
C:\Users\<Your-User>\AppData\Local\Programs\Python\Python38\python.exe
```

**Extra step (NOTE! This is done automatically if 'Add python to PATH checkbox is checked in the installer):**

Add Python and Python-related scripts to the Path environment variable

- Right Click on My Computer(This PC) > Properties > Advanced System Settings > Environment Variables
- Edit the `Path` variable from `User Variables` and add the following paths
  - `C:\Users\<Your-User>\AppData\Local\Programs\Python\Python38\`
  - `C:\Users\<Your-User>\AppData\Local\Programs\Python\Python38\Scripts`

Make sure you change with the actual User name

## Install PyCharm

PyCharm Community Edition can be downloaded from the official website :

https://www.jetbrains.com/pycharm/download/#section=windows

Please follow steps on the default installer.

Enable VCS integration:

Goto menu : VCS > Enable VCS integration, and select the installed VCS.

## virtualenv

Virtualenv is a python package that helps us manage dev environments.

From terminal install virtualenv by running the following command:

- `pip install virtualenv`

Create a virtualenv

- `python -m virtualenv .venv`

## Debugger Showcase

Live Demo

## Python syntax

### Comments

Comments in Python start with the hash character, #, and extend to the end of the physical line. A comment may appear at the start of a line or following whitespace or code, but not within a string literal. A hash character within a string literal is just a hash character.

```
# this is the first comment
spam = 1  # and this is the second comment
          # ... and now a third!
text = "# This is not a comment because it's inside quotes."
print(text)
```

```
# This is not a comment because it's inside quotes.
```

### Numbers

The Python interpreter can act as a simple calculator: type an expression at it outputs the value.

```
2 + 2
```

```
4
```

```
8 / 5
```

```
1.6
```

```
5 ** 2
```

```
25
```

## Variables

The equal sign (=) assigns a value to a variable:

```
width = 20
height = 5 * 90
width * height
```

```
9000
```

## Strings

Besides numbers, Python can also manipulate strings. Strings can be enclosed in single quotes ('...') or double quotes ("...") with the same result.

```
'some text'
```

```
'some text'
```

```
"some text"
```

```
'some text'
```

The print() function produces a more readable output by omitting the enclosing quotes and by printing escaped and special characters:

```
print('Hello World')
```

```
Hello World
```

## Lists

Python knows several compound data types, which are used to group together with other values. The most versatile is the list, which can be written as a sequence of comma-separated values (items) between square brackets.

```
squares = [1, 4, 9, 16, 25]
squares
```

```
[1, 4, 9, 16, 25]
```

```
squares[0]
```

```
1
```

```
squares[-1]
```

```
25
```

## Statements

In python, we have a lot of statements that can help us write code. I will present some examples of conditionals using `if-else` and iteration using `for`

```
items = [1, 2, 3, 4, 5, 6, 7]

for item in items:
    if item % 2 == 0:
        print(item)
```

```
2
4
6
```