Basic Python Programming

Computational Design Laboratory

Department of Aerospace Engineering
Iowa State University

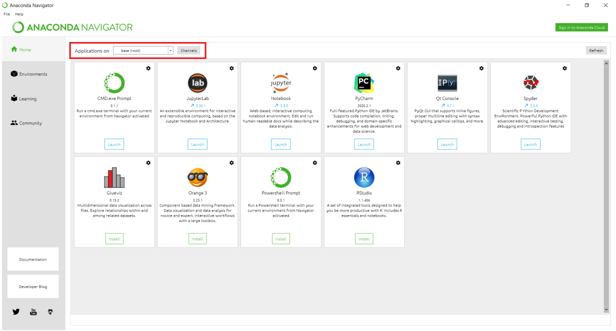
September 2, 2020

Outline

- Launching the Spyder notebook
- Download codes from Github
- Running your first python program
- Using Python as a Calculator
- Use of variables
- Lists
- Functions
- Control Flow Tools
- Further reading

Launching the Spyder Notebook

- Open anaconda-navigator
- Select channel at the top
- Choose base (root), if you selected default location while installing
- Otherwise select the filename you have chosen



Applications on base (root)

CMD.exe Prompt
0.1.1
Run a cmd.exe terminal with your current environment from Navigator activated

JupyterLab
↗ 0.32.1
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook
↗ 5.5.0
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

PyCharm
2020.2.1
Full-featured Python IDE by JetBrains. Supports code completion, linting, debugging, and domain-specific enhancements for web development and data science.

Qt Console
↗ 4.7.1
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder
↗ 3.3.5
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Glueviz
0.15.2
Multidimensional data visualization across files. Explore relationships within and among related datasets.

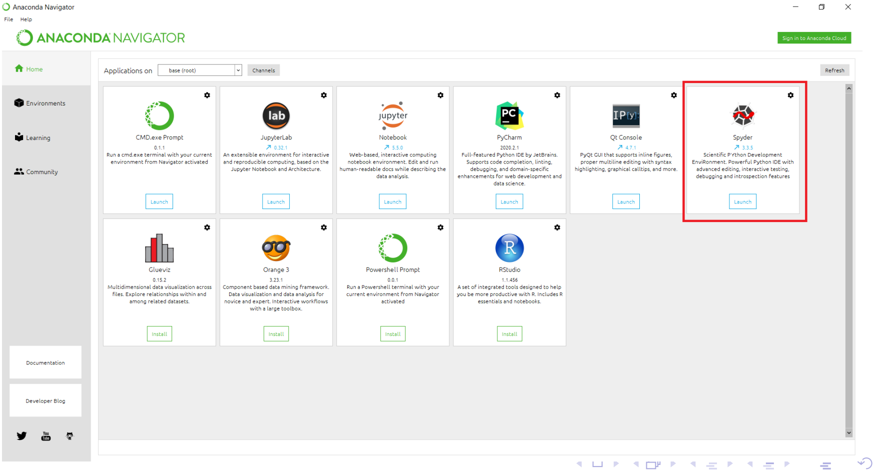
Orange 3
3.23.1
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

Powershell Prompt
0.0.1
Run a Powershell terminal with your current environment from Navigator activated

RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

Launching the Spyder Notebook

- Under spyder, click on install
- Once installed, click launch (always launch Spyder from Anaconda)



CMD.exe Prompt
0.1.1
Run a cmd.exe terminal with your current environment from Navigator activated

JupyterLab
0.32.1
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook
5.5.0
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

PyCharm
2020.2.1
Full-featured Python IDE by JetBrains. Supports code completion, linting, debugging, and domain-specific enhancements for web development and data science.

Qt Console
4.7.1
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder
3.3.5
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Glueviz
0.15.2
Multidimensional data visualization across files. Explore relationships within and among related datasets.

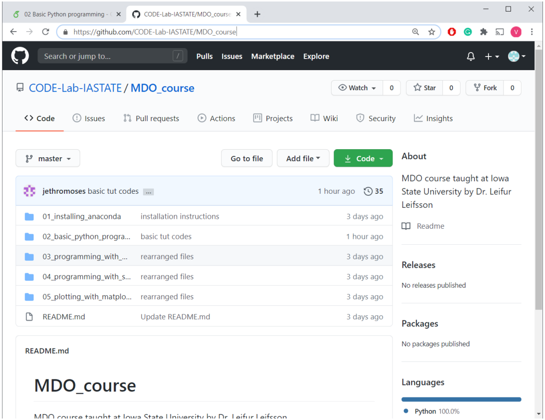
Orange 3
3.23.1
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

Powershell Prompt
0.0.1
Run a Powershell terminal with your current environment from Navigator activated

RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

Downloading codes from Github

- Download codes from here: ▸ Link
- Click on Code button in green on Github



CODE-Lab-IASTATE / **MDO_course**

jethromoses basic tut codes … 1 hour ago 35

01_installing_anaconda	installation instructions	3 days ago
02_basic_python_progra...	basic tut codes	1 hour ago
03_programming_with_...	rearranged files	3 days ago
04_programming_with_s...	rearranged files	3 days ago
05_plotting_with_matplo...	rearranged files	3 days ago
README.md	Update README.md	3 days ago

About

MDO course taught at Iowa State University by Dr. Leifur Leifsson

Releases

No releases published

Packages

No packages published

Languages

● Python 100.0%

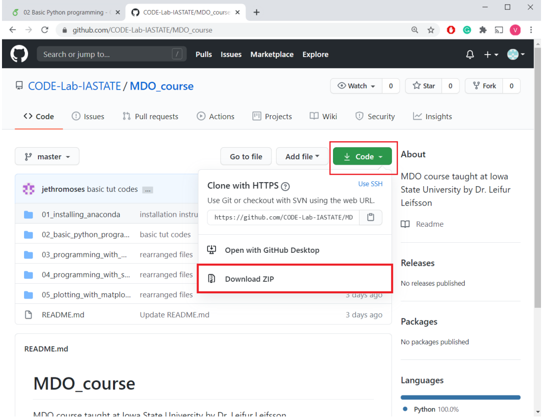
README.md

MDO_course

MDO course taught at Iowa State University by Dr. Leifur Leifsson

Download codes from Github

- Download all the codes as a zip file
- Save it to the location of your choice



github.com/CODE-Lab-IASTATE/MDO_course

CODE-Lab-IASTATE / MDO_course

Clone with HTTPS

Use Git or checkout with SVN using the web URL.

`https://github.com/CODE-Lab-IASTATE/MD`

jethromoses basic tut codes

01_installing_anaconda	installation instru	
02_basic_python_progra...	basic tut codes	
03_programming_with_...	rearranged files	
04_programming_with_s...	rearranged files	
05_plotting_with_matplo...	rearranged files	3 days ago
README.md	Update README.md	3 days ago

About

MDO course taught at Iowa State University by Dr. Leifur Leifsson

Releases

No releases published

Packages

No packages published

Languages

Python 100.0%

README.md

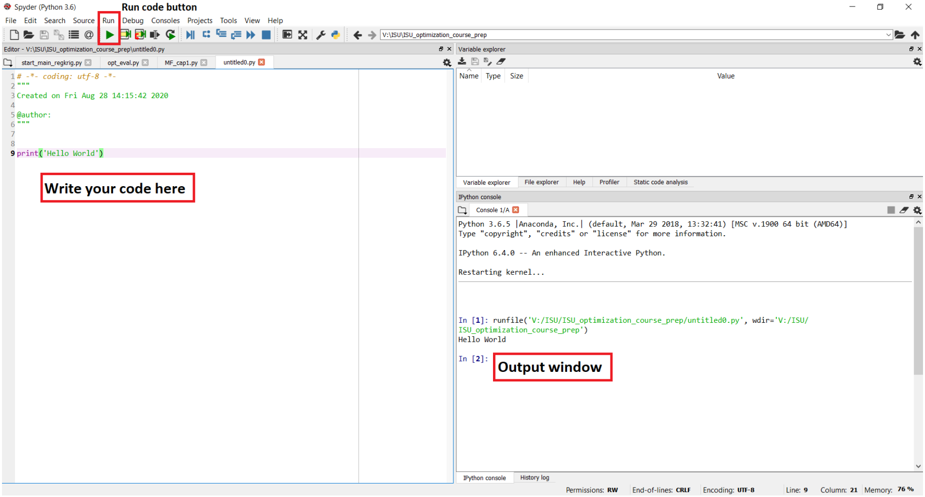
MDO_course

MDO course taught at Iowa State University by Dr. Leifur Leifsson

Running your first python program

- Click on new file in the top left corner
- Rename it to hello_world.py and save it to the location of your choice
- Type the command print('Hello World')
- Run it using the arrow button on the top

Running your first python program



Run code button

```python
# -*- coding: utf-8 -*-
"""
Created on Fri Aug 28 14:15:42 2020

@author:
"""


print('Hello World')
```

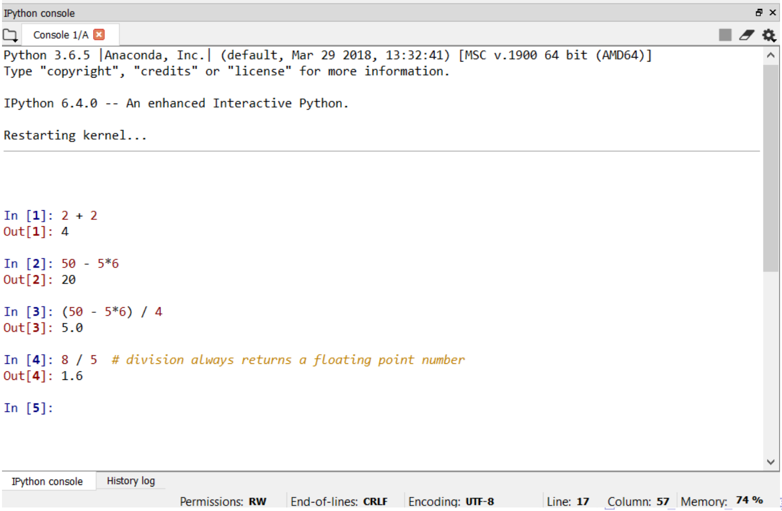
Write your code here

```
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.4.0 -- An enhanced Interactive Python.

Restarting kernel...



In [1]: runfile('V:/ISU/ISU_optimization_course_prep/untitled0.py', wdir='V:/ISU/
ISU_optimization_course_prep')
Hello World

In [2]:
```

Output window

Using Python as a Calculator

- Type directly in output window and press enter for output
- Link for more details : ▸ Link

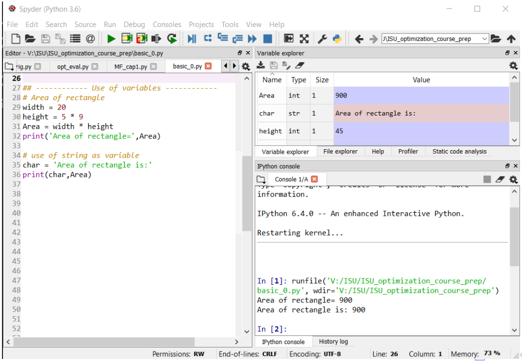


```
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.4.0 -- An enhanced Interactive Python.

Restarting kernel...



In [1]: 2 + 2
Out[1]: 4

In [2]: 50 - 5*6
Out[2]: 20

In [3]: (50 - 5*6) / 4
Out[3]: 5.0

In [4]: 8 / 5  # division always returns a floating point number
Out[4]: 1.6

In [5]:
```

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 17 Column: 57 Memory: 74 %

Use of variables

- Open the folder you have downloaded from GitHub
- Open the 02_basic_python_programming folder
- Open basic.py from Spyder
- This code can also be found here : ▸ Link
- Run the code
- Example: Calculate the area of rectangle (Spyder GUI shown below)



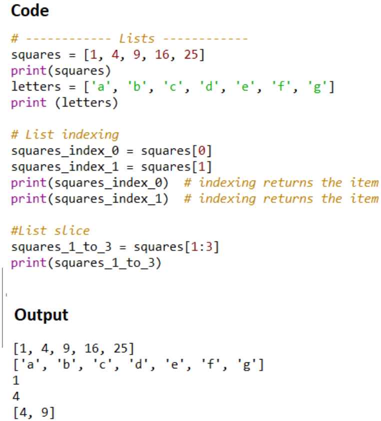
```python

## ------------ Use of variables ------------
# Area of rectangle
width = 20
height = 5 * 9
Area = width * height
print('Area of rectangle=',Area)

# use of string as variable
char = 'Area of rectangle is:'
print(char,Area)
```

Name	Type	Size	Value
Area	int	1	900
char	str	1	Area of rectangle is:
height	int	1	45

```
information.

IPython 6.4.0 -- An enhanced Interactive Python.

Restarting kernel...



In [1]: runfile('V:/ISU/ISU_optimization_course_prep/
basic_0.py', wdir='V:/ISU/ISU_optimization_course_prep')
Area of rectangle= 900
Area of rectangle is: 900

In [2]:
```

Lists

- basic.py also contains example on list datatype
- The most versatile datatype in python is the list, which can be written as a list of comma-separated values (items) between square brackets.
- Lists might contain items of different types, but usually the items all have the same type.



Code

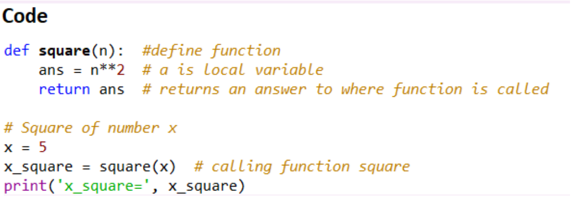
```
# ------------ Lists ------------
squares = [1, 4, 9, 16, 25]
print(squares)
letters = ['a', 'b', 'c', 'd', 'e', 'f', 'g']
print (letters)

# List indexing
squares_index_0 = squares[0]
squares_index_1 = squares[1]
print(squares_index_0)  # indexing returns the item
print(squares_index_1)  # indexing returns the item

#List slice
squares_1_to_3 = squares[1:3]
print(squares_1_to_3)
```

Output

```
[1, 4, 9, 16, 25]
['a', 'b', 'c', 'd', 'e', 'f', 'g']
1
4
[4, 9]
```

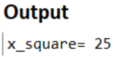
Functions

- Open the folder you have downloaded from GitHub
- Open the 02_basic_python_programming folder
- Open functions.py from Spyder and run the code
- This code can also be found here : ▸ Link
- Functions are used to perform particular task
- Example squaring the number



Code

```
def square(n):  #define function
    ans = n**2  # a is local variable
    return ans  # returns an answer to where function is called

# Square of number x
x = 5
x_square = square(x)  # calling function square
print('x_square=', x_square)
```

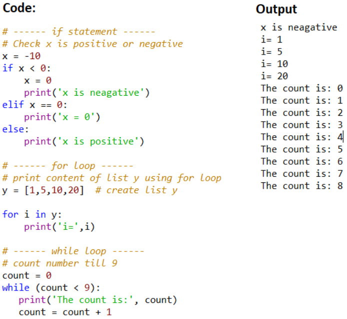


Output

```
x_square= 25
```

Control Flow Tools: if statement, for loop and while loop

- Open the folder you have downloaded from GitHub
- Open the 02_basic_python_programming folder
- Open loops.py from Spyder and run the code
- This code can also be found here : ▸ Link
- A program's control flow is the order in which the program's code execute
- The control flow of a Python program is regulated by conditional statements, loops, and function calls.



Code:

```python
# ------ if statement ------
# Check x is positive or negative
x = -10
if x < 0:
    x = 0
    print('x is neagative')
elif x == 0:
    print('x = 0')
else:
    print('x is positive')

# ------ for loop ------
# print content of list y using for loop
y = [1,5,10,20]  # create list y

for i in y:
    print('i=',i)

# ------ while loop ------
# count number till 9
count = 0
while (count < 9):
   print('The count is:', count)
   count = count + 1
```

Output

```
x is neagative
i= 1
i= 5
i= 10
i= 20
The count is: 0
The count is: 1
The count is: 2
The count is: 3
The count is: 4
The count is: 5
The count is: 6
The count is: 7
The count is: 8
```

Further reading

- Link for more details : ▸ Link