

# Python environments

---

Dr. Tushar Sandhan

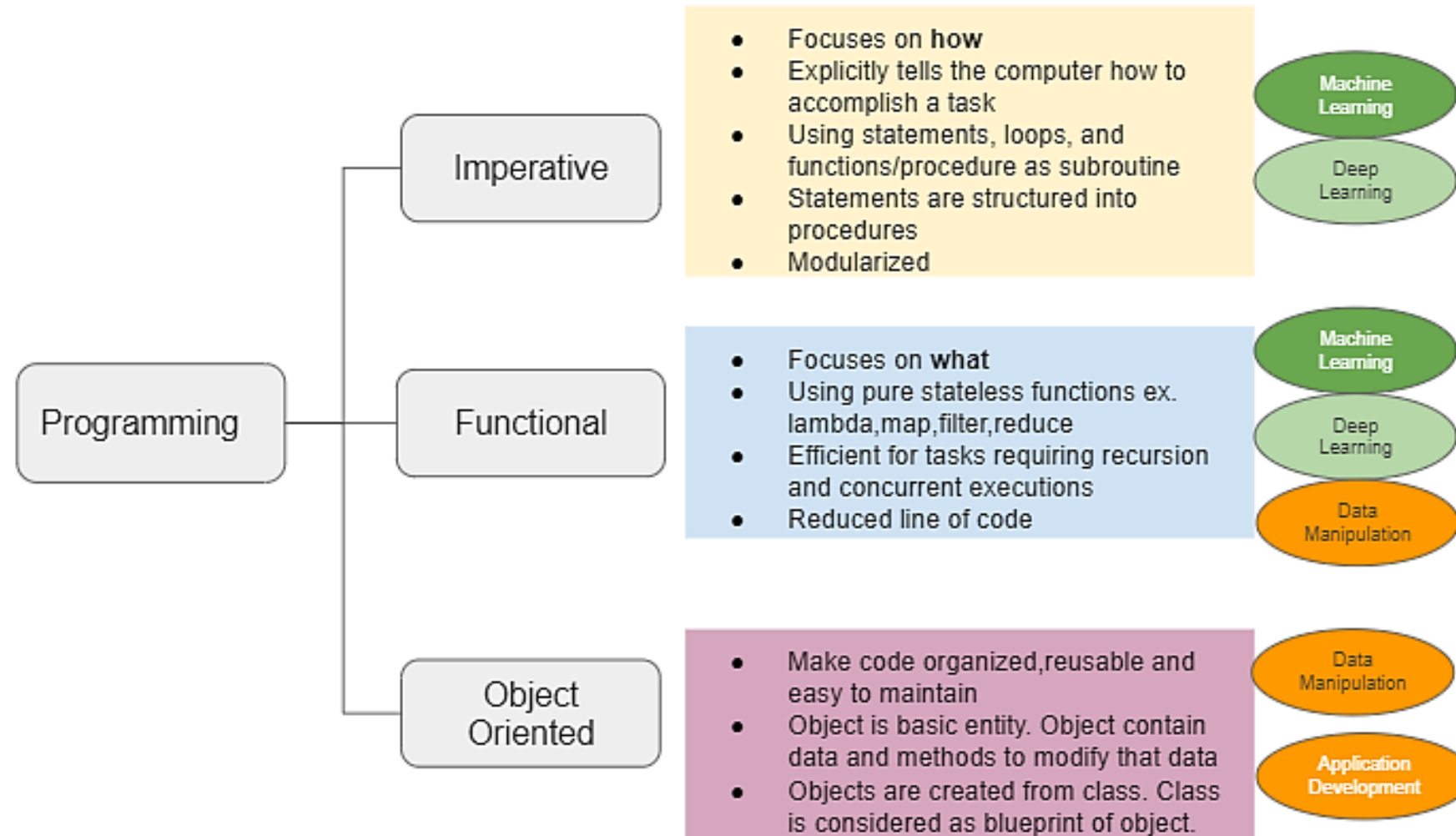
# Introduction

---

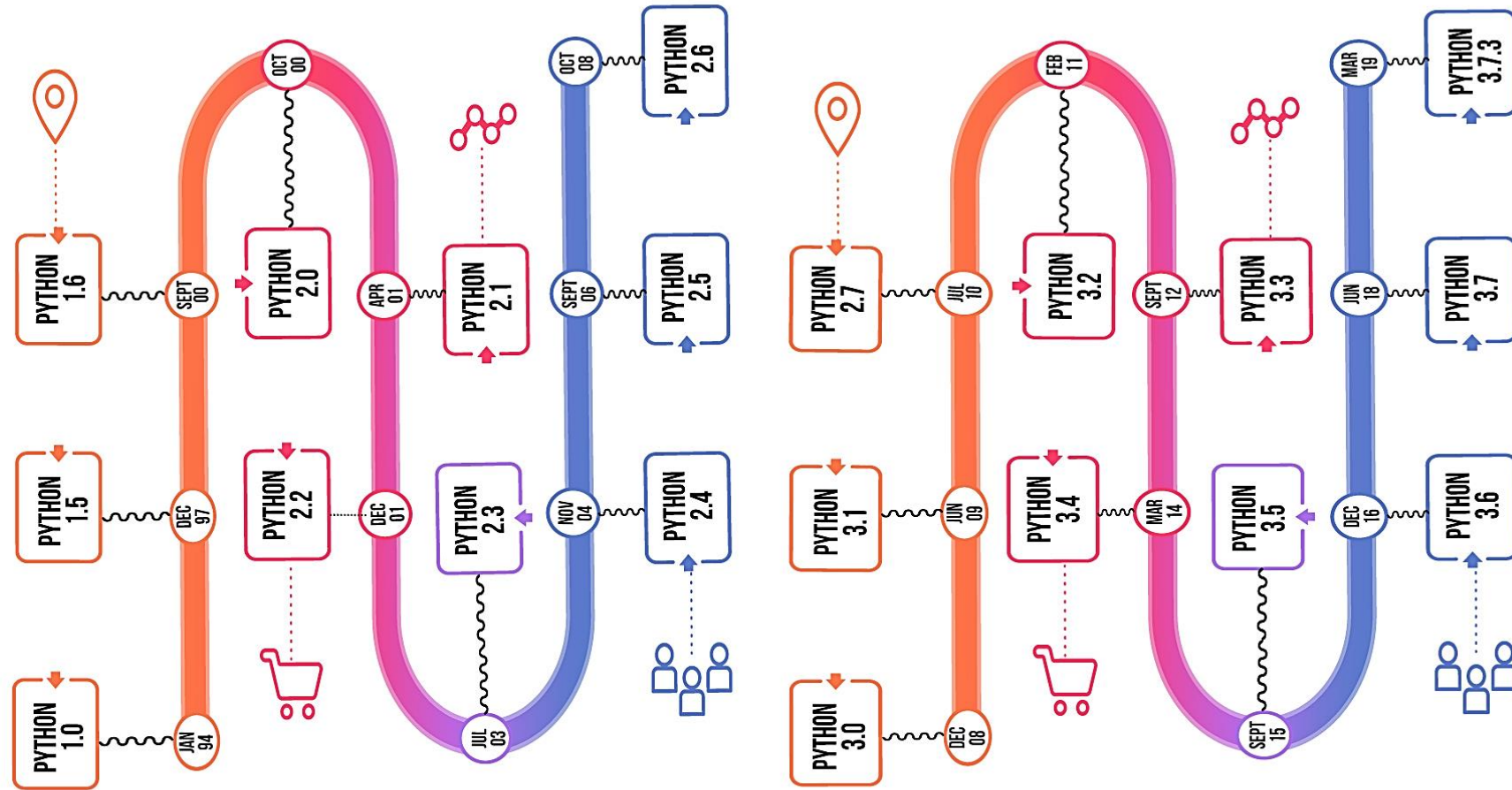
```
>>> Python
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Python' is not defined
>>>
```

- Python
  - Is a modern, general-purpose, object-oriented, high-level programming language
    - Dynamic
    - Expressive
    - Interpreted

# Python

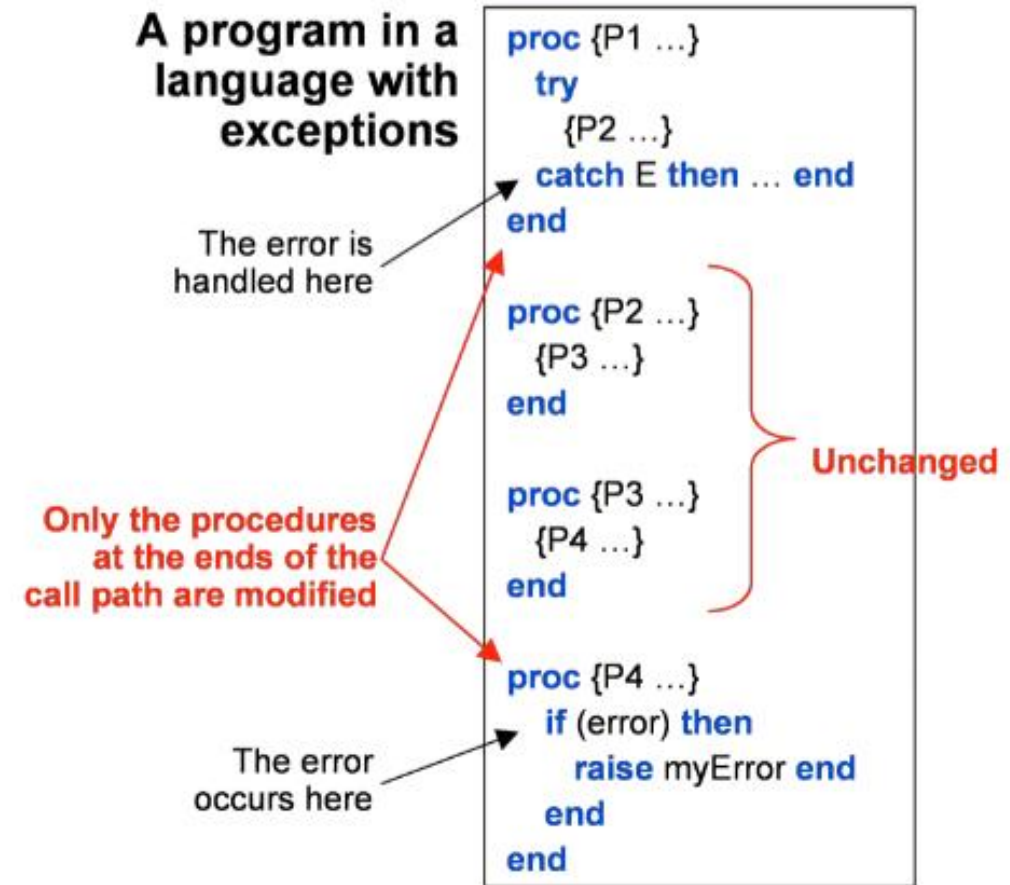
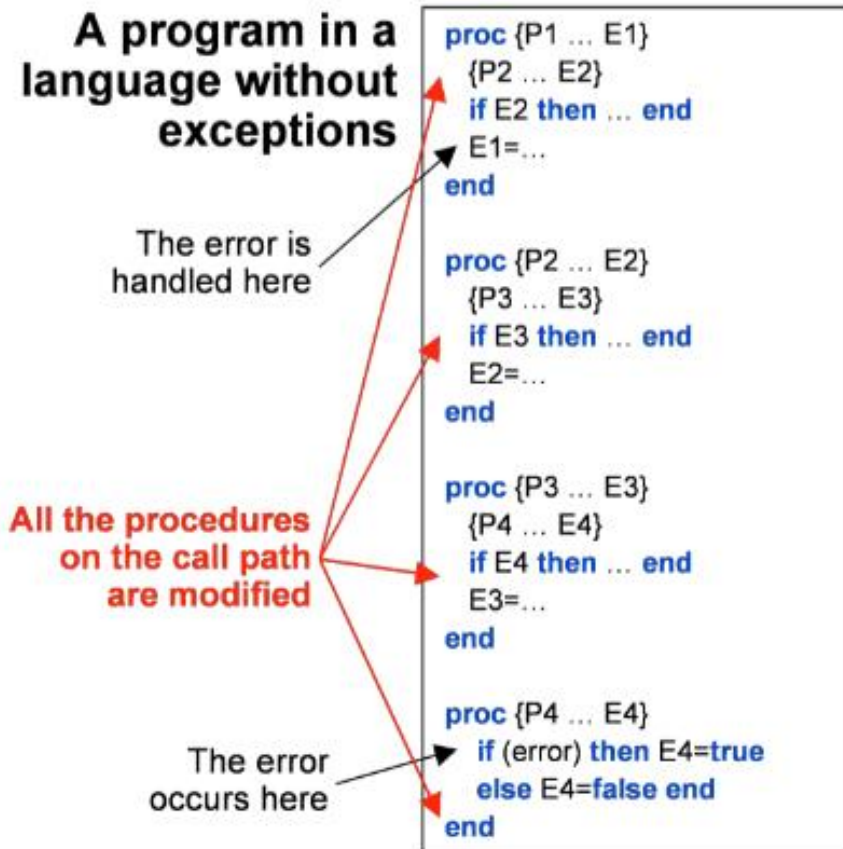


# Python



# Handle exceptions

- Possible in Python



# Python

---



# Python

---

- Extensive ecosystem of scientific libraries and environments
  - numpy: <http://numpy.scipy.org> - Numerical Python
  - scipy: <http://www.scipy.org> - Scientific Python
  - matplotlib: <http://www.matplotlib.org> - graphics library
- Environments where python interpreter can be used
  - Core python installation (Cpython)
  - Ipython (Interactive shell)
  - matplotlib: <http://www.matplotlib.org> - graphics library

# Python

---

- Core python installation (Cpython)
- Ipython (Interactive shell)
- Ipython notebook (HTML based GUI)
- IDE: SW package with inbuilt editor, interpreter (e.g. Spyder)
- Package manager (cross platform)  
e.g. conda

```
(base) C:\Users\sandhan>python
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 20
23, 13:26:23) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more inf
ormation.
>>>
```

```
(base) C:\Users\sandhan>ipython
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 20
23, 13:26:23) [MSC v.1916 64 bit (AMD64)]
Type 'copyright', 'credits' or 'license' for more information
IPython 8.20.0 -- An enhanced Interactive Python. Type '?' fo
r help.

In [1]:
```

```
(base) C:\Users\sandhan>conda
usage: conda-script.py [-h] [-v] [--no-plugins] [-V]
                        COMMAND ...

conda is a tool for managing and deploying applications, envi
ronments and packages.
```



# Python

- IDE:  
(e.g. Spyder)

The screenshot displays the Spyder Python IDE interface. The main editor window shows a Python script named `temp.py` with the following code:

```
1  # -*- coding: utf-8 -*-
2  """
3  Prof. T. Sandhan
4  This is a temporary script file for sorting input array
5  """
6
7  arr = [5, 9, 1, 10, 3, 8, 4, 2, 7, 6]
8  temp = 0
9  max_size = len(arr)
10 print("The elements of the array before sorting: ")
11
12 for i in range(0, max_size):
13     print(arr[i], end=" ")
14
15 print()
16 for i in range(0, max_size):
17     for j in range(i+1, len(arr)):
18         if(arr[i] > arr[j]):
19             temp = arr[i]
20             arr[i] = arr[j]
21             arr[j] = temp
22 print("The elements of the array after sorting: ")
23
24 for i in range(0, max_size):
25     print(arr[i], end=" ")
```

The right-hand pane shows the Variable Explorer, displaying the following variables and their values:

Name	Type	Size	Value
arr	list	10	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
i	int	1	9
j	int	1	9
max_size	int	1	10
temp	int	1	10
test_dict_tour	dict	2	{'a':1, 'b':2}
test_list_tour	list	5	[1, 2, 3, 4, 5]

The bottom pane shows the IPython Console, displaying the output of the script:

```
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.
1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.20.0 -- An enhanced Interactive Python.

In [1]: test_list_tour = [1, 2, 3, 4, 5]

In [2]: test_dict_tour = {'a': 1, 'b': 2}

In [3]: runfile('C:/Users/sandhan/.spyder-py3/temp.py', wdir='C:/Users/
sandhan/.spyder-py3')
The elements of the array before sorting:
5 9 1 10 3 8 4 2 7 6
The elements of the array after sorting:
1 2 3 4 5 6 7 8 9 10
```



C:\Users\sandhan\.spyder-py3

C:\Users\sandhan\.spyder-py3\temp.py

temp.py x

```
1  # -*- coding: utf-8 -*-
2  """
3  Prof. T. Sandhan
4  This is a temporary script file for sorting input array
5  """
6
7  arr = [5, 9, 1, 10, 3, 8, 4, 2, 7, 6]
8  temp = 0
9  max_size = len(arr)
10 print("The elements of the array before sorting: ")
11
12 for i in range(0, max_size):
13     print(arr[i], end=" ")
14
15 print()
16 for i in range(0, max_size):
17     for j in range(i+1, len(arr)):
18         if(arr[i] > arr[j]):
19             temp = arr[i]
20             arr[i] = arr[j]
21             arr[j] = temp
22 print("The elements of the array after sorting: ")
23
24 for i in range(0, max_size):
25     print(arr[i], end=" ")
```

Name	Date Modified
> autosave	15-01-2024 11:25
> config	15-01-2024 11:16
> lsp_paths	15-01-2024 11:16
> plugins	15-01-2024 11:15
> spyder.lock	15-01-2024 11:16
history_internal.py	15-01-2024 11:16
history.py	15-01-2024 11:25
langconfig	15-01-2024 11:15
path	15-01-2024 11:16
pdb_history.sqlite	15-01-2024 11:16
profiler.results	15-01-2024 11:28

Help Variable Explorer Plots Files Profiler Code Analysis

Console 1/A x

```
In [4]: runfile('C:/Users/sandhan/.spyder-py3/temp.py', wdir='C:/Users/
sandhan/.spyder-py3')
File <unknown>:10
print("The elements of the array before sorting: ");;
```

SyntaxError: invalid syntax

```
In [5]: runfile('C:/Users/sandhan/.spyder-py3/temp.py', wdir='C:/Users/
sandhan/.spyder-py3')
The elements of the array before sorting:
5 9 1 10 3 8 4 2 7 6
The elements of the array after sorting:
1 2 3 4 5 6 7 8 9 10
```

IPython Console History

C:\Users\sandhan\.spyder-py3

C:\Users\sandhan\.spyder-py3\temp.py

temp.py x

```
1  # -*- coding: utf-8 -*-
2  """
3  Prof. T. Sandhan
4  This is a temporary script file for sorting input array
5  """
6
7  arr = [5, 9, 1, 10, 3, 8, 4, 2, 7, 6]
8  temp = 0
9  max_size = len(arr)
10 print("The elements of the array before sorting: ")
11
12 for i in range(0, max_size):
13     print(arr[i], end=" ")
14
15 print()
16 for i in range(0, max_size):
17     for j in range(i+1, len(arr)):
18         if(arr[i] > arr[j]):
19             temp = arr[i]
20             arr[i] = arr[j]
21             arr[j] = temp
22 print("The elements of the array after sorting: ")
23
24 for i in range(0, max_size):
25     print(arr[i], end=" ")
```

C:\Users\sandhan\.spyder-py3\temp.py

2024-01-15 11:28:39

Function/Module	Total Time	Diff	Local Time	Diff
<built-in method builtins.print>	12.40 µs		12.40 µs	
<built-in method builtins.len>	500.00 ns		500.00 ns	

Help Variable Explorer Plots Files Profiler Code Analysis

Console 1/A x

```
In [6]: runfile('C:/Users/sandhan/.spyder-py3/temp.py', wdir='C:/Users/sandhan/.spyder-py3')
The elements of the array before sorting:
5 9 1 10 3 8 4 2 7 6
The elements of the array after sorting:
1 2 3 4 5 6 7 8 9 10

In [7]:
```

IPython Console History

conda (Python 3.11.5) Completions: conda LSP: Python Line 25, Col 27 UTF-8 CRLF RW Mem 52%

C:\Users\sandhan\.spyder-py3\temp.py

```

temp.py x
1  # -*- coding: utf-8 -*-
2  """
3  Prof. T. Sandhan
4  This is a temporary script file for sorting input array
5  """
6
7  arr = [5, 9, 1, 10, 3, 8, 4, 2, 7, 6]
8  temp = 0
9  max_size = len(arr)
10 print("The elements of the array before sorting: ")
11
12 for i in range(0, max_size):
13     print(arr[i], end=" ")
14
15 print()
16 for i in range(0, max_size):
17     for j in range(i+1, len(arr)):
18         if(arr[i] > arr[j]):
19             temp = arr[i]
20             arr[i] = arr[j]
21             arr[j] = temp
22 print("The elements of the array after sorting: ")
23
24 for i in range(0, max_size):
25     print(arr[i], end=" ")
26
27 print()
28 print()
29 import keyword
30 print(keyword.kwlist)

```

Name	Type	Size	Value
arr	list	10	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
i	int	1	9
j	int	1	9
max_size	int	1	10
temp	int	1	10
test_dict_tour	dict	2	{'a':1, 'b':2}
test_list_tour	list	5	[1, 2, 3, 4, 5]

[Help](#)
[Variable Explorer](#)
[Plots](#)
[Files](#)
[Profiler](#)
[Code Analysis](#)

```

Console 1/A x
In [9]: runfile('C:/Users/sandhan/.spyder-py3/temp.py', wdir='C:/Users/
sandhan/.spyder-py3')
The elements of the array before sorting:
5 9 1 10 3 8 4 2 7 6
The elements of the array after sorting:
1 2 3 4 5 6 7 8 9 10

['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break',
'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for',
'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or',
'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']

In [10]:

```

[IPython Console](#)
[History](#)

# Python keywords

---

<code>and</code>	A logical operator
<code>as</code>	To create an alias
<code>assert</code>	For debugging
<code>break</code>	To break out of a loop
<code>class</code>	To define a class
<code>continue</code>	To go to the next iteration of a loop
<code>def</code>	To define a function
<code>del</code>	To delete an object
<code>elif</code>	A conditional statements, like else if
<code>else</code>	A conditional statements
<code>except</code>	Used with exceptions, what to do when an exception occurs
<code>False</code>	Boolean value
<code>finally</code>	Used with exceptions, will be executed no matter if there is an exception or not
<code>for</code>	To create a for loop
<code>from</code>	To import specific parts of a module
<code>global</code>	To declare a global variable

<code>if</code>	To make a conditional statement
<code>import</code>	To import a module
<code>in</code>	To check if a value is in a list, tuple
<code>is</code>	To test if two variables are equal
<code>lambda</code>	To create an anonymous function
<code>None</code>	Represents a null value
<code>nonlocal</code>	To declare a non-local variable
<code>not</code>	A logical operator
<code>or</code>	A logical operator
<code>pass</code>	A statement that will do nothing (null)
<code>raise</code>	To raise an exception
<code>return</code>	To exit a function and return a value
<code>True</code>	Boolean value
<code>try</code>	To make a try...except statement
<code>while</code>	To create a while loop
<code>with</code>	Used to simplify exception handling
<code>yield</code>	To end a function, returns a generator



# Python

---

- Expressive
- Interpreted (compilation is hidden from user)
- Cross platform (Win, Linux, Mac)
- Open Source
- Object-Oriented
- Large standard library
- GUI possible
- Integration possible with C, C++, Java,





# Conclusion

- Dynamic
- Interpreted
  - Fast bytecode by CPython
  - Run by PVM

```
>>> Python
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Python' is not defined
>>> pass
>>>
```