PYTHON

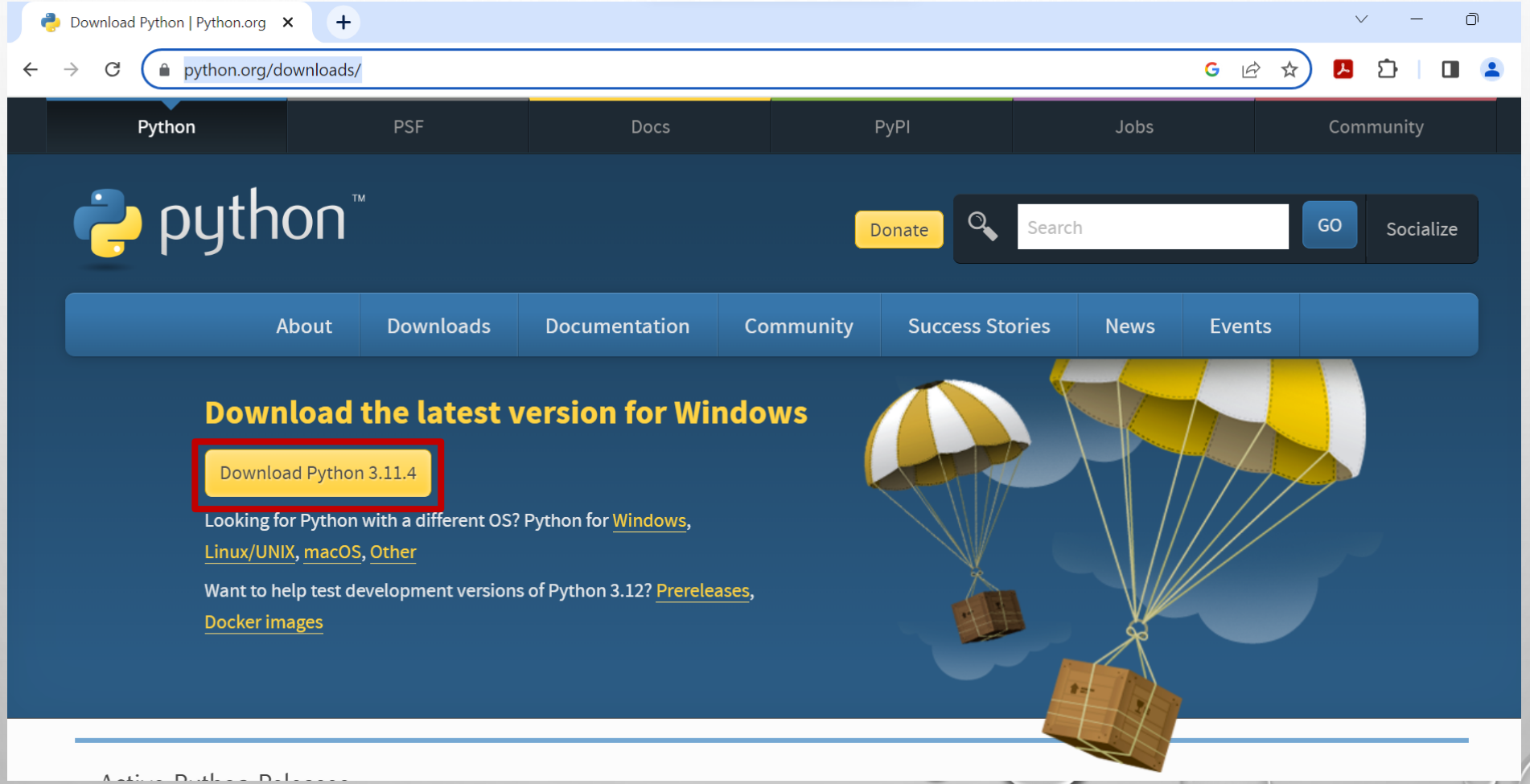
CHAPTER 1

Introduction

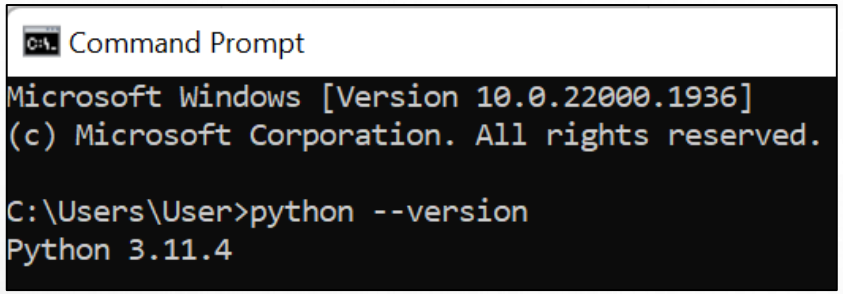
- writing computer programs
  - teaching the computer to do new things
- there are different languages for humans to speak to one another
- there are different programming languages for humans to speak to computers

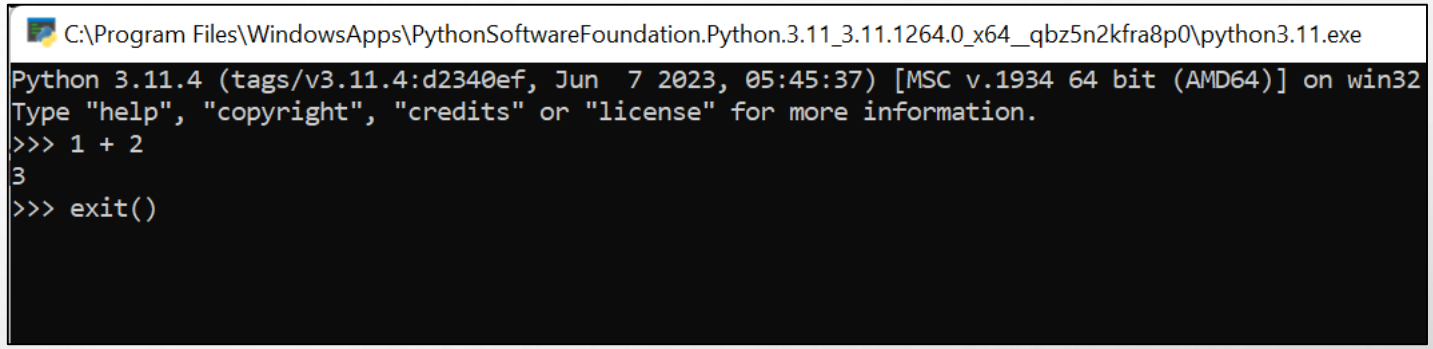
Requirement

- https://www.python.org/downloads/

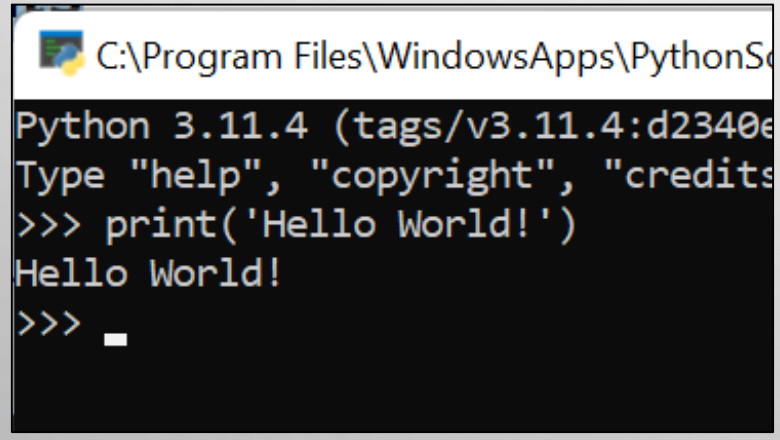


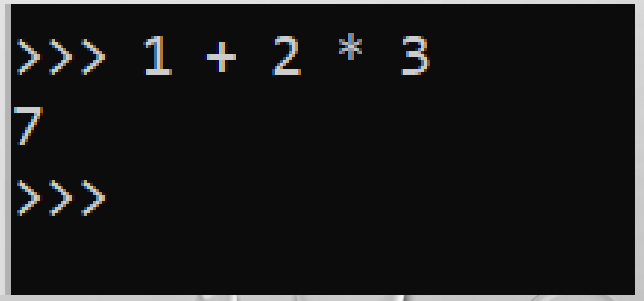
install





- Expressions and statements





```
>>> print('Can't write a single quotation marks')
  File "<stdin>", line 1
    print('Can't write a single quotation marks')
          ^^^^^^
SyntaxError: invalid syntax. Perhaps you forgot a comma?
>>>
```

```
>>> print("Can't write a single quotation marks")
Can't write a single quotation marks
>>>
```

```
>>> print(12)
12
>>>
```

```
>>> 'Hello'
'Hello'
>>> print('Hello')
Hello
>>>
```

```
>>> 1000000000 + 2000000000 * 3000000000
6000000001000000000
>>> _
```

```
>>> 1_000_000_000 + 2_000_000_000 * 3_000_000_000
6000000001000000000
>>> _
```

Operator	Description
a + b	addition
a - b	subtract b from a
a * b	multiplication

Truth and falsity

- either something is true or it is not
- **boolean** named after the English mathematician George Boole (1815–1864)

1. True
2. False

- How can we use these?
- **Comparison**

```
>>> 99 > 100
False
>>>
```

```
>>> 4 + 3 + 2 + 1 == 10
True
>>>
```

Comparison Operators

Operator	Description
a == b	True if ***a*** and ***b*** are equal
a < b	True if ***a*** is less than ***b***
a <= b	True if ***a*** is less than or equal to ***b***
a > b	True if ***a*** is greater than ***b***
a >= b	True if ***a*** is greater than or equal to ***b***
a != b	True if ***a*** is not equal to ***b***

```
>>> 1 == 1 and 10 > 9
True
>>>
```

```
>>> 1 == 1 or 9 > 10
True
>>>
```

```
>>> not 1 == 1
False
>>>
```

```
>>> 1 == 2 or not 9 > 10
True
>>>
```

The types of things

- Three types of data
    1. strings
    2. integers
    3. Booleans

- >>> type('Hello!')

<class 'str'>

- >>> type(25)

<class 'int'>

- >>> type(1 + 2 * 3)

<class 'int'>

- >>> type(True)

<class 'bool'>

- >>>

Common Problems

- When Python **does not recognize** what we type in as a valid program, **an error message** is shown instead of an answer.

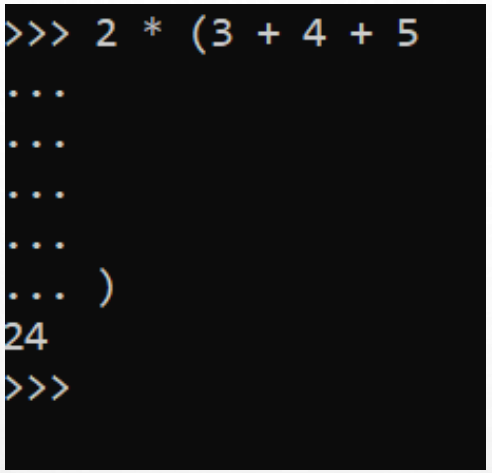
```
>>> print('A string without a proper end)
  File "<stdin>", line 1
    print('A string without a proper end)
          ^
SyntaxError: unterminated string literal (detected at line 1)
>>>
```

```
>>> 2 < '5'
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: '<' not supported between instances of 'int' and 'str'
>>>
```

```
>>> 2 * (3 + 4 + 5
...
...
...
...
...
```

type Ctrl-C

```
>>> 2 * (3 + 4 + 5
...
...
...
...
...
KeyboardInterrupt
>>>
```


>>> 2 * (3 + 4 + 5
...
...
...
...
...)
24
>>>

Summary

- how to interact with Python by typing statements and reading the answers
- learned about three types of data: strings, whole numbers, and Booleans
- how to perform arithmetic on numbers
- how to test things for equality with one another, using operators and operands
- learned the type of something

Thank You