Python 3 Cheat Sheet

Download via https://labs.alta3.com/posters/python3.pdf

Data Types

Integer	22, -10, 99
Float	2.2, 42.3, -1.1
String	"Hello", 'single quotes also works'
Boolean	True, 1, False, 0
List	[value, value value, …]
Tuple	(value, value, value, …)
Dictionary	{key:value, key:value, …}

Operators

`x+y`	Addition
`x-y`	Subtraction
`x*y`	Multiplication
`x/y`	Division
`x%y`	Modulo
`x**y`	Exponentiation

Conditions

`<`	Less than
`>`	Greater than
`<=`	Less than or equal to
`>=`	Greater than or equal to
`==`	Equal
`!=`	Not equal to
`in`	Contains
`not in`	Does not contain

Output

```
print("Hello Alta3 Research")
print("Hello","Alta3 Research")
name = "Alta3 Research"
print("Hello", name)
print("Hello {}".format(name))
print() # generate empty line
        # and a break line
```

Common Functions

int()	Force an integer data type
float()	Force a float data type
str()	Force a string data type
input()	Collect user input
len()	Returns the length
min()	Returns the minimum value
max()	Returns the maximum value
list()	Converts to a list
type()	Returns the data type (string, float, int, etc…)

Error Handling

```
try:
    # this code will execute
except:
    # but on an error
    # this code executes
```

Shebang

```
#!/usr/bin/env python3
```

Defining Functions

```
def a3funct(param1, param2):
    magic = param1 + param2
    return magic
```

if, else if, else

```
if (condition):
    # This code will execute
elif (condition):
    # This code will execute
else:
    # This code executes
```

for and while Loops

```
# for-loop
myIterator = [1,2,3]
for x in myIterator:
    print(x)

# while-loop
x = 0
while x < 5:
    print("x: {}".format(x))
    x += 1 # same as x = x+1
```

Docstrings & Comments

```
"""
Docstrings can be
Multi-line
"""

# Always comment code
```

Methods for Lists

```
mylist = [1,2,3]
mylist.append([4,5,6])
# mylist = [1,2,3,[4,5,6]]

newlist = [1,2,3]
newlist.extend([4,5,6])
# newlist = [1,2,3,4,5,6]
```

Reading and Writing Files

```
# readonly from a file
with open(<path>,'r') as f:
    f.read()
    f.readlines()


# read/write from a file
with open(<path>,'w') as f:
    g.write(<str>)
```