

Files

```
file = open("data.txt") # open file
contents = file.read() # whole file
as string
lines = file.readlines() # all lines
as list
line = file.readline() # one line
for line in file: # loop lines
line = line.strip() # remove \n and
spaces
parts = line.split(",") # split by
comma
file.close() # close file
```

annoying for loops

```
# --- Compact Python Cheat
Sheet with Examples ---
# 1. In-place list update
def threshold_filter(readings,
threshold):
for i in range(len(readings)):
readings[i] = 0 if readings[i] < 0
else (threshold if readings[i] <=
threshold else readings[i])
r = [-3, 0, 2, 8, 15]
threshold_filter(r, 5)
print(r) # [0, 5, 5, 8, 15]
# 2. Count words in order
def word_counter(s):
counts = {}
for w in s.lower().split():
counts[w] = counts.get(w, 0) + 1
return counts
```

annoying for loops (cont)

```
print(word_counter("Hi there hi
There")) # {'hi': 2, 'there': 2}
# 3. Sum list until threshold
(while loop)
def sum_list_until(nums, t):
i, total = 0, 0
while i < len(nums) and total < t:
total += nums[i]
i += 1
return i, total
print(sum_list_until([5,6,6,4],
17)) # (3, 17)
# 4. Running total with labels
def units_available(data):
total = 0
return [(total:=total+change,
label) for change, label in data]
print(units_available([(15,"-
Mon"), (-5,"Tue"), (0,"Wed"))])
# [(15, 'Mon'), (10, 'Tue'), (10,
'Wed')]
# 5. Index courses by keys
def index_course(grades):
d = {}
for i, entry in enumerate(gr-
ades):
d.setdefault(entry['course_id'],
[]).append(i)
return d
grades = [
{'student_id':1,'course_id':'cosc-
121','grade':'A'},
```

annoying for loops (cont)

```
{'student_id':2,'course_id':'cosc-
122','grade':'B'},
{'student_id':3,'course_id':'cosc-
121','grade':'C'}
]
print(index_course(grades)) #
{'cosc121':[0,2], 'cosc122':[1]}
# 6. Dict comprehension
words = ['hello', 'world']
word_lengths = {w: len(w) for w
in words}
print(word_lengths) # {'hello': 5,
'world': 5}
# 7. List comprehension with
condition
thresholded = [x if x > 5 else 5
for x in [-2,4,8,10]]
print(thresholded) # [5, 5, 8, 10]
# 8. Using enumerate
for i, v in enumerate(['a','b','c']):
print(i, v)
# 0 a
# 1 b
# 2 c
# 9. Sorted dict keys output
my_dict = {'b': 2, 'a': 1, 'c': 3}
for k in sorted(my_dict):
print(k, my_dict[k])
# a 1
# b 2
# c 3
```

exceptions

```
except ZeroDivisionError:
except ValueError:
except IndexError:
except TypeError:
```

list slicing

```
nums = [10, 20, 30, 40, 50, 60]
nums[1:4]
[20, 30, 40]
nums[:3]. [10, 20, 30]
nums[-2:]
last 2 items
nums[:-1] all but
last
nums[::2]
every 2nd
item
nums[::-1]. reverse
```

If statement

```
if num % 2 == 0: #even
if num % 2 != 0: #odd
6 / 2 #3.0
7 // 2 #3
7 % 2 #1
3 ** 2 #9
```

matplotlib

```
import matplotlib.pyplot as plt
import numpy as np
# ===== BASIC LINE PLOT
=====
x = np.linspace(0, 10, 100)
y = np.sin(x)
plt.plot(x, y, label='sin(x)') # Line
plot
plt.title("Sine Curve") # Title
plt.xlabel("x-axis") # X label
plt.ylabel("y-axis") # Y label
plt.grid(True) # Show grid
plt.legend() # Show legend
```



By **luckytime**

cheatography.com/luckytime/

Not published yet.

Last updated 18th June, 2025.

Page 1 of 3.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

matplotlib (cont)

```
plt.xlim(0, 10) # Limit x-axis
plt.ylim(-1, 1) # Limit y-axis
plt.xticks([0, 5, 10]) # Custom x
ticks
plt.yticks([-1, 0, 1]) # Custom y
ticks
plt.savefig("sine_plot.png") #
Save to file
plt.show() # Show the plot
# ===== SCATTER PLOT
=====
x = np.random.rand(50)
y = np.random.rand(50)
plt.scatter(x, y)
plt.title("Random Scatter")
plt.show()
# ===== BAR CHART
=====
categories = ['A', 'B', 'C']
values = [10, 20, 15]
plt.bar(categories, values)
plt.title("Bar Chart")
plt.show()
# ===== HISTOGRAM
=====
data = np.random.randn(1000)
plt.hist(data, bins=30)
plt.title("Histogram")
plt.show()
```

matplotlib (cont)

```
# ===== CLEARING /
CLOSING =====
plt.clf() # Clear current figure (if
needed)
plt.close() # Close the plot
window
```

For Loops

```
for i in range(5): print(i) # 0 1 2 3
4
for i in range(2,6): print(i) # 2 3 4
5
for i in range(1,10,3): print(i) # 1
4 7
lst=['a','b','c']
for i in range(len(lst)): print(i,l-
st[i]) # 0 a 1 b 2 c
for item in lst: print(item) # a b c
for i,item in enumerate(lst):
print(i,item) # 0 a 1 b 2 c
```

switching while loop to for loop

```
# For loops
for op1 in range(0, value1):
for op2 in range(value1, value2,
-1):
print(f"{op1:3} x {op2:3} =
{op1*op2:3}")
# Equivalent while loops
op1 = 0
while op1 < value1:
op2 = value1
while op2 > value2:
print(f"{op1:3} x {op2:3} =
{op1*op2:3}")
op2 -= 1
op1 += 1
```

annoying for loop on dict

```
# for key, value in dict.items()
d = {'a':1,'b':2}
for k,v in d.items(): print(k,v)
# Before: d = {'a':1,'b':2}
# After:
# a 1
# b 2
# for word in string.split()
s = " hello world "
for w in s.strip().split(): print(w)
# Before: " hello world "
# After:
# hello
# world
# for i in range(len(list))
lst = ['x','y']
for i in range(len(lst)): print(i,l-
st[i])
# Before: ['x','y']
# After:
# 0 x
# 1 y
# for i in range(start, stop, step)
for i in range(1,6,2): print(i)
# Before: range(1,6,2)
# After:
# 1
# 3
# 5
# for i, val in enumerate(list)
lst = ['a','b']
for i,v in enumerate(lst): print(i,v)
```

annoying for loop on dict (cont)

```
# Before: ['a','b']
# After:
# 0 a
# 1 b
```

Sets and Dictionaries

```
# Sets from list and operations
lst=[1,2,2,3,4,4]
s=set(lst) # {1,2,3,4}
s.add(5) # {1,2,3,4,5}
s.remove(2) # {1,3,4,5}
s.discard(6) # {1,3,4,5} no error
print(3 in s) # True
s2={3,4,5,6}
print(s.union(s2)) # {1,3,4,5,6}
print(s.intersection(s2)) # {3,4,5}
print(s.difference(s2)) # {1}
print(s.issubset(s2)) # False
for x in s: print(x) # 1 3 4 5 (any
order)
# Dictionaries and methods
d={"a":1,"b":2}
print(d.keys()) # dict_keys(['-
a','b'])
print(d.values()) # dict_values(-
[1,2])
print(d.items()) # dict_items(['-
('a',1),('b',2)])
print(d.get("a")) # 1
print(d.get("z",0)) # 0 default if
missing
```



By **luckytime**

cheatography.com/luckytime/

Not published yet.

Last updated 18th June, 2025.

Page 2 of 3.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

Sets and Dictionaries (cont)

```
d["c"]=3
for k in d: print(k) # a b c
for v in d.values(): print(v) # 1 2
3
for k,v in d.items(): print(k,v) # a
1 b 2 c 3
```

Lists

```
fruits = ["apple",    fruits[0]
"banana", "che-    #apple
rry"]

fruits[-1] #cherry  len(fruits)
                    #3

fruits.append("k-    fruits.in-
iwi")              sert(1, "pea-
r")

fruits.remove("b-    fruits.pop()
anana")            #removes
                    last

fruits.sort()      sorted(fr-
                    uits)
```



By **luckytime**

cheatography.com/luckytime/

Not published yet.

Last updated 18th June, 2025.

Page 3 of 3.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>