

Python 1 Cheat Sheet

Python for Biologists · Session 1

Variables · Data types · Printing · Operators · Comparisons · Built-ins

Variables

A name that holds a value: **name = value**

```
# store values under a label
gene    = "TP53"
gc       = 0.48
length  = len(gene)

# naming: snake_case, start with a
# letter, be descriptive (gc_count
# not x). Reassigning is fine.
```

Data types

Every value has a type — check it with **type(x)**

```
int      20000      # whole number
float    0.48       # decimal
str      "ATGC"     # text in quotes
bool     True / False # yes / no

type(gc)  # -> <class 'float'>
```

Printing & f-strings

Show a result on screen

```
print("hello")
print(gene, length) # space between

# f-string: drop values into text
print(f"{gene}: GC={gc:.1%}")
# -> TP53: GC=48.0%

f"{x:.2f}" # 2 decimals
f"{n:,}"  # 1,000s separator
```

Operators — numbers

```
2 + 3    # 5    add
10 - 4   # 6    subtract
6 * 7    # 42   multiply
9 / 2    # 4.5  divide (float)
9 // 2   # 4    integer divide
9 % 2    # 1    remainder
2 ** 8   # 256  power
```

Operators — strings

```
"AT" + "GC"      # 'ATGC'  join
"AT" * 3         # 'ATATAT' repeat
len("ATGC")      # 4       length
"ATGC".count("G") # 1       count

# mixing types crashes:
"2" + 3          # TypeError
```

Comparisons → bool

Every comparison returns **True** or **False**

```
gc == 0.48      # equal?
gc != 0         # not equal?
gc > 0.5        # greater / less: > < >= <=

# combine with and / or / not
0.4 <= gc <= 0.6 # in range?
```

Built-in functions

```
len(x)          # length / count
round(x, 2)     # round to 2 dp
type(x)         # the type of x
print(...)      # show value(s)
int(x) float(x) str(x)
# convert between types
```

Reading errors

Read the **last line first** — it names the problem

```
NameError      # used a name you
                  # never defined (typo?)
SyntaxError   # a typo in the code
                  # structure itself
TypeError     # wrong type, e.g.
                  # "2" + 3
```