Types

str

```
a = "hello!"
a.upper()       # HELLO!
a.capitalize()  # Hello!
a.strip("!")    # hello
a.index("e")    # 2
a.split("e") # ["h", "llo!"]
```

number types

```
count = 3  # int
pi = 3.14  # float
```

list

```
a = ["a", "b", 3]
a[0]         # "a"
a[1]         # "b"
a[-1]        # "3"
a[1:2]       # ["b", 3]
```

tuple same as list, but immutable

```
a = ("a", "b", 3)
```

dict

```
a = {"test": 1, "b": "hello"}
a["test"]      # 1
a["b"]         # "hello"
del a["test"]  # delete "test"
a["c"] = 3  # add "c" to dict
```

sets same as dicts, but no values & can do arithmetic

```
a = {"a", 1, 4, "b"}
b = {"a", "b"}
print(a - b)  #  {1, 4}
```

list methods

```
a = ["a", "b", 3]
a.append(4) # ["a", "b", 3, 4]
a.reverse() # [4, 3, "b", "a"]
```

dict methods

```
a = {"a": 1, "b": 2}
a.get("c", 3) # 3 as default
a.update({"d": 4}) # add more
a.keys()   # iterable of keys
a.values() # ... of values
a.items()  # ... of both
```

Input/Output

Prompting user

```
name = input("Name? ")
print("Hi ", name)
```

Reading text from file

```
a = open("file.txt").read()
print("file1.txt has: ", a)
```

Writing to file

```
a = "Some text for o.txt"
open("o.txt", "w+").write(a)
```

Branching

Basic if Optionally execute indented code based on the truth value of the condition

```
if cost < 10:
    print("impulse buy")
```

Boolean operators "and", "or"

```
if age > 17 and place == "UK":
    print("can buy alcohol")
if age < 18 or s == "student":
    print("can get discount")
```

If-elif-else

```
if beer == "Darkwing":
    print("IPA")
elif beer == "Hefe":
    print("Hefeweizen")
elif beer == "Stonehenge":
    print("Stout")
else:
    print("Unknown beer")
```

Pass placeholder that does nothing

```
if cost > 1.99:
    pass # TODO: finish this
```

Iteration

For loop Execute the indented code for each item in a list or other "iterable", temporarily putting that item in a given variable

```
names = ["John", "Paul", "G"]
for name in names:
    print("name:", name)
```

Range for-loop Useful for looping through numbers

```
for x in range(0, 100):
    print("x:", x)
```

While loop Repeat indented code until condition is no longer true

```
i = 2
while i < 10000:
    print("square:", i)
    i = i ** 2
```

Interruption Exit loops prematurely with `break`, skip to next iteration with `continue`

```
for i in range(0, 50):
    choice = input("quit/skip? ")
    if choice == "quit":
        break
    elif choice == "skip":
        continue
    print("i", i, "i^2", i ** 2)
```

Functions

Positional parameters

```
def add(a, b):
    c = a + b
    print("the sum is", c)
add(1, 2)
```

Keyword parameters

```
def greet(name="Jack"):
    print("Hello", name)
greet(name="Jill")
```

Return value

```
def in_file(name):
    path = "./src/" + name
    return path + ".html"
path = in_file("home")
html = open(path).read()
```

Comment aka "docstring"

```
def plural(word):
    """
    Return the plural of
    an English word.
    """
    if word.endswith("s"):
        return word + "es"
    return word + "s"
print("Many", plural("cat"))
```

Lambda alternative syntax for one-liners

```
cubed = lambda i: i ** 3
print("5^3 is ", cubed(5))
```