typehinting:

def fn(self, x: int, s: str = 'abc') -> None:

y: int = x + ord(s[0])

print(y // 2)

<https://www.youtube.com/watch?v=yScuF1UgGU0&ab_channel=KieCodes>

<https://docs.python.org/3.7/library/typing.html>

typing:

mypy scrip.py (to check for typing issues)

Optional[X] = Union[X, None]

Callable[[Arg1Type, Arg2Type], ReturnType]

List[X], Tuple[X], Dict[key, value], Generator …

Generics:

from typing import Generic, TypeVar  
  
T = TypeVar('T', int, float) # T = TypeVar('T', bound=Any)  
  
  
class Test(Generic[T]):  
  
 def \_\_init\_\_(self, value: T):  
 self.value = value  
  
 def check\_type(self, x: T):  
 print(x)  
  
  
t1 = Test(5) # Test[float](5) to make T=float  
t1.check\_type(2.5)  
t2 = Test(5.6)  
t2.check\_type(3)  
t3 = Test('d')  
t3.check\_type(3)

**mypy results:**

other\main.py:17: error: Argument 1 to "check\_type" of "Test" has incompatible type "float"; expected "int"

other\main.py:20: error: Value of type variable "T" of "Test" cannot be "str"

other\main.py:21: error: Argument 1 to "check\_type" of "Test" has incompatible type "int"; expected "str"

Found 3 errors in 1 file (checked 1 source file)