Python Typing Cheat Sheet: Any & Callable

Any

`Any` means the value can be of any type:

Example:

```
from typing import Any


def show(value: Any):
    print(value)
```

Usage:

```
show(123)        #  int
show("hello")    #  str
show([1, 2, 3])  #  list
```

Callable

`Callable` means a function or callable object.

Basic:

```
from typing import Callable


def run(action: Callable[[], None]):
    action()


def greet():
    print("Hello")


run(greet)
```

Detailed:

```
Callable[[int], str]    # function that takes an int and returns a str
Callable[[Any], Any]    # takes any type, returns any type
```

Python Typing Cheat Sheet: Any & Callable

Combined Example

```
from typing import Callable, Any


def process(func: Callable[[Any], Any], value: Any) -> Any:
    return func(value)


def shout(text: str) -> str:
    return text.upper() + "!"


print(process(shout, "hello"))  # Output: HELLO!
```

Summary

```
+-------------------------------+
|        typing hints           |
+-------------------------------+
| Any        -> any type        |
| Callable   -> a function type |
+-------------------------------+

Callable[[], None]      -> no args, returns None
Callable[[int], str]    -> takes int, returns str
Callable[[Any], Any]    -> takes any, returns any
```