

Cheatsheet 01: Variables & Datatypes, Operations, Indexing and Strings

Variables & Datatypes

```
# Variables
x = 5          # integer
y = 3.14       # float
name = "Alice" # string
is_active = True # boolean

# Data Types
int(5)         # integer
float(3.14)    # float
str("Alice")   # string
bool(True)     # boolean
list([1, 2, 3]) # list
tuple((1, 2, 3)) # tuple
dict({"key": "value"}) # dictionary
set({1, 2, 3}) # set
```

Operations

```
# Arithmetic
x + y  # addition
x - y  # subtraction
x * y  # multiplication
x / y  # division
x % y  # modulo
x ** y # exponentiation
x // y # floor division

# Assignment Operators
x += y # x = x + y
x -= y # x = x - y
x *= y # x = x * y
x /= y # x = x / y
x %= y # x = x % y
x **= y # x = x ** y
```

```

x //= y # x = x // y

# Comparison
x == y # equal
x != y # not equal
x < y  # less than
x > y  # greater than
x <= y # less than or equal to
x >= y # greater than or equal to

# Logical
True and False # logical AND
True or False  # logical OR
not True       # logical NOT

```

Indexing

```

my_list = [0, 1, 2, 3, 4]

my_list[0]    # access first element
my_list[-1]   # access last element
my_list[1:4]  # access elements from index 1 to 3
my_list[:3]   # access elements from index 0 to 2
my_list[2:]   # access elements from index 2 to the end
my_list[::2]  # access every second element

```

Strings

```

my_str = "Hello, World!"

len(my_str)          # length of the string
my_str[0]            # access first character
my_str[-1]           # access last character
my_str[7:12]         # access characters from index 7 to 11
my_str.lower()       # convert to lowercase
my_str.upper()       # convert to uppercase
my_str.split(", ")   # split string into list of substrings
my_str.find("World") # find the index of the first occurrence of "World"

```

```
my_str.startswith("H")# check if string starts with "H"
my_str.endswith("!") # check if string ends with "!"
my_str.replace("World", "Python") # replace "World" with "Python"
", ".join(["A", "B", "C"]) # join list of strings with ", "
" Hello ".strip() # remove leading and trailing whitespace
f"Hello, {name}!" # formatted string
```