Data Analytics Track

Python Basics Cheat Sheet

Data Types & Variables

Types

```
type()            # asking for type
type('le wagon') # <class 'str'>
type(42)          # <class 'int'>
type(3.14)        # <class 'float'>
type(True)        # <class 'bool'>
      /False
type(None)        # <class 'NoneType'>
```

Variables Operations

```
x = 5             # assignment
+, -, *, /        # arithmetic operators
//                # floor division
%                 # modulus
**                # exponent
```

Booleans Resulting

```
and, or, not, in # logical operators
==, !=           # comparison operators
<, <=, >, >=
```

Strings

```
n =  'Tanguy'
hello = 'Hello' + n    # concatenation
pres = f"Name's {n}"   # formatting
```

Lists

```
type([1, 2])              # => <class 'list'>
[]                        # empty list
name = ['le', 'wagon']    # list assignment
```

CRUD on list

```
name.append(new_val)          # create
name[index_n]                 # read
name[index_n] = updated_val   # update
del(name[index_n])            # delete
```

Slices

```
str, list    # not working with dict
seq[i0: i1]  # items from i0 to i1 excluded
seq[i0:]     # all items starting from i0
seq[: i1]    # all items up to i1 excluded
seq[:]       # copy of seq
             # index can be negatives
```

Dictionaries

```
type({'n': 'Tanguy'})   # <class 'dict'>
{}                      # empty dict
id = {'age': 26,        # dict assignment
      'n': 'Tanguy'}
```

CRUD on dict

```
id[key] = new_val              # create
id[key]                        # read
id[key] = updated_val          # update
del(id[key])                   # delete
```

Loops

Conditional Loop

```
while condition:
    # code executed each iteration
```

Perusing Loop

```
for item in sequence:
    # item = each element of sequence
    # code executed each iteration
```

Sequences & Sequence Methods

```
len(sequence)     # size of sequence
str, list, dict   # sequences
range(end_n)      # start & step optional
range(start_n, end_n, step_n)
enumerate(list)   # index, element
dict.items()      # key, value
dict.keys()       # keys as list
                  # dict default in for loop
dict.values()     # values as list
```

Control Flow

```
if condition:
    # code executed if condition is True
elif other_condition:
    # code executed if condition is False
    # but other_condition is True
else:
    # code executed if no condition is True
```

Functions

```
def fun_name(param0, param1, …):
    # code executed upon call on
    return output # optional

fun_name(arg0, arg1, …) # call
```

le wagon
for Business