Python Crash P1

- **DataTypes**
  - Data Types
    - Numbers
    - Strings
    - Print Formatting
    - Lists
    - Dictionaries
    - Booleans
    - Tuples and Sets
- **Constructs**
  - Comparison Operators
  - If, elif, and else Statements
  - For Loops
  - While Loops
  - range()
  - List Comprehension
  - Functions
  - Lambda Expressions
  - Map and Filter
- **Data Types 1**
  - Number
    - Number - Int & Float

```
In [1]: 1
Out[1]: 1

In [2]: 1.0
Out[2]: 1.0
```

    - Modulus or Mod Function - %
    - Operations & order = BODMAS (, **, /, *, +,-)

- Variable Assignment =

```
In [13]: var = 2

In [14]: var
Out[14]: 2

In [15]: x = 2
         y = 3

In [16]: x + y
Out[16]: 5
```

  - Variables cannot start with number or special character
- Strings
  - Single Quote or Double Quote

```
In [29]: # Strings

In [30]: ' single quote'
Out[30]: ' single quote'

In [31]: "this is a string"
Out[31]: 'this is a string'
```

  - SingleQuote in Double Quote

```
In [32]: "I can't go"
Out[32]: "I can't go"
```

- Printing
  - Print('Hello')
  - Print(x)
  - Print('formated string {}.format(stringvar))
  - Print('formated string {one}.format(one=stringvar))
- Indexing Strings
  - s='hello' s[0]

- Slice Notation: s[0:] hello

```
In [44]: s = 'abcdefghijk'

In [46]: s[0:]
Out[46]: 'abcdefghijk'

In [47]: s[:3]
Out[47]: 'abc'
```

- s[:3] hell
- s[1:3] ell

Data Types 2

- List (Sequence of elements separated by Commas)
  - Append

```
In [52]: my_list = ['a','b','c']

In [53]: my_list.append('d')

In [54]: my_list
Out[54]: ['a', 'b', 'c', 'd']
```

  - Indexing is same as string. Even Slice Notation works

```
In [54]: my_list
Out[54]: ['a', 'b', 'c', 'd']

In [55]: my_list[0]
Out[55]: 'a'
```

  - Nested List

```
In [59]: nest = [1,2,[3,4]]

In [60]: nest
Out[60]: [1, 2, [3, 4]]

In [62]: nest[2][1]
Out[62]: 4
```

- Dictionaries - key Value pair

```
In [3]: d = {'key1':'value','key2':123}

In [5]: d['key1']
Out[5]: 'value'
```

- List Within Dictionary

```
In [7]: d = {'k1':[1,2,3]}

In [11]: my_list = d['k1']

In [13]: my_list[0]
Out[13]: 1
```

- Nested Dictionary

```
In [15]: d = {'k1':{'innerkey':[1,2,3]}}

In [18]: d['k1']['innerkey'][1]
Out[18]: 2
```

- Boolean - True or False
- Tuples
  - Instead of [ ] Tuples use ( )

```
In [23]: t = (1,2,3)

In [24]: t[0]
Out[24]: 1
```

  - Tuples are immutable

```
In [33]: t[0] = 'NEW'
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-33-97e4e33b36c2> in <module>()
----> 1 t[0] = 'NEW'

TypeError: 'tuple' object does not support item assignment
```

- Sets
  - Collection of Unique Elements

```
In [34]: {1,2,3}
Out[34]: {1, 2, 3}

In [35]: {1,1,1,2,2,2,3,3,3}
Out[35]: {1, 2, 3}
```

  - Set from a List

```
In [36]: set([1,1,1,1,2,2,2,5,5,5,6,6,6])
Out[36]: {1, 2, 5, 6}
```

- Add an element to Set

```
In [37]: s = {1,2,3}

In [38]: s.add(5)

In [39]: s
Out[39]: {1, 2, 3, 5}
```