TUPLES

- A tuple in Python is similar to a list.
- The difference between the two is that we cannot change the elements of a tuple
- Once it is assigned whereas we can change the elements of a list.

In [1]:
```
my_tuple=()
print(my_tuple)
```

```
()
```

In [2]:
```
pgr=("hello how are you")
print(pgr)
```

```
hello how are you
```

INDEX

- We can use the index operator [] to access an item in a tuple, where the index starts from 0.

In [4]:
```
pgr=("G","E","E","T","H","A")
```

In [9]:
```
print(pgr[0])
print(pgr[1])
print(pgr[2])
print(pgr[3])
print(pgr[4])
print(pgr[5])
```

```
G
E
E
T
H
A
```

NEGATIVE INDEX

- Python allows negative indexing for its sequences.

In [11]:
```
pgr=("G","E","E","T","H","A")
```

In [12]:
```
print(pgr[-1])
```

```
A
```

In [15]:
```
print(pgr[-0])
print(pgr[-4])
print(pgr[-5])
print(pgr[-3])
print(pgr[-2])
print(pgr[-1])
```

```
G
E
E
T
H
A
```

SLICING

We can access a range of items in a tuple by using the slicing operator colon :.

In [17]:
```
pgr=("G","E","E","T","H","A")
```

In [18]:
```
print(pgr[1:4])
```

```
('E', 'E', 'T')
```

In [19]:
```
print(pgr[1:5])
```

```
('E', 'E', 'T', 'H')
```

In [20]:
```
print(pgr[0:5])
```

```
('G', 'E', 'E', 'T', 'H')
```

In [21]:
```
print(pgr[0:6])
```

```
('G', 'E', 'E', 'T', 'H', 'A')
```

COUNT

The count() method returns the number of times a specified value appears in the tuple.

In [30]:
```
pgr=(8,6,8,8,9,0,8,6,9,4)
```

In [43]:
```
pgr = (8, 6, 8, 8, 9, 0, 8, 6, 9, 4)
p = pgr.count(8)
print(p)
```

```
4
```

LEN

Return the number of elements in the tuple.

In [44]:
```
pgr=(8,6,8,8,9,0,8,6,9,4)
print(len(pgr))
```

```
10
```

In [45]:
```
pgr=("G","E","E","T","H","A")
print(len(pgr))
```

```
6
```

Min

In [ ]:
```
returns the elements from the tuple with minimum value.
```

In [50]:
```
pgr=("G","E","E","T","H","A")
min(pgr)
```

Out[50]:
```
'A'
```

MAX

In [ ]:
```
returns the elements from the tuple with maximum value.
```

In [51]:
```
pgr=("G","E","E","T","H","A")
max(pgr)
```

Out[51]:
```
'T'
```

SUM

- Python sum () function is a built-in function that returns the sum of all numerical values provided in an iterable.
- The numerical values that are passed in the function can be integer and floating-point numbers as well.
- In Python, one of the most used functions is the sum.

In [54]:
```
pgr=(8,6,8,8,9,0,8,6,9,4)
result= sum(pgr)
print(result)
```

```
66
```

SORT

The sort() method sorts the items of a list in ascending or descending order.

In [60]:
```
def pgr(p):
  return p['year']

cars = [
  {'car': 'Ford', 'year': 2005},
  {'car': 'Mitsubishi', 'year': 2000},
  {'car': 'BMW', 'year': 2019},
  {'car': 'VW', 'year': 2011}
]

cars.sort(key=pgr)
print(cars)
```

```
[{'car': 'Mitsubishi', 'year': 2000}, {'car': 'Ford', 'year': 2005}, {'car': 'VW', 'year': 2011}, {'car': 'BMW', 'year': 2019}]
```

In [ ]:
```
```