# Python Strings. Lists. Dictionaries

Key points and links out to more information

## Strings

A string is typically a piece of text. Most importantly, it is not a number. So it have maths functions applied to it. There are a list of [methods](https://www.w3schools.com/python/python_ref_string.asp) that are specifically for use with strings

You can put lots of these methods together in one line to do lots of different things to a string:

### Some useful methods

print(mystring)

hey guys how are you doing

In [36]:

mystring.count("e")

Out[36]:

2

In [12]:

mystring.replace("guys", "girls")

Out[12]:

'hey girls how are you doing'

In [13]:

mystring.upper()

Out[13]:

'HEY GUYS HOW ARE YOU DOING'

In [14]:

mystring.lower()

Out[14]:

'hey guys how are you doing'

In [15]:

mystring.capitalize()

Out[15]:

'Hey guys how are you doing'

In [16]:

mystring.title()

Out[16]:

'Hey Guys How Are You Doing'

Indexing:

mystring[2]

Out[16]:’y’

## Lists

Lists are lists of callable objects of any type. (string, int, float)

Come in square brackets:   
ta = ["Null", "asd", 404]

Can be indexed like a string:  
ta[0] = ‘Null’

ta[‘Null’] = 0

ta[1][2] = ‘d’

ta.sort() = [‘404’,‘asd’,’Null’)

### Iterating through a list

for num in num\_list:

print(f"{num}% done...")

More here

C:\Users\iLoveyou\Week1\Week2.Python Fundamentals\Lists

## Dictionaries

Dictionaries are lists of paired objects. With a key and value

ages = {"Flo":100,

"Jonas":3,

"Kosta":310,

"Sian":22

}

In: Ages

Out: {'Flo': 100, 'Jonas': 3, 'Kosta': 310, 'Sian': 22}

ages["Kosta"] = 310

Can also have dictionaries within dictionaries:

teachers\_dict = {

"Flo":{

"age":100,

"role":"LT",

"species":"giganotosaurus\_crocodile"

},

"Jonas":{

"age":3,

"role":"PM",

"species":"titanosauria\_caiman"

},

"Kosta":{

"age":310,

"role":"TA",

"species":"triceratops\_alligator"

},

"Sian":{

"age":22,

"role":"LT",

"species":"plateosaurus\_imperator"

}

}

teachers\_dict["Jonas"]["species"] = 'titanosauria\_caiman'