

Ayudantía 4

Profesor: Carlos Alvarado

Ayudante: Pablo González

Actividad	Objetivo
Repaso	<ul style="list-style-type: none">• Listas.• Diccionarios.• Loops.
Ejercicios con diccionarios	<ul style="list-style-type: none">• Loops en diccionarios.• Aplicando el método items()
Ejercicios tipo tareas	<ul style="list-style-type: none">• Ejercicios Kattis.• Propuestos.
Cierre	<ul style="list-style-type: none">• Ayudantía / Stata / Tarea.

Repaso

Listas

In [27]:

```
mi_lista = [0,1,2,3,4,5,6,7,8,9]
cuadrado = []
cubo = []
for num in mi_lista:
    cuadrado.append(num**2)
    cubo.append(num**3)
print(mi_lista, '\n', cuadrado, '\n', cubo)
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
[0, 1, 8, 27, 64, 125, 216, 343, 512, 729]
```

In [4]:

```
notas = [['Micro I', 45.4, 'Macro II', 50.0, 'Economía y Ciencia de los datos', 70, 'Ética', 10, 'C
alculo IV', 51] ,
         ["Micro II", 50, "Conta II", 45, "Macro Internacional", 45, "OFG", 68, "OPR", 60],
         ["Competencia", 46, "Cyt", 50, "Mercados", 50, "OFG", 60, "OPR", 65],
         ["Teoría Micro", 46, "Teoría Macro", 52, "Eco Mate", 48, "Análisis Real", 39.5]
        ]
print(notas[0] , "\n" , notas[3][-1])
print(notas[2][4:])
notas.remove(notas[0])
print(notas)
```

```
['Micro I', 45.4, 'Macro II', 50.0, 'Economía y Ciencia de los datos', 70, 'Ética', 10, 'Calculo I
V', 51]
39.5
['Mercados', 50, 'OFG', 60, 'OPR', 65]
[['Micro II', 50, 'Conta II', 45, 'Macro Internacional', 45, 'OFG', 68, 'OPR', 60],
['Competencia', 46, 'Cyt', 50, 'Mercados', 50, 'OFG', 60, 'OPR', 65], ['Teoría Micro', 46, 'Teoría
Macro', 52, 'Eco Mate', 48, 'Análisis Real', 39.5]]
```

Diccionarios

In [26]:

```
empty_dict = {} #Para vaciar un diccionario.
empty_dict2 = dict() #Otra forma (menos estándar)
notas = { "Nicolas" : 80, "Juan" : 95 }
nota_nicolas = notas["Nicolas"]
print(nota_nicolas)
```

80

In [29]:

```
#Qué pasaría si intentamos obtener un valor que no existe en nuestro diccionario?
try:
    nota_camila = notas["Camila"]
except KeyError:
    print("No tiene nota")
```

No tiene nota

In [31]:

```
#Podemos comprobar quienes están en el diccionario.
intento1 = "Nicolas" in notas #True
intento2 = "Camila" in notas #False
print(intento1,intento2)
```

True False

In [45]:

```
#También podemos alterar los valores de nuestros diccionarios
notas["Nicolas"] = 90 #Cambiamos la nota de Nicolas
notas["Camila"] = 95 #Agregamos a Camila
num = len(notas) #¿Cuántos estudiantes tenemos?
print("La nota de Nicolás es: " + str(notas["Nicolas"]))
print("Contamos con" , num , "estudiantes") #Ahora tenemos a tres estudiantes.
```

La nota de Nicolás es: 90
Contamos con 3 estudiantes

In [2]:

```
tweet = {
    "usuario" : "Alan Brito",
    "texto" : "Data Science is Awesome",
    "caracteres" : 100,
    "hashtags" : ["#data", "#science", "#datascience", "#awesome"]
}
```

In [4]:

```
tweet_keys = tweet.keys() # list de llaves
tweet_values = tweet.values() # list of values
tweet_items = tweet.items()
print(tweet_keys)
"usuario" in tweet_keys
"usuario" in tweet
"Alan Brito" in tweet_values
```

dict_keys(['usuario', 'texto', 'caracteres', 'hashtags'])

Out[4]:

True

Ejercicios

Ejercicio 1:

Utilizando el diccionario "acciones", imprima el nombre y valor de cada uno de estos activos utilizando un loop.

In []:

```
def main():
    acciones = {
        'Google': 146.48,
        'UC': 44.11,
        'Tesla': 67
    }
    #Imprimamos cuáles son las llaves
    #for c in acciones:
    #    print(c)

    #Imprimamos las llaves y los valores
    for k, v in acciones.items():
        print("Nombre : {0}, Valor : {1}".format(k, v))

if __name__ == '__main__':
    main()
```

Ejercicio 2

En base a los dos diccionarios provistos, calcule el valor del inventario. Muestre cuántos bienes de cada fruta quedan en inventario, su valor y el valor de la totalidad.

In [2]:

```
#Creamos el diccionario inventario
stock={}
#Añadimos valores
stock["uva"]= 7
stock["manzana"]= 0
stock["naranja"] =15
stock["piña"]= 12

#Creamos el diccionario: Precios
precios={}
#Add values
precios["uva"]=2
precios["manzana"]= 3
precios["naranja"]= 0.5
precios["piña"]= 5

#Muestra los precios y el stock

for bien in precios:
    print(bien + ": precio= %s dólares y cantidad= %d" % (precios[bien],stock[bien]))

total=0
for precio in precios:
    dinero= precios[precio]*stock[precio]

total= total + dinero

print("El total es de %s dólares" % total)
```

```
uva: precio= 2 dólares y cantidad= 7
manzana: precio= 3 dólares y cantidad= 0
naranja: precio= 0.5 dólares y cantidad= 15
piña: precio= 5 dólares y cantidad= 12
El total es de 60 dólares
```

Ejercicio 3

Escriba una función "duplicados()" que tome los elementos de una lista y elimine aquellos que están repetidosl. Por ejemplo: duplicados([4,3,3,3]) debiese devolver el valor [4,3].

```
In [5]:
```

```
def duplicados(num):
    result = []
    for item in num:
        if item not in result:
            result.append(item)
    return result

duplicados([4,5,5,5,5,4,3,2,1,4,5,6,7,8])
```

```
Out[5]:
```

```
[4, 5, 3, 2, 1, 6, 7, 8]
```

Ejercicios Propuestos

Ejercicio 1

Usando un loop, y en base al diccionario entregado, genere un programa que entregue un puntaje a las palabras escritas, como si de scrabble se tratara.

```
In [4]:
```

```
puntaje = {"a": 1, "c": 3, "b": 3, "e": 1, "d": 2, "g": 2,
           "f": 4, "i": 1, "h": 4, "k": 5, "j": 8, "m": 3,
           "l": 1, "o": 1, "n": 1, "q": 10, "p": 3, "s": 1,
           "r": 1, "u": 1, "t": 1, "w": 4, "v": 4, "y": 4,
           "x": 8, "z": 10}
```

Ejercicio 2:

Genere una tabla que muestre compare las críticas de películas de las 2 personas que se ingresen.

```
In [18]:
```

```
critica_peliculas = {}
critica_peliculas = {
    'Juan': {'Avenger: Age of Ultron': 7.0, 'Django Unchained': 7.5, 'Gone Girl': 6.5, 'Kill the Messenger': 8.0},
    'Diego': {'Avenger: Age of Ultron': 10.0, 'Django Unchained': 6.0, 'Gone Girl': 6.5, 'Kill the Messenger': 6.0, 'Zoolander': 6.5},
    'Carlos': {'Avenger: Age of Ultron': 7.0, 'Django Unchained': 7.0, 'Gone Girl': 10.0, 'Horrible Bosses 2': 6.0, 'Kill the Messenger': 5.0, 'Zoolander': 10.0},
    'Rodrigo': {'Avenger: Age of Ultron': 8.0, 'Django Unchained': 7.0, 'Horrible Bosses 2': 5.0, 'Kill the Messenger': 9.0, 'Zoolander': 9.0},
    'Vicente': {'Avenger: Age of Ultron': 10.0, 'Django Unchained': 7.5, 'Gone Girl': 6.0, 'Horrible Bosses 2': 3.0, 'Kill the Messenger': 5.5, 'Zoolander': 7.0},
    'Javier': {'Avenger: Age of Ultron': 8.5, 'Django Unchained': 9.0, 'Zoolander': 1.0},
    'Celso': {'Avenger: Age of Ultron': 6.0, 'Django Unchained': 8.0, 'Gone Girl': 7.0, 'Horrible Bosses 2': 4.0, 'Kill the Messenger': 6.5, 'Zoolander': 3.0}

critica_peliculas['Javier']['Zoolander']
```

```
Out[18]:
```

```
1.0
```

```
In [4]:
```

```
import pandas as pd #Nos permitirá tratar las tablas.

def comparacion(input_data, user1, user2):

    #----- Ingrese su código aquí-----

    return pd.DataFrame(data = data, columns = ['movie', user1, user2])
```

```
respuesta = comparacion(critica_peliculas, 'Juan', 'Diego')
respuesta
```

Ejercicio 3 (Kattis)

[A new Alphabet](#)

¿Dudas?