

listas-tuplas-set-diccionarios

Google Colab interface showing a code cell with a green border containing Python code. The code defines a list of names, performs various list operations like append, insert, remove, and pop, and prints the final state of the list. The output shows the list being modified step-by-step.

```
1 nombres = ["juan", "carla", "ricardo", "maria", "rodrigo"]
2 print(nombres)
3 print(type(nombres))
4 print(len(nombres))
5 print(nombres[1])
6 print(nombres[0:2])
7 print(nombres[1:3])
8 print(nombres[2:])
9 nombres[3] = "linux"
10 print(nombres)
11 for i in nombres:
12     print(i)
13 nombres.append("lorenzo")
14 print(nombres)
15 nombres.insert(1, "octavio")
16 print(nombres)
17 nombres.remove("octavio")
18 print(nombres)
19 nombres.append("ubuntu")
20 print(nombres)
21 nombres.pop()
22 print(nombres)
23 del nombres[1]
24 print(nombres)

[{"name": "juan", "carla", "ricardo", "maria", "rodrigo"}]
```

Code cell [27]:

```
1 print("numeros divisibles entre 3")
2 for i in range(11):
```

Google Colab interface showing a code cell with a green border containing Python code. The code prints numbers divisible by 3, generates a range from 2 to 7, and prints values from 3 to 10 with an increment of 2. The output shows the results of each print statement.

```
1 print("numeros divisibles entre 3")
2 for i in range(11):
3     if i % 3 == 0:
4         print(i)

numeros divisibles entre 3
0
3
6
9

1 print("rango de 2 a 6")
2 rango = range(2,7)
3 for i in rango :
4     print(i)

rango de 2 a 6
2
3
4
5
6

1 print("rango con valores de 3 a 10 e incremento de 2")
2 rango = range(3,11,2)
3 for i in rango:
4     print(i)

rango con valores de 3 a 10 e incremento de 2
3
5
7
9
```

The screenshot shows a Google Colab notebook titled "tuplas.ipynb". A specific code cell is highlighted with a red border. The code in the cell is:

```
1 print("esto es una tupla tiene las mismas funciones que las listas")
2 frutas =('naranja', 'platano', 'uva')
3 print(frutas)
```

The output of the code is displayed below the cell, also enclosed in a red box:

```
estos es una tupla tiene las mismas funciones que las listas
('naranja', 'platano', 'uva')
```

set.ipynb

```
1 print("esto es un set")
2 print("no soporta los elementos duplicados")
3 planetas={'marte','jupiter', 'venus'}
4 print(planetas)
5 print(len(planetas))
6 print('marte'in planetas)
7 print("mercurio"in planetas)
8 planetas.add("tierra")
9 print(planetas)
10 planetas.remove("tierra")
11 print(planetas)
```

0 s se ejecutó 15:36

Output:

```
estos es un set
no soporta los elementos duplicados
{'venus', 'marte', 'jupiter'}
3
True
False
{'venus', 'marte', 'jupiter', 'tierra'}
{'venus', 'marte', 'jupiter'}
```

diccionarios.ipynb

```
1 print("esto es un diccionario")
2 dic={"ide":"sos", "oop":"objet", "db":"base"}
3 print(dic)
4 print(len(dic))
5 print(dic['ide'])
6 print(dic.get('oop'))
7 dic['ide']="help"
8 print(dic)
9 for termino , valor in dic.items():
10    print(termino,valor)
11    for termino in dic.keys():
12        print(termino)
13    for valor in dic.values():
14        print(valor)
15 print('ide' in dic)
16 dic['pk']="primary key"
17 print(dic)
18 dic.pop('db')
19 print(dic)
```

0 s se ejecutó 15:48

Output:

```
estos es un diccionario
{'ide': 'sos', 'oop': 'objet', 'db': 'base'}
sos
objet
{'ide': 'help', 'oop': 'objet', 'db': 'base'}
ide help
oop objet
db base
ide
oop
db
help
objet
base
True
{'ide': 'help', 'oop': 'objet', 'db': 'base', 'pk': 'primary key'}
{'ide': 'help', 'oop': 'objet', 'pk': 'primary key'}
```