




Diccionarios Resolución mediante doble diccionario



Fundamentos de Programación
Departamento de Lenguajes y Sistemas Informáticos



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

Hemos visto en la lección anterior como construir diccionario en el que los *valores* son contadores, sumas, listas o un conjunto.

Si se dispone de una lista de tuplas de *estudiantes* (suponemos que las tuplas se crearon con el tipo):
`Estudiante=NamedTuple("estudiante", [("nombre",str),("edad",int),("curso",str)])`

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
( 'Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
( 'Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
( 'Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
( 'Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
( 'Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
( 'Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

Básicamente tenemos esquemas para dar respuesta a preguntas como:

- *¿Cuántos alumnos hay en cada curso?*
- *¿Cuánto suman las edades de los alumnos de un mismo curso?*
- *Devuelve una lista de los nombres de los alumnos con la misma edad*
- *Devuelve los distintos nombres de los alumnos con la misma edad*



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuántos alumnos hay en cada curso?

```
dic1=dict()  
for e in estudiantes:  
    if e.curso not in dic1:  
        dic1[e.curso]=0  
    dic1[e.curso]+=1  
return dic1
```

```
{'C4': 7, 'C2': 6, 'C1': 9, 'C3': 5}
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuánto suman las edades de los alumnos de un mismo curso?

```
dic2=dict()
```

```
for e in estudiantes:
```

```
    if e.curso not in dic2:
```

```
        dic2[e.curso]=0
```

```
        dic2[e.curso]+=e.edad
```

```
return dic2
```

```
{'C4': 132, 'C2': 119, 'C1': 172, 'C3': 99}
```




Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

Devuelve una lista de los nombres de los alumnos con la misma edad

```
dic3=dict()  
for e in estudiantes:  
    if e.edad not in dic3:  
        dic3[e.edad]=list()  
    dic3[e.edad].append(e.nombre)  
return dic3
```



```
{19:['Ismael','M.Mar','Daniel','Mario','Javier','Luis'],18:['Ruben','Rocío','David',  
, 'Javier','Adrián','Rocío','Pedro','Manuel'],20:['Lorena','Mario','Daniel','Carlos',  
, 'Guillermo','José','Fernando','Ana'],17:['Javier'],21:['Javier', 'Javier'],22:  
['Celia'],23:['David']}
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

Devuelve los distintos nombres de los alumnos con la misma edad

```
dic4=dict()  
for e in estudiantes:  
    if e.edad not in dic4:  
        dic4[e.edad]=set()  
    dic4[e.edad].add(e.nombre)  
return dic4
```

{19: {'Daniel', 'Mario', 'Luis', 'Ismael', 'M.Mar', 'Javier'}, 18: {'Ruben', 'Rocío', 'David', 'Javier', 'Pedro', 'Adrián', 'Manuel'}, 20: {'Daniel', 'Mario', 'Carlos', 'Guillermo', 'José', 'Lorena', 'Ana', 'Fernando'}, 17: {'Javier'}, 21: {'Javier'}, 22: {'Celia'}, 23: {'David'}}



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

Y si las preguntas hubiesen sido:

- *¿Cuál es el estudiante de más edad de cada curso?*
- *Devuelve una lista alfabética de los nombres de los alumnos de la misma edad*
- *¿Cuáles son los promedios de las edades de cada curso?*
- *¿Cuáles son los nombres de los 3 alumnos de más edad de cada curso?*

Ahora se trata de aprender esquemas que permitan proporcionar respuesta a dichas preguntas:

*El esquema en general se basa en construir un primer diccionario que agrupe los datos en **listas o conjuntos** —si no debe haber repetidos— y obtener un segundo diccionario a partir del primero en el que se aplique una **operación adicional** a los **valores** (máximo, ordenación, promedio, ...).*

*El segundo diccionario se obtiene recorriendo el primer diccionario (hay que usar el método “**items()**”).*

- Las **claves** del segundo diccionario serán las del primero (*no hay que comprobar si ya están o no en este segundo diccionario*)
- Los **valores** del segundo diccionario se obtienen realizando la **operación adicional** a los valores del primero.



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuál es el estudiante de más edad de cada curso?

```
aux=dict()  
for e in estudiantes:  
    if e.curso not in aux:  
        aux[e.curso]=list()  
        aux[e.curso].append(e)
```

```
{'C4': [('Ismael',19,'C4'),('Daniel',19,'C4'),('Adrián',18,'C4'),...],  
'C2': [('Ruben',18,'C2'),('Lorena',20,'C2'),('Daniel',20,'C2'),...],  
'C1': [('Rocío',18,'C1'),('M.Mar',19,'C1'),('Javier',17,'C1'),...],  
'C3': [('David',18,'C3'),('Mario',20,'C3'),('David',23,'C3'),...]}
```

```
res=dict()  
for c,v in aux.items():  
    res[c]=max(v,key=lambda e:e.edad).nombre  
return res
```

```
{'C4': 'Javier','C2': 'Javier','C1': 'Celia','C3': 'David'}
```




Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

Devuelve una lista alfabética de los nombres de los alumnos de la misma edad

```
aux=dict()  
for e in estudiantes:  
    if e.edad not in aux:  
        aux[e.edad]=list()  
    aux[e.edad].append(e.nombre)
```

```
{19:['Ismael','M.Mar','Daniel','Mario','Javier','Luis'],18:  
['Ruben','Rocío','David','Javier','Adrián','Rocío','Pedro',  
'Manuel'],20:['Lorena','Mario','Daniel','Carlos','Guillermo',  
'José','Fernando','Ana'],17:['Javier'],21:['Javier','Javi  
er'],22:['Celia'],23:['David']}
```

```
res=dict()  
for c,v in aux.items():  
    res[c]=sorted(v)  
return res
```

```
{19:['Daniel','Ismael','Javier','Luis','M.Mar','Mario'],18:['Adrián',  
'David','Javier','Manuel','Pedro','Rocío','Rocío','Ruben'],20:['Ana',  
'Carlos','Daniel','Fernando','Guillermo','José','Lorena','Mario'],17:  
['Javier'],21:['Javier','Javier'],22:['Celia'],23:['David']}
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuáles son los promedios de las edades de cada curso?

```
aux=dict()  
for e in estudiantes:  
    if e.curso not in aux:  
        aux[e.curso]=list()  
    aux[e.curso].append(e.edad)
```

```
{'C4': [19, 19, 18, 19, 18, 21, 18], 'C2': [18,  
20, 20, 21, 20, 20], 'C1': [18, 19, 17, 18, 22,  
19, 19, 20, 20], 'C3': [18, 20, 23, 20, 18]}
```

```
res=dict()  
for c,v in aux.items():  
    res[c]=sum(v)/len(v)  
return res
```

```
{'C4':18.857142857142858,'C2':19.833333333333332,  
'C1': 19.111111111111111, 'C3': 19.8}
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuáles son los nombres de los 3 alumnos de más edad de cada curso? Estrategia 1

```
aux=dict()  
for e in estudiantes:  
    if e.curso not in aux:  
        aux[e.curso]=list()  
    aux[e.curso].append(e)  
  
res=dict()  
for c,v in aux.items():  
    res[c]=list()  
    for e in sorted(v,key=lambda e:e.edad, reverse=True)[:3]:  
        res[c].append(e.nombre)  
return res
```

```
{'C4': [('Ismael',19,'C4'),('Daniel',19,'C4'),('Adrián',18,'C4'),...],  
'C2': [('Ruben',18,'C2'),('Lorena',20,'C2'),('Daniel',20,'C2'),...],  
'C1': [('Rocío',18,'C1'),('M.Mar',19,'C1'),('Javier',17,'C1'),...],  
'C3': [('David',18,'C3'),('Mario',20,'C3'),('David',23,'C3'),...]}
```

```
{'C4': ['Javier','Ismael','Daniel'],'C2': ['Javier','Lorena','Daniel'],  
'C1': ['Celia','Fernando','Ana'],'C3': ['David','Mario','José']}
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuáles son los nombres de los 3 alumnos de más edad de cada curso? Estrategia 2

```
aux=dict()  
for e in estudiantes:  
    if e.curso not in aux:  
        aux[e.curso]=list()  
    aux[e.curso].append((e.nombre,e.edad))
```

```
{'C4': [('Ismael',19), ('Daniel',19), ('Adrián',18),...],  
 'C2': [('Ruben',18), ('Lorena',20), ('Daniel',20),...],  
 'C1': [('Rocío',18), ('M.Mar',19), ('Javier',17),...],  
 'C3': [('David',18), ('Mario',20), ('David',23),...]}
```

```
res=dict()  
for c,v in aux.items():  
    res[c]=list()  
    for e in sorted(v,key=lambda e:e[1], reverse=True)[:3]:  
        res[c].append(e[0])  
return res
```

```
{'C4': ['Javier', 'Ismael', 'Daniel'], 'C2': ['Javier', 'Lorena', 'Daniel'],  
 'C1': ['Celia', 'Fernando', 'Ana'], 'C3': ['David', 'Mario', 'José']}
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

Existen otro tipo de preguntas en que se busca “*el que más... o el que menos...*”, “*el mayor de los... o el menor de los...*”, “*los n-primeros que... o los n-últimos que...*”, “*los n-mayores que... o los n-menores que...*”. Estos ejercicios son muy sencillos de resolver usando las funciones **max**, **min** o **sorted**, realmente la dificultad radica en ver si se dispone de un contenedor con dato para aplicar directamente alguna de estas funciones.

Supongamos las siguientes preguntas:

- ¿*Cuáles son los datos de los 5 alumnos de más edad?*
- ¿*Cuál es el curso con menos alumnos?*
- ¿*Cuál es el nombre que más se repite y cuánto suman las edades del nombre de que se trate?*
- ¿*Qué curso es el de alumnos “más viejos”? Es decir, el que tiene mayor promedio de edad*



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuáles son los datos de los 5 alumnos de más edad?

Vemos que en la lista original de *estudiantes* se dispone de la edad de cada persona, por lo que resulta sencillo una de las dos siguientes opciones .

```
res=sorted(estudiantes, key=lambda e:e.edad, reverse=True)[:5] ★  
return res
```

```
[('David',23,'C3'),('Celia',22,'C1'),('Javier',21,'C2'),  
('Javier',21,'C4'), ('Lorena',20,'C2')]
```

O también

```
res=sorted(estudiantes, key=lambda e:e.edad)[-5:]  
return res
```

```
[('Ana',20,'C1'),('Javier',21,'C2'),('Javier',21,'C4'),  
('Celia',22,'C1'),('David',23,'C3')]
```



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuál es el curso con menos alumnos?

*En este caso vemos que es necesario, antes de aplicar la función **min**, tener contados los alumnos de cada curso, por ello lo suyo es hacer un diccionario que a cada curso el asocie el número de alumnos.*

```
dic1=dict()  
for e in estudiantes:  
    if e.curso not in dic1:  
        dic1[e.curso]=0  
    dic1[e.curso]+=1  
mínimo=min(dic1.items(), key=lambda e:e[1])  
return mínimo[0]
```

Diagram illustrating the dictionary construction and the result:

- The dictionary `dic1` is built by iterating over `estudiantes`. The final state of `dic1` is shown as: `{'C4': 7, 'C2': 6, 'C1': 9, 'C3': 5}`.
- The `min` function is applied to `dic1.items()` with the key function `lambda e:e[1]`. The result is `('C3', 5)`.
- The final result is `C3`.



Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Cuál es el nombre que más se repite y cuánto suman las edades del nombre de que se trate?

Antes de aplicar la función `max`, es necesario agrupar en un primer diccionario las edades de cada nombre para poder contar y sumar dichas edades. Con un segundo diccionario se cuentan y suman esas listas de edades.

```
aux=dict()  
for e in estudiantes:  
    if e.nombre not in aux:  
        aux[e.nombre]=list()  
    aux[e.nombre].append(e.edad)
```

```
{'Ismael':[19], 'Ruben':[18], 'Lorena':[20], 'Rocío':[18,18],  
'M.Mar':[19], 'David':[18,23], 'Mario':[20,19], 'Daniel':[20,  
19], 'Javier':[17,18,21,19,21], 'Adrián':[18], 'Celia':[22], '  
Carlos':[20], 'Guillermo':[20], 'José':[20], 'Luis':[19], 'Fer  
nando':[20], 'Pedro':[18], 'Ana':[20], 'Manuel':[18]}
```

```
res=dict()  
for c,v in aux.items():  
    res[c]=(len(v),sum(v))
```

```
{'Ismael':(1,19), 'Ruben':(1,18), 'Lorena':(1,20), 'Rocío':(2  
,36), 'M.Mar':(1,19), 'David':(2, 41), 'Mario':(2, 39), ... }
```

```
máximo=max(res.items(),key=lambda e:e[1][0]) → ('Javier',(5, 96))
```

```
return (máximo[0],máximo[1][0],máximo[1][1]) → ('Javier',5, 96)
```




Diccionarios (Esquema para resolución de problemas mediante el uso de dos diccionarios)

```
estudiantes=[('Ismael',19,'C4'),('Ruben',18,'C2'),('Lorena',20,'C2'),  
('Rocío',18,'C1'),('M.Mar',19,'C1'),('David',18,'C3'),('Mario',20,'C3'),  
('Daniel',20,'C2'),('Javier',17,'C1'),('Daniel',19,'C4'),('Javier',18,'C1'),  
('Adrián',18,'C4'),('Javier',21,'C2'),('Celia',22,'C1'),('David',23,'C3'),  
('Mario',19,'C4'),('Rocío',18,'C4'),('Javier',19,'C1'),('Carlos',20,'C2'),  
('Guillermo',20,'C2'),('José',20,'C3'),('Luis',19,'C1'),('Javier',21,'C4'),  
('Fernando',20,'C1'),('Pedro',18,'C3'),('Ana',20,'C1'),('Manuel',18,'C4')]
```

¿Qué curso es el de alumnos “más viejos”? Es decir, el que tiene mayor promedio de edad

Antes de aplicar la función **max**, es necesario agrupar en un primer diccionario las edades de cada curso para poder calcular el promedio. Con un segundo diccionario se promedia las listas de edades.

```
aux=dict()  
for e in estudiantes:  
    if e.curso not in aux:  
        aux[e.curso]=list()  
    aux[e.curso].append(e.edad)
```

```
{'C4': [19, 19, 18, 19, 18, 21, 18], 'C2': [18,  
20, 20, 21, 20, 20], 'C1': [18, 19, 17, 18, 22,  
19, 19, 20, 20], 'C3': [18, 20, 23, 20, 18]}
```

```
res=dict()  
for c,v in aux.items():  
    res[c]=sum(v)/len(v)
```

```
{'C4':18.857142857142858,'C2':19.833333333333332,  
'C1': 19.111111111111111, 'C3': 19.8}
```

```
máximo=max(res.items(),key=lambda e:e[1])
```

```
('C2', 19.833333333333332)
```

```
return máximo[0]
```

C2



Ejercicio:

Proyecto L09_ITV:

- Descargue el fichero de enunciados “*README_v3.md*”
- *Cópielo en el proyecto.*
- *Realice los ejercicios 10, 11 y 12*