

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos. Mantiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.

Guia Python

•Named Tuple

```
Nombre=namedtuple('Nombre', 'Variables (separadas por ,)')
```

•Ler

```
lista=[]
    with open(fichero, 'r', encoding='utf-8') as f:
        next(f)
        lector=csv.reader(f)
        for Variables in lector:
            lista.append(Nombre namedtuple(Variables))
    return lista
```

▪Teniendo que usar castear para convertir

```
registros.append(Registro(name,province,city,address,int(postalCode),
                           float(latitude),float(longitude)))
```

▪Teniendo que usar funciones para convertir(Es necesario colocarlo por separado antes) (Importante esa _ para obviar una varibale)

```
for _,pu,lo,fe,de,ba,pr in lector:
    pu=pu.strip()
    lo=lo.strip()
    fe=datetime.strptime(fe, '%B %d, %Y')
    de=texto_a_palabras(de)
    ba=texto_a_palabras(ba)
    pr=texto_a_palabras(pr)
    registros.append(Registro(pu,lo,fe,de,ba,pr))
return registros
```

•Contenedores

►Lista: Usa[]

```
[r.bicis for r in estaciones]
```

▪Lista de tuplas: Cuando las listas llevan mas de una variable [(r.bicis, r.nombre) for r in estaciones]

►Conjunto:

►Diccionario: Usa {}

```
{r.nombre:r.bicis for r in estaciones}
```

•Graficas

•Operaciones

--Ordenar (menor a mayor)

```
bic_nom=sorted([(r.bicis,r.nombre) for r in estaciones])
```

--Ordenar (mayor a menor)

```
bic_nom=sorted([(r.bicis,r.nombre) for r in estaciones],reverse=True)
```

--Ordenar (funcion Lambda: Permite ordenar respecto de cualquier elemento del contenedor, no solo el primero)

```
return max(tuplas, key=lambda t:t[2])
```

--Limitar elementos

```
x=[tupla[1] for tupla in bic_nom][:10]
```

Consulta condiciones aquí



WUOLAH

--Suma de elementos

```
sum(bicis_disp)
```

--Unir 2 contenedores

```
puntos=list(zip(x,y))
```

En este caso, será una lista(list), con tuplas de pares de elementos de los contenedores x e y

--Sacar pares clave/valor de un diccionario (Lista de tuplas)

```
di.items()
```

--Formato de la fecha al leer

```
datetime.strptime(Fecha, '%d/%m/%Y').date()
```

--Contar las veces que aparece un elemento

```
basicas=[r.basicas for r in registros if r.fecha.year==anyo][:100]
palabras=[]
for descrip in basicas:
    for pal in descrip:
        palabras.append(pal)
tuplas=[(palabras.count(p),p) for p in palabras]
tupla_max=max(tuplas)
return tupla_max[1]
```

--Defaultdicc (Permite realizar conteos sin preocuparnos por si la clave existe)

```
dicc = defaultdict(int) #--El tipo del que seran los valores
for report in reports:
    if report.unit == 'cases':
        dicc[report.date.year] = dicc[report.date.year]+ report.value
return dicc
```

--Counter (Permite contar cuantas veces aparece la clave, lo que facilita buscar maximos)

```
lis_distribuidoras=[juego.publisher for juego in juegos]
contPorDistribuidora = Counter(lis_distribuidoras)
return max(contPorDistribuidora, key=contPorDistribuidora.values())
```

●Notas

--Para pillar la menor/mayor fecha(datetime), mejor usar min/max

--Añadir valor a una clave

```
Dicc[Clave] = Valor
```

Se pueden llamar a funciones usando r.varibale, aunque es bastante lento, es mejor hacer primero algun set de dicha variable

```
#Forma rapida
nombres=set(r.name for r in registros)
lista_presencia=[(porc_rest_nombre(registros, n),n for n in nombres)]
return max(lista_presencia)

#Forma lenta
lista_presencia=[(porc_rest_nombre(registros, r.name),r.name)
                 for r in registros]
return max(lista_presencia)
```