SABES GESTIONAR I PROCESSAR LES DADES: SOLUCIÓ

Abans de resoldre els punts que se'ns presenten, necessitem importar la llibreria *pandas* i guardar la base de dades en una variable.

```
import pandas as pd

data_base = pd.read_csv('insurance.csv')
```

Una vegada fet aquest pas, ja podem començar a resoldre les qüestions. La primera ens demana aplicar la funció *describe()*, sumar els *bmi* i calcular la mitjana dels *charges* per a l'interval de files des de la 45 fins a la 68. Bé, comencem per crear una nova base de dades que estigui formada només per aquest interval de files:

```
data_base_2 = data_base.iloc[45:69,]
```

Nota: recorda que el 69 no s'inclou ja que estem indexant una llista

Apliquem, ara, la funció *describe()* i obtenim aquesta taula:

	age	bmi	children	charges
count	24.000000	24.000000	24.000000	24.000000
mean	38.375000	31.503958	1.250000	15380.546530
std	15.747498	5.280530	1.151558	13523.381241
min	18.000000	22.420000	0.000000	1743.214000
25%	24.000000	27.098750	0.000000	4411.907213
50%	38.000000	32.727500	1.000000	10618.057050
75%	53.500000	35.766250	2.000000	23325.660650
max	64.000000	39.100000	4.000000	47496.494450

A continuació, seleccionem la columna *bmi* i calculem la suma dels seus valors:

```
data_base_2['bmi'].sum()
```

El resultat que obtenim és 756,095.

Tot seguit, calculem la mitjana de *charges* mitjançant la funció *mean().* Ho fem d'aquesta manera:

data_base_2['charges'].mean()

La funció ens torna el valor de 15380,54, que és el mateix que ens apareix a la taula de *describe().*

Seguim amb el mateix interval de files per al segon punt. Ara, ens demanen que mostrem els 12 primers registres de manera ordenada en ordre descendent, segons els valors de *bmi*. Bé, necessitem aplicar la funció *sort_values()* i la funció *head().* Ho fem així:

data_base_2.sort_values(['bmi'], ascending=False).head(12)

La taula apareix d'aquesta manera:

	age	sex	bmi	children	smoker	region	charges
66	61	female	39.100	2	no	southwest	14235.07200
46	18	female	38.665	2	no	northeast	3393.35635
59	34	female	37.335	2	no	northwest	5989.52365
45	55	male	37.300	0	no	southwest	20630.28351
55	58	male	36.955	2	yes	northwest	47496.49445
68	40	female	36.190	0	no	southeast	5920.10410
50	18	female	35.625	0	no	northeast	2211.13075
49	36	male	35.200	1	yes	southeast	38709.17600
47	28	female	34.770	0	no	northwest	3556.92230
53	36	male	34.430	0	yes	southeast	37742.57570
61	25	male	33.660	4	no	southeast	4504.66240
51	21	female	33.630	2	no	northwest	3579.82870

Ara, ens demanen afegir un segon nivell amb la variable *charges*, deixant per a aquest nivell l'ordre predeterminat. També ens demanen canviar el primer nivell per un ordre ascendent i ho farem així:

data_base_2.sort_values(['bmi', 'charges']).head(12)

Si ens hi fixem, ara la funció *sort_values* no necessita especificar valor en el paràmetre *ascending*, ja que, per al segon nivell, ens demanen deixar el valor predeterminat i, per al

primer, ens demanen canviar-lo en sentit ascendent i és aquest el que la funció té per predeterminat.

En el tercer punt, hem de tornar a fer servir la base de dades principal *data_base*. Per tal de saber quantes persones fumadores hi ha a la zona *northwest*, caldrà agrupar la base de dades en dos nivells: primer, segons *region* i, després, segons *smoker*. El codi que necessitem és:

data_base.groupby(['region','smoker']).size()

La taula que obtenim ens ensenyarà el que estem buscant:

```
region     smoker
northeast  no        257
           yes        67
northwest  no        267
           yes        58
southeast  no        273
           yes        91
southwest  no        267
           yes        58
dtype: int64
```

El nombre de persones fumadores per a la regió *northwest* és de 58.

Per saber la mitjana del nombre de fills i filles que hi ha a cada regió, necessitem una altra vegada agrupar les dades entorn a la variable *region* i calcular la mitjana en funció de la variable *children*. El codi és aquest:

data_base.groupby('region')['children'].mean()

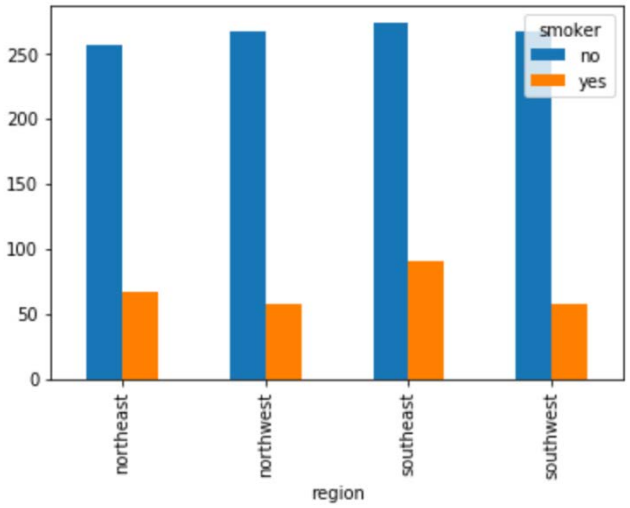
```
region
northeast    1.046296
northwest    1.147692
southeast    1.049451
southwest    1.141538
Name: children, dtype: float64
```

Per generar un gràfic de barres que visualitzi la taula de dalt, caldrà fer ús de la funció *plot.bar()* d'aquesta manera:

```
group1 = data_base.groupby(['region','smoker']).size()

group1.unstack(fill_value=0).plot.bar()
```

El gràfic que obtenim és aquest:



Ara, necessitarem crear una altra base de dades a partir de la principal *data_base* amb files que només tinguin valors de *bmi* superiors a 35 i que les persones no siguin fumadores. El codi que necessitem és:

```
data_base_3 = data_base[(data_base['bmi'] > 35) & (data_base['smoker'] == 'no')]

data_base_3.head(12)
```

La taula que resulta, després d'executar el codi, és aquesta:

	age	sex	bmi	children	smoker	region	charges
13	56	female	39.820	0	no	southeast	11090.71780
18	56	male	40.300	0	no	southwest	10602.38500
20	60	female	36.005	0	no	northeast	13228.84695
41	31	female	36.630	2	no	southeast	4949.75870
44	38	male	37.050	1	no	northeast	6079.67150
45	55	male	37.300	0	no	southwest	20630.28351
46	18	female	38.665	2	no	northeast	3393.35635
50	18	female	35.625	0	no	northeast	2211.13075
59	34	female	37.335	2	no	northwest	5989.52365
66	61	female	39.100	2	no	southwest	14235.07200
68	40	female	36.190	0	no	southeast	5920.10410
77	21	male	35.530	0	no	southeast	1532.46970

Fixem-nos que aquí hem fet ús de la indexació a través de la introducció d'una sèrie de condicions: la primera és dir que els valors de la variable *bmi* han de ser superiors a 35 i que els valors de la variable *smoker* només poden tenir el valor de no. Totes aquelles observacions que no compleixin estrictament les dues condicions no es carregaran a la nova taula que hem anomenat *data_base_3*. Les files han de complir les dues condicions. En el cas que només en compleixin una no seran seleccionades. Per tal que sí siguin seleccionades només complint una de les dues condicions, hem de canviar el codi d'aquesta manera:

```
data_base_3 = data_base[(data_base['bmi'] > 35) | (data_base['smoker'] == 'no')]

data_base_3.head(12)
```

El que hem fet és senzillament canviar el valor de l'operador condicional. En el primer cas, teníem l'operador *&*, que implica que les dues condicions s'han de complir. En el segon cas, en canvi, tenim l'operador /, que implica que només complint una de les dues condicions, la fila serà seleccionada.

Si hem d'afegir l'edat superior a 25 com a condició obligatòria, mantenint que, només cal que el valor *bmi* sigui superior a 35 o que la persona no sigui fumadora, haurem de modificar el codi i escriure'l així:

```
data_base_3 = data_base[((data_base['bmi'] > 35) | (data_base['smoker'] == 'no')) & (data_base['age'] > 25)]

data_base_3.head(12)
```

Ara, estem fent ús dels operadors & i | en el mateix codi. Amb els parèntesis, el que fem és agrupar les condicions. De fet, hi ha dues condicions que s'ha de complir. La primera és:

```
(data_base['bmi'] > 35) | (data_base['smoker'] == 'no')
```

I la segona és:

```
data_base['age'] > 25)]
```

Totes dues s'han de complir, però, per tal que es compleixi la primera, només cal que una de les dues opcions que conté sigui certa.

Descobreix tot el que Barcelona Activa pot fer per a tu

Acompanyament durant tot el procés de recerca de feina

barcelonactiva.cat/treball

Suport per posar en marxa la teva idea de negoci

barcelonactiva.cat/emprenedoria

Serveis a les empreses i iniciatives socioempresarials

barcelonactiva.cat/empreses

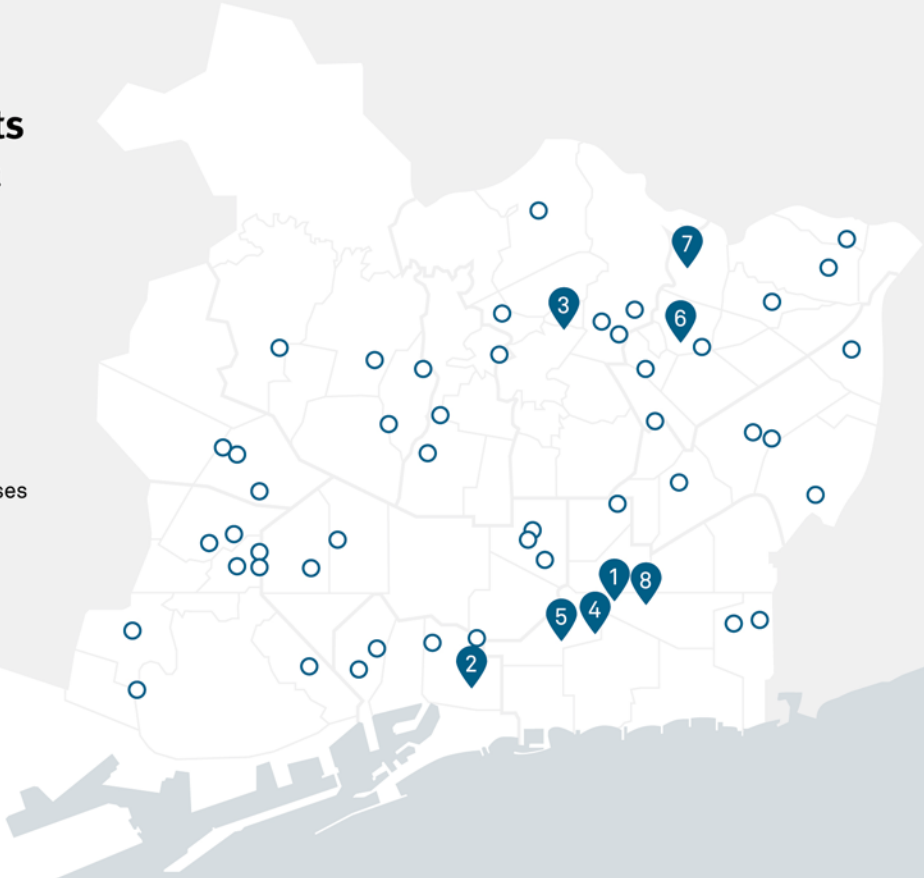
Formació tecnològica i gratuïta per a la ciutadania

barcelonactiva.cat/cibernarium

Xarxa d'equipaments de Barcelona Activa

1. Seu Central Barcelona Activa
   Porta 22
   Centre per a la Iniciativa Emprenedora Glòries
   Incubadora Glòries
2. Convent de Sant Agustí
3. Ca n'Andalet
4. Oficina d'Atenció a les Empreses
   Cibernàrium
   Incubadora MediaTIC
5. Incubadora Almogàvers
6. Parc Tecnològic
7. Nou Barris Activa
8. innoBA

○ Punts d'atenció a la ciutat



Barcelona Activa

Ajuntament de Barcelona

Darrera actualització 2019

Cofinançat per:

UNIÓ EUROPEA
Fons Europeu de Desenvolupament Regional

Segueix-nos a les xarxes socials:

barcelonactiva.cat/cibernarium
barcelonactiva
barcelonactiva
company/barcelona-activa