

MySQL Workbench

aldo - Warning - not supported

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

autos

Tables

autos_filtrados

vehiculos

Views

Stored Procedures

Functions

europa_soccer

hospital_general_ra_idgs_220219

hospital_general_ra_idgs_220700

imagenes_db

login_voe

loginapp

nba

nba2

nba_fixed

production

proyecto-deseguridad

sys

test

todo_list

Administration Schemas

Information

No object selected

Object Info Session

Ukvia intensa

Hoy

SQL File 3 autos_ filtrados

Limit to 1000 rows

1 SELECT * FROM autos.autos_filtrados;

Result Grid

	id	marca	modelo	año	precio	disponible
1	1	Toyota	Corolla	2018	220000	SI
2	2	Honda	Civic	2020	250000	SI
3	3	Ford	Fiesta	2015	140000	No
4	4	Nissan	Versa	2019	200000	SI
5	5	Volkswagen	Jetta	2021	280000	SI
6	6	Chevrolet	Aveo	2016	150000	No
7	7	Kia	Rio	2022	300000	SI
8	8	Toyota	Corolla	2018	220000	SI
9	9	Toyota	Corolla	2018	220000	SI
10	10	Honda	Civic	2020	250000	SI
11	11	Ford	Fiesta	2015	140000	No
12	12	Nissan	Versa	2019	200000	SI
13	13	Volkswagen	Jetta	2021	280000	SI
14	14	Chevrolet	Aveo	2016	150000	No
15	15	Kia	Rio	2022	300000	SI
16	16	Mazda	3	2023	320000	SI

Output

Action Output

1 08:45:53 SELECT * FROM autos.autos_filtrados LIMIT 0-1000

Duration / Fetch 0.016 sec / 0.000 rows

08:45 a. m. 25/06/2025

MySQL Workbench

aldo - Warning - not supported

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

autos

Tables

autos_filtrados

vehiculos

Views

Stored Procedures

Functions

europa_soccer

hospital_general_ra_idgs_220219

hospital_general_ra_idgs_220700

imagenes_db

login_voe

loginapp

nba

nba2

nba_fixed

production

proyecto-deseguridad

sys

test

todo_list

Administration Schemas

Information

No object selected

Object Info Session

Ukvia intensa

Hoy

SQL File 3 autos_ filtrados vehiculos

Limit to 1000 rows

1 SELECT * FROM autos.vehiculos;

Result Grid

	id	marca	modelo	año	precio	disponible
1	1	Toyota	Corolla	2018	220000.00	1
2	2	Honda	Civic	2020	250000.00	1
3	3	Ford	Fiesta	2015	140000.00	0
4	4	Nissan	Versa	2019	200000.00	1
5	5	Volkswagen	Jetta	2021	280000.00	1
6	6	Chevrolet	Aveo	2016	150000.00	0
7	7	Kia	Rio	2022	300000.00	1
8	8	Toyota	Corolla	2018	220000.00	1
9	9	Toyota	Corolla	2018	220000.00	1
10	10	Honda	Civic	2020	250000.00	1
11	11	Ford	Fiesta	2015	140000.00	0
12	12	Nissan	Versa	2019	200000.00	1
13	13	Volkswagen	Jetta	2021	280000.00	1
14	14	Chevrolet	Aveo	2016	150000.00	0
15	15	Kia	Rio	2022	300000.00	1
16	16	Mazda	3	2023	320000.00	1

Output

Action Output

1 08:45:53 SELECT * FROM autos.vehiculos LIMIT 0-1000

Duration / Fetch 0.016 sec / 0.000 rows

08:46 a. m. 25/06/2025

Archivo Editar Selección Ver Ir Ejecutar ...

Practicad

EXPLORADOR

my_practique6.ipynb

Practica 6: Preparación de Datos para su Análisis (ETL)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema ...

Seleccionar el kernel

```
import pandas as pd
import mysql.connector
```

```
# librerías necesarias
from sqlalchemy import create_engine
```

```
# url de conexión
url = "mysql+pymysql://root:aldr@localhost:3306/autos"
engine = create_engine(url)
```

```
df = pd.read_sql("SELECT * FROM vehiculos", engine)
```

```
df.head()
```

	id	marca	modelo	año	precio	disponible
0	1	Toyota	Corolla	2018	220000.0	1
1	2	Honda	Civic	2020	250000.0	1
2	3	Ford	Fiesta	2015	140000.0	0
3	4	Nissan	Versa	2019	200000.0	1

Celda 1 de 79 | Go Live | Qodo Gen | Ninja | Colorize: 0 variables | Colorize | helper

Archivo Editar Selección Ver Ir Ejecutar ...

Practicad

EXPLORADOR

my_practique6.ipynb

Practica 6: Preparación de Datos para su Análisis (ETL)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema ...

Seleccionar el kernel

```
df.info()
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22 entries, 0 to 21
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  -
 0   id          22 non-null    int64
 1   marca       22 non-null    object
 2   modelo      21 non-null    object
 3   año         22 non-null    int64
 4   precio      21 non-null    float64
 5   disponible  22 non-null    int64
dtypes: float64(1), int64(3), object(2)
memory usage: 1.2+ KB
```

	id	año	precio	disponible
count	22.000000	22.000000	21.000000	22.000000
mean	11.500000	2018.954545	220476.190476	0.727273
std	6.493587	2.439307	55450.535658	0.455842
min	1.000000	2015.000000	140000.000000	0.000000
25%	6.250000	2017.250000	170000.000000	0.250000
50%	11.500000	2019.000000	220000.000000	1.000000
75%	16.750000	2021.000000	270000.000000	1.000000
max	22.000000	2023.000000	300000.000000	1.000000

Limpiar Datos

Celda 1 de 79 | Go Live | Qodo Gen | Ninja | Colorize: 0 variables | Colorize | helper

1 mm de lluvia
Hoy

Búsqueda

08:47 a. m.
25/06/2025

Archivo Editar Selección Ver Ir Ejecutar ...

Practicad

EXPLORADOR

my_practique6.ipynb U

Practica 6: Preparación de Datos para su Análisis (ETL)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema ...

Seleccionar el kernel

Limpiar Datos

```
df_nonulos = df.dropna(subset=['marca'])
```

```
#muestro metadatos despues del filtro
df_nonulos.info()
df_nonulos.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22 entries, 0 to 21
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype  
---  --
 0   id          22 non-null    int64  
 1   marca       22 non-null    object  
 2   modelo     21 non-null    object  
 3   año        22 non-null    int64  
 4   precio     21 non-null    float64 
 5   disponible 22 non-null    int64  
dtypes: float64(1), int64(3), object(2)
memory usage: 1.2+ KB
```

	id	año	precio	disponible
count	22.000000	22.000000	21.000000	22.000000
mean	11.500000	2018.954545	220476.190476	0.727273
std	6.493587	2.439307	55450.535658	0.455842
min	1.000000	2015.000000	140000.000000	0.000000
25%	6.250000	2017.250000	170000.000000	0.250000

Celda 1 de 79 | Go Live | Qodo Gen | Ninja | Colorize: 0 variables | Colorize | helper

1 mm de lluvia
Hoy

Archivo Editar Selección Ver Ir Ejecutar ...

Practicad

EXPLORADOR

my_practique6.ipynb U

Practica 6: Preparación de Datos para su Análisis (ETL)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema ...

Seleccionar el kernel

Transformacion de datos

```
# Muestro los primeros registros del dataframe
df_nonulos['disponible'].head()
```

```
df_nonulos.loc[:, 'disponible'] = df_nonulos['disponible'].replace({0: 'No', 1: 'Si'})
```

```
df_nonulos['disponible'].head()
```

```
0   Sí
1   Sí
2   No
3   Sí
4   Sí
Name: disponible, dtype: object
```

Como exportarlo a una base de datos temporal de SQL (MySQL)

Celda 1 de 79 | Go Live | Qodo Gen | Ninja | Colorize: 0 variables | Colorize | helper

1 mm de lluvia
Hoy

Archivo Editar Selección Ver Ir Ejecutar ... Q Practic6

EXPLORADOR

- PRÁCTICAS
 - .qodo
 - gitignore
 - autos_autos_filtrados.sql
 - autos_vehiculos.sql
 - my.practic6.ipynb
 - readme.md
 - salaries.csv

my.practic6.ipynb > Practica 6: Preparación de Datos para su Análisis (E11)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema ...

Importar datos de una API - get

```
# librerías necesarias
import pandas as pd
import requests

# obtenemos los datos
url = "https://api.covidtracking.com/v1/us/daily.json"
response = requests.get(url)
comentarios = response.json()

df_comentarios = pd.DataFrame(comentarios)

df_comentarios.head()
```

	date	states	positive	negative	pending	hospitalizedCurrently	hospitalizedCumulative	inCuCurrently	inCuCumulative	onVentilatorCurrently	...	lastM
0	20210307	56	28756489.0	74582825.0	11808.0	40199.0	776361.0	8134.0	45475.0	2802.0	...	0712
1	20210306	56	28714654.0	74450990.0	11783.0	41401.0	775635.0	8409.0	45453.0	2811.0	...	0612
2	20210305	56	28654639.0	74307155.0	12213.0	42541.0	775132.0	8634.0	45373.0	2889.0	...	0512
3	20210304	56	28585852.0	74035238.0	12405.0	44172.0	772351.0	8970.0	45293.0	2973.0	...	0412

main Bootstrap v5.3.6 0 4 Git Graph Celda 1 de 79 Go Live Qodo Gen Ninja Colorize: 0 variables Colorize helper

1 mm de Buía Hoy 08:48 a. m. 25/06/2025

Archivo Editar Selección Ver Ir Ejecutar ... Q Practic6

EXPLORADOR

- PRÁCTICAS
 - .qodo
 - gitignore
 - autos_autos_filtrados.sql
 - autos_vehiculos.sql
 - my.practic6.ipynb
 - readme.md
 - salaries.csv

my.practic6.ipynb > Practica 6: Preparación de Datos para su Análisis (E11)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema ...

df_comentarios.info()
df_comentarios.describe()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 420 entries, 0 to 419
Data columns (total 25 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   date                  420 non-null   int64
 1   states                420 non-null   int64
 2   positive              419 non-null   float64
 3   negative              372 non-null   float64
 4   pending               369 non-null   float64
 5   hospitalizedCurrently  356 non-null   float64
 6   hospitalizedCumulative 369 non-null   float64
 7   inCuCurrently         347 non-null   float64
 8   inCuCumulative        348 non-null   float64
 9   onVentilatorCurrently  348 non-null   float64
10  onVentilatorCumulative 341 non-null   float64
11  dateChecked           420 non-null   object
12  death                 392 non-null   float64
13  hospitalized           369 non-null   float64
14  totalTestResults       420 non-null   int64
15  lastModified           420 non-null   object
16  recovered              0 non-null     object
17  total                  420 non-null   int64
18  posNeg                 420 non-null   int64
19  deathIncrease          420 non-null   int64
...
23  totalTestResultsIncrease 420 non-null   int64
24  hash                   420 non-null   object
dtypes: float64(11), int64(10), object(4)
memory usage: 82.2+ KB

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

main Bootstrap v5.3.6 0 4 Git Graph Celda 1 de 79 Go Live Qodo Gen Ninja Colorize: 0 variables Colorize helper

1 mm de Buía Hoy 08:48 a. m. 25/06/2025

Archivo Editar Selección Ver Ir Ejecutar ...

Practicad

EXPLORADOR

my_practique6.ipynb

Practica 6: Preparación de Datos para su Análisis (ETL)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema

Seleccionar el kernel

Limpiar Datos

```
# elimino todos los json que tengan states = 56.000000
df_filtrado = df_comentarios[df_comentarios["states"] != 56.000000].copy()
```

```
# hago una consulta para ver el filtrado
df_filtrado.head()
```

	date	states	positive	negative	pending	hospitalizedCurrently	hospitalizedCumulative	inlcuCurrently	inlcuCumulative	onVentilatorCurrently	...	lastMod
357	20200315	51	5738.0	9396.0	2117.0	NaN	37.0	NaN	NaN	NaN	-	2020-03-15T24:00:00Z
358	20200314	51	4476.0	7409.0	1236.0	NaN	27.0	NaN	NaN	NaN	-	2020-03-14T24:00:00Z
359	20200313	51	3450.0	5533.0	1035.0	NaN	23.0	NaN	NaN	NaN	-	2020-03-13T24:00:00Z
360	20200312	51	2592.0	4292.0	673.0	NaN	17.0	NaN	NaN	NaN	-	2020-03-12T24:00:00Z
361	20200311	51	1915.0	3301.0	568.0	NaN	12.0	NaN	NaN	NaN	-	2020-03-11T24:00:00Z

5 rows x 25 columns

```
# Muestro metadatos actuales
df_filtrado.info()
```

main Bootstrap v5.3.6 0 4 Git Graph Celda 1 de 79 Go Live Qodo Gen Ninja Colorize: 0 variables Colorize helper

1 mm de lluvia Hoy 08:48 a. m. 25/06/2025

Archivo Editar Selección Ver Ir Ejecutar ...

Practicad

EXPLORADOR

my_practique6.ipynb

Practica 6: Preparación de Datos para su Análisis (ETL)

Generar + Código + Markdown | Ejecutar todo | Borrar todas las salidas | Esquema

Seleccionar el kernel

Transformación de datos

```
# Se transforman todos los states a id 2 a 99
# Muestro los primeros registros del dataframe
df_filtrado["states"].head()
```

```
#reemplazo de los datos
df_filtrado.loc[:, "states"] = df_filtrado["states"].replace(51, "Total")
```

```
#ejecutamos el comando para ver si se reemplazo
df_filtrado["states"].head()
```

	states
357	Total
358	Total
359	Total
360	Total
361	Total

Name: states, dtype: object

main Bootstrap v5.3.6 0 4 Git Graph Celda 1 de 79 Go Live Qodo Gen Ninja Colorize: 0 variables Colorize helper

1 mm de lluvia Hoy 08:48 a. m. 25/06/2025