Generating SQL for SQLite using Ollama, ChromaDB

This notebook runs through the process of using the `vanna` Python package to generate SQL using AI (RAG + LLMs) including connecting to a database and training. If you're not ready to train on your own database, you can still try it using a sample SQLite database.

Which LLM do you want to use?

- OpenAI via Vanna.AI (Recommended)
  Use Vanna.AI for free to generate your queries
- OpenAI
  Use OpenAI with your own API key
- Azure OpenAI
  If you have OpenAI models deployed on Azure
- [Selected] Ollama
  Use Ollama locally for free. Requires additional setup.
- Mistral via Mistral API
  If you have a Mistral API key
- Other LLM
  If you have a different LLM model

Where do you want to store the 'training' data?

- Vanna Hosted Vector DB (Recommended)
  Use Vanna.AIs hosted vector database (pgvector) for free. This is usable across machines with no additional setup.
- [Selected] ChromaDB
  Use ChromaDBs open-source vector database for free locally. No additional setup is necessary -- all database files will be created and stored locally.
- Marqo
  Use Marqo locally for free. Requires additional setup. Or use their hosted option.
- Other VectorDB
  Use any other vector database. Requires additional setup.

Setup

!pwd!pip install vanna!pip install 'vanna[chromadb]'!pip install ollama!pip show vanna # 0.5.5, 0.2.1!pip show ollama # 0.2.0

In [1]:
```
import warnings
import re
```

```
warnings.filterwarnings('ignore', category=DeprecationWarning, message='^Num
# warnings.filterwarnings('ignore', category=DeprecationWarning, message=re.

import os

import re
from time import time

from vanna.ollama import Ollama
from vanna.chromadb.chromadb_vector import ChromaDB_VectorStore
```

In [2]:
```
class MyVanna(ChromaDB_VectorStore, Ollama):
    def __init__(self, config=None):
        ChromaDB_VectorStore.__init__(self, config=config)
        Ollama.__init__(self, config=config)
```

In [3]:
```
file_db = "~/Downloads/chinook.sqlite"
model_name = "gemma2" # 'llama3'

clean_and_train = True  # False
```

In [4]:
```
config = {
    'model': model_name,   # 'mistral' # "starcoder2"
}
vn = MyVanna(config=config)
```

In [5]:
```
hostname = os.uname().nodename
print("Hostname:", hostname)
```

```
Hostname: ducklover1
```

In [6]:
```
file_db = os.path.abspath(os.path.expanduser(file_db))
vn.connect_to_sqlite(file_db)
```

In [7]:
```
vn.run_sql_is_set
```

Out[7]:
```
True
```

In [8]:
```
def remove_collections(collection_name=None, ACCEPTED_TYPES = ["sql", "ddl",
    if not collection_name:
        collections = ACCEPTED_TYPES
    elif isinstance(collection_name, str):
        collections = [collection_name]
    elif isinstance(collection_name, list):
        collections = collection_name
    else:
        print(f"\t{collection_name} is unknown: Skipped")
        return

    for c in collections:
        if not c in ACCEPTED_TYPES:
            print(f"\t{c} is unknown: Skipped")
            continue
```

```
        # print(f"vn.remove_collection('{c}')")
        vn.remove_collection(c)
```

In [9]:
```
def strip_brackets(ddl):
    """
    This function removes square brackets from table and column names in a D

    Args:
        ddl (str): The DDL script containing square brackets.

    Returns:
        str: The DDL script with square brackets removed.
    """
    # Use regular expressions to match and replace square brackets
    pattern = r"\[([^\]]+)]"  # Match any character except ] within square b
    return re.sub(pattern, r"\1", ddl)
```

In [10]:
```
if clean_and_train:
    remove_collections()
```

Training

SQLite sample database

You only need to train once. Do not train again unless you want to add more training data.

In [11]:
```
df_ddl = vn.run_sql("SELECT type, sql FROM sqlite_master WHERE sql is not nu
```

In [12]:
```
df_ddl
```

Out[12]:

	type	sql
0	table	CREATE TABLE "albums"\r\n(\r\n [AlbumId] IN...
1	table	CREATE TABLE sqlite_sequence(name,seq)
2	table	CREATE TABLE "artists"\r\n(\r\n [ArtistId] ...
3	table	CREATE TABLE "customers"\r\n(\r\n [Customer...
4	table	CREATE TABLE "employees"\r\n(\r\n [Employee...
5	table	CREATE TABLE "genres"\r\n(\r\n [GenreId] IN...
6	table	CREATE TABLE "invoices"\r\n(\r\n [InvoiceId...
7	table	CREATE TABLE "invoice_items"\r\n(\r\n [Invo...
8	table	CREATE TABLE "media_types"\r\n(\r\n [MediaT...
9	table	CREATE TABLE "playlists"\r\n(\r\n [Playlist...
10	table	CREATE TABLE "playlist_track"\r\n(\r\n [Pla...
11	table	CREATE TABLE "tracks"\r\n(\r\n [TrackId] IN...
12	index	CREATE INDEX [IFK_AlbumArtistId] ON "albums" (...
13	index	CREATE INDEX [IFK_CustomerSupportRepId] ON "cu...
14	index	CREATE INDEX [IFK_EmployeeReportsTo] ON "emplo...
15	index	CREATE INDEX [IFK_InvoiceCustomerId] ON "invoi...
16	index	CREATE INDEX [IFK_InvoiceLineInvoiceId] ON "in...
17	index	CREATE INDEX [IFK_InvoiceLineTrackId] ON "invo...
18	index	CREATE INDEX [IFK_PlaylistTrackTrackId] ON "pl...
19	index	CREATE INDEX [IFK_TrackAlbumId] ON "tracks" ([...
20	index	CREATE INDEX [IFK_TrackGenreId] ON "tracks" ([...
21	index	CREATE INDEX [IFK_TrackMediaTypeId] ON "tracks...
22	table	CREATE TABLE sqlite_stat1(tbl,idx,stat)

In [13]:
```
if clean_and_train:
    for ddl in df_ddl['sql'].to_list():
        ddl = strip_brackets(ddl)
        vn.train(ddl=ddl)

    # Sometimes you may want to add documentation about your business termin
    vn.train(documentation="In the chinook database invoice means order")
```

```
Adding ddl: CREATE TABLE "albums"
(
    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    Title NVARCHAR(160)  NOT NULL,
    ArtistId INTEGER  NOT NULL,
    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE TABLE sqlite_sequence(name,seq)
Adding ddl: CREATE TABLE "artists"
(
    ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    Name NVARCHAR(120)
)
Adding ddl: CREATE TABLE "customers"
(
    CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    FirstName NVARCHAR(40)  NOT NULL,
    LastName NVARCHAR(20)  NOT NULL,
    Company NVARCHAR(80),
    Address NVARCHAR(70),
    City NVARCHAR(40),
    State NVARCHAR(40),
    Country NVARCHAR(40),
    PostalCode NVARCHAR(10),
    Phone NVARCHAR(24),
    Fax NVARCHAR(24),
    Email NVARCHAR(60)  NOT NULL,
    SupportRepId INTEGER,
    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE TABLE "employees"
(
    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    LastName NVARCHAR(20)  NOT NULL,
    FirstName NVARCHAR(20)  NOT NULL,
    Title NVARCHAR(30),
    ReportsTo INTEGER,
    BirthDate DATETIME,
    HireDate DATETIME,
    Address NVARCHAR(70),
    City NVARCHAR(40),
    State NVARCHAR(40),
    Country NVARCHAR(40),
    PostalCode NVARCHAR(10),
    Phone NVARCHAR(24),
    Fax NVARCHAR(24),
    Email NVARCHAR(60),
    FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE TABLE "genres"
(
    GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    Name NVARCHAR(120)
```

```
)
Adding ddl: CREATE TABLE "invoices"
(
    InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    CustomerId INTEGER  NOT NULL,
    InvoiceDate DATETIME  NOT NULL,
    BillingAddress NVARCHAR(70),
    BillingCity NVARCHAR(40),
    BillingState NVARCHAR(40),
    BillingCountry NVARCHAR(40),
    BillingPostalCode NVARCHAR(10),
    Total NUMERIC(10,2)  NOT NULL,
    FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE TABLE "invoice_items"
(
    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    InvoiceId INTEGER  NOT NULL,
    TrackId INTEGER  NOT NULL,
    UnitPrice NUMERIC(10,2)  NOT NULL,
    Quantity INTEGER  NOT NULL,
    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE TABLE "media_types"
(
    MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    Name NVARCHAR(120)
)
Adding ddl: CREATE TABLE "playlists"
(
    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    Name NVARCHAR(120)
)
Adding ddl: CREATE TABLE "playlist_track"
(
    PlaylistId INTEGER  NOT NULL,
    TrackId INTEGER  NOT NULL,
    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),
    FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE TABLE "tracks"
(
    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    Name NVARCHAR(200)  NOT NULL,
    AlbumId INTEGER,
    MediaTypeId INTEGER  NOT NULL,
    GenreId INTEGER,
    Composer NVARCHAR(220),
    Milliseconds INTEGER  NOT NULL,
```

```
    Bytes INTEGER,
    UnitPrice NUMERIC(10,2)  NOT NULL,
    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId)
                ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId)
                ON DELETE NO ACTION ON UPDATE NO ACTION
)
Adding ddl: CREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)
Adding ddl: CREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRep
Id)
Adding ddl: CREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)
Adding ddl: CREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)
Adding ddl: CREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (Invoic
eId)
Adding ddl: CREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)
Adding ddl: CREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (Track
Id)
Adding ddl: CREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)
Adding ddl: CREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)
Adding ddl: CREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)
Adding ddl: CREATE TABLE sqlite_stat1(tbl,idx,stat)
Adding documentation....
```

In [14]:
```
# show training data
training_data = vn.get_training_data()
training_data
```

Out[14]:

	id	question	content	training_data_type
0	039f9d54-59f7-5f29-8c04-14dbc3e95671-ddl	None	CREATE TABLE "artists"\r\n(\r\n ArtistId IN...	ddl
1	0db84e3d-ef41-563c-803e-21c1b985dc19-ddl	None	CREATE TABLE "invoices"\r\n(\r\n InvoiceId ...	ddl
2	10cba811-ddba-5042-9e90-d764dfcd1629-ddl	None	CREATE INDEX IFK_InvoiceCustomerId ON "invoice...	ddl
3	2c711317-b93d-5f60-a728-cb1c6fcbc040-ddl	None	CREATE INDEX IFK_CustomerSupportRepId ON "cust...	ddl
4	37319c81-65f7-50ee-956b-795de244bee5-ddl	None	CREATE TABLE sqlite_stat1(tbl,idx,stat)	ddl
5	40bd77cd-e1de-5872-8693-624117ff413c-ddl	None	CREATE INDEX IFK_InvoiceLineInvoiceId ON "invo...	ddl
6	41130543-7164-562a-90a7-0fd0a409c154-ddl	None	CREATE TABLE "albums"\r\n(\r\n AlbumId INTE...	ddl
7	458debc8-8082-5450-a17a-66028bd55ace-ddl	None	CREATE TABLE "playlists"\r\n(\r\n PlaylistI...	ddl
8	4815f3fd-925b-53ce-9dfa-0e4285d5abd3-ddl	None	CREATE TABLE "invoice_items"\r\n(\r\n Invoi...	ddl
9	48d484e9-984c-58ff-b391-75521c69d486-ddl	None	CREATE INDEX IFK_PlaylistTrackTrackId ON "play...	ddl
10	551e1120-a6ee-554f-8b8a-ccf4f22d3636-ddl	None	CREATE INDEX IFK_AlbumArtistId ON "albums" (Ar...	ddl
11	5ff4911e-45c1-5a59-9566-243a9b6a3320-ddl	None	CREATE TABLE "employees"\r\n(\r\n EmployeeI...	ddl
12	65df0648-bf05-5f75-9365-c21f54b2302d-ddl	None	CREATE TABLE "media_types"\r\n(\r\n MediaTy...	ddl
13	6b585176-e66d-5b23-8d86-ca8a80e3af3d-ddl	None	CREATE INDEX IFK_EmployeeReportsTo ON "employe...	ddl
14	868758b8-e018-55e7-8cc3-75c0e6d211c8-ddl	None	CREATE INDEX IFK_TrackAlbumId ON "tracks" (Alb...	ddl
15	9ea4613d-c1be-5a77-ada9-c54ee3f0cab7-ddl	None	CREATE INDEX IFK_TrackMediaTypeId ON "tracks" ...	ddl
16	a9c9a852-608d-5ef2-aede-26ba098d83d1-	None	CREATE INDEX IFK_TrackGenreId ON "tracks" (Gen...	ddl

	id	question	content	training_data_type
	ddl			
17	b42cc9e1-9219-5a42-9a06-de906f76239e-ddl	None	CREATE TABLE "tracks"\r\n(\r\n TrackId INTE...	ddl
18	c387b9d2-5ff4-5a07-8364-f5dab45bb2a9-ddl	None	CREATE TABLE "genres"\r\n(\r\n GenreId INTE...	ddl
19	d654f328-dc36-549e-84c3-06ee0db7e0f7-ddl	None	CREATE TABLE "playlist_track"\r\n(\r\n Play...	ddl
20	d93f0d68-023d-5afb-8121-ba346699d318-ddl	None	CREATE TABLE "customers"\r\n(\r\n CustomerI...	ddl
21	e5879308-329e-543f-a693-0c14e2f9972e-ddl	None	CREATE INDEX IFK_InvoiceLineTrackId ON "invoic...	ddl
22	ea84418b-1a28-59b4-a1f4-2fb674208adc-ddl	None	CREATE TABLE sqlite_sequence(name,seq)	ddl
0	2b4dda0a-a6ac-5e34-8f76-e41c0734d55e-doc	None	In the chinook database invoice means order	documentation

Asking the AI

Whenever you ask a new question, it will find the 10 most relevant pieces of training data and use it as part of the LLM prompt to generate the SQL.

In [15]:
```
ts_start = time()
```

```
SELECT name FROM sqlite_master WHERE type = 'table';
```

In [16]:
```
vn.ask(question="Can you list all tables in the SQLite database catalog?")
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

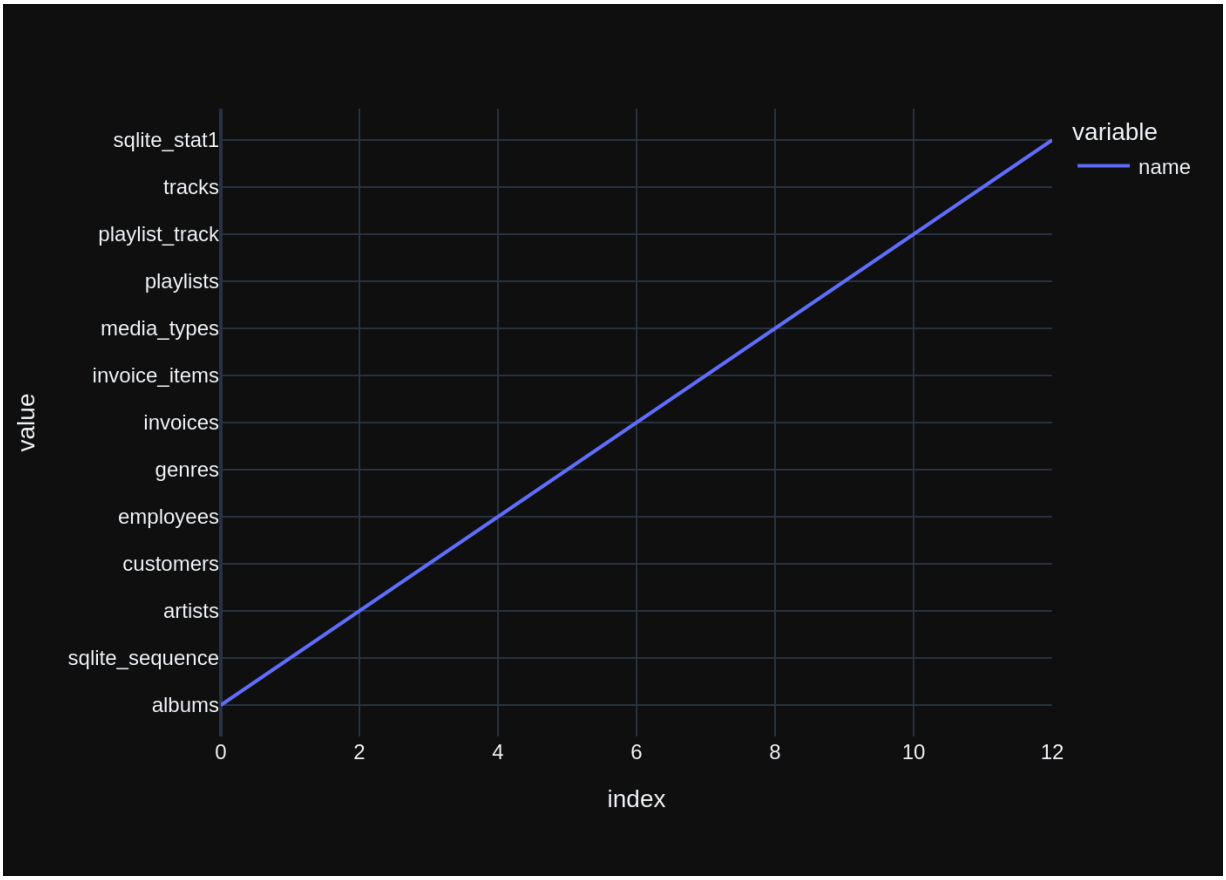
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCR
EATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "playlists"\r\n(\r\n
PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR
(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "trac
ks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    N
ame NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTE
GER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Genr
eId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaType
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "me
dia_types"\r\n(\r\n    MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "artists"\r\n(\r\n    Arti
stId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r
\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMAR
Y KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    Tra
ckId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Qua
ntity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoice
s" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREI
GN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n    Playlis
tId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT P
K_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (Play
listId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON U
PDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "album
s"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Ti
tle NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREI
GN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook dat
abase invoice means order\n\n===Response Guidelines \n1. If the provided con
text is sufficient, please generate a valid SQL query without any explanatio
ns for the question. \n2. If the provided context is almost sufficient but r
equires knowledge of a specific string in a particular column, please genera
te an intermediate SQL query to find the distinct strings in that column. Pr
epend the query with a comment saying intermediate_sql \n3. If the provided
context is insufficient, please explain why it can\'t be generated. \n4. Ple
ase use the most relevant table(s). \n5. If the question has been asked and
answered before, please repeat the answer exactly as it was given before.
\n'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite dat
abase catalog?'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE TABLE s
```

qlite_sequence(name,seq)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistI
d INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)
\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"tracks\"\r
\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name N
VARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER
NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Mill
iseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(1
0,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreI
d) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTy
peId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE
\"media_types\"\r\n(\r\n   MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NO
T NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n
ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(12
0)\r\n)\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n
TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n
Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invo
ices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   F
OREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n
PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONST
RAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KE
Y (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\"
(TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TAB
LE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NUL
L,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nI
n the chinook database invoice means order\n\n===Response Guidelines \n1. If
the provided context is sufficient, please generate a valid SQL query withou
t any explanations for the question. \n2. If the provided context is almost
sufficient but requires knowledge of a specific string in a particular colum
n, please generate an intermediate SQL query to find the distinct strings in
that column. Prepend the query with a comment saying intermediate_sql \n3. I
f the provided context is insufficient, please explain why it can't be gener
ated. \n4. Please use the most relevant table(s). \n5. If the question has b
een asked and answered before, please repeat the answer exactly as it was gi
ven before. \n"}, {"role": "user", "content": "Can you list all tables in th
e SQLite database catalog?"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:37:49.471387611Z',
'message': {'role': 'assistant', 'content': "SELECT name FROM sqlite_master
WHERE type='table'; \r\n"}, 'done_reason': 'stop', 'done': True, 'total_dur
ation': 42445754181, 'load_duration': 4380960695, 'prompt_eval_count': 849,
'prompt_eval_duration': 34552394000, 'eval_count': 15, 'eval_duration': 3412
909000}
LLM Response: SELECT name FROM sqlite_master WHERE type='table';

Info: Output from LLM: SELECT name FROM sqlite_master WHERE type='table';

Extracted SQL: SELECT name FROM sqlite_master WHERE type='table'
SELECT name FROM sqlite_master WHERE type='table'

```
              name
0           albums
1   sqlite_sequence
2          artists
3        customers
4        employees
5           genres
6         invoices
7    invoice_items
8      media_types
9        playlists
10  playlist_track
11          tracks
12    sqlite_stat1
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: 'Ca
n you list all tables in the SQLite database catalog?'\n\nThe DataFrame was
produced using this query: SELECT name FROM sqlite_master WHERE type='tabl
e'\n\nThe following is information about the resulting pandas DataFrame 'd
f': \nRunning df.dtypes gives:\n name    object\ndtype: object"}, {"role":
"user", "content": "Can you generate the Python plotly code to chart the res
ults of the dataframe? Assume the data is in a pandas dataframe called 'df'.
If there is only one value in the dataframe, use an Indicator. Respond with
only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:38:17.42997324Z', 'm
essage': {'role': 'assistant', 'content': '```python\nimport plotly.express
as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df[\'name\'].iloc
[0],\n    title="SQLite Database Tables",\n    domain={\'x\': [0, 1], \'y\':
[0, 1]},\n  )\nelse:\n  px.bar(df, x=\'name\', y=\'\', title="SQLite Databas
e Tables")\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 27
926690954, 'load_duration': 19893477, 'prompt_eval_count': 149, 'prompt_eval
_duration': 6010050000, 'eval_count': 97, 'eval_duration': 21805952000}
```


variable
name
value
sqlite_stat1
tracks
playlist_track
playlists
media_types
invoice_items
invoices
genres
employees
customers
artists
sqlite_sequence
albums
0
2
4
6
8
10
12
index

Out[16]:
```
("SELECT name FROM sqlite_master WHERE type='table'",
                name
 0             albums
 1   sqlite_sequence
 2            artists
 3          customers
 4          employees
 5             genres
 6           invoices
 7     invoice_items
 8       media_types
 9         playlists
 10   playlist_track
 11           tracks
 12     sqlite_stat1,
 Figure({
     'data': [{'hovertemplate': 'variable=name<br>index=%{x}<br>value=%{y}<
extra></extra>',
               'legendgroup': 'name',
               'line': {'color': '#636efa', 'dash': 'solid'},
               'marker': {'symbol': 'circle'},
               'mode': 'lines',
               'name': 'name',
               'orientation': 'v',
               'showlegend': True,
               'type': 'scatter',
               'x': array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11,
12]),
               'xaxis': 'x',
               'y': array(['albums', 'sqlite_sequence', 'artists', 'custome
rs', 'employees',
                           'genres', 'invoices', 'invoice_items', 'media_ty
pes', 'playlists',
                           'playlist_track', 'tracks', 'sqlite_stat1'], dty
pe=object),
               'yaxis': 'y'}],
      'layout': {'legend': {'title': {'text': 'variable'}, 'tracegroupgap':
0},
                 'margin': {'t': 60},
                 'template': '...',
                 'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'index'}},
                 'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'value'}}}
 }))
```

In [17]:
```
vn.ask(question="which table stores customer's orders")
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "playlists"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "media_types"\r\n(\r\n   MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': 'Can you list all tables

```
in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT
name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': "w
hich table stores customer's orders"}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n
InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    B
illingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER
NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER
NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n    CustomerId INTEGER PRI
MARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r
\n    LastName NVARCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    A
ddress NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r
\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVA
RCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n
SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"employees\"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVA
RCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r
\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR
(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NV
ARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo)
REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE \"p
laylists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)
\n\nCREATE TABLE \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGE
R  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playl
ist_track\"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER
NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, Track
Id),\r\n    FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n    MediaTypeId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===A
dditional Context \n\nIn the chinook database invoice means order\n\n===Resp
onse Guidelines \n1. If the provided context is sufficient, please generate
```

```
a valid SQL query without any explanations for the question. \n2. If the pro
vided context is almost sufficient but requires knowledge of a specific stri
ng in a particular column, please generate an intermediate SQL query to find
the distinct strings in that column. Prepend the query with a comment saying
intermediate_sql \n3. If the provided context is insufficient, please explai
n why it can't be generated. \n4. Please use the most relevant table(s). \n
5. If the question has been asked and answered before, please repeat the ans
wer exactly as it was given before. \n"}, {"role": "user", "content": "Can y
ou list all tables in the SQLite database catalog?"}, {"role": "assistant",
"content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "u
ser", "content": "which table stores customer's orders"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:39:09.56174088Z', 'm
essage': {'role': 'assistant', 'content': 'invoices  \r\n'}, 'done_reason':
'stop', 'done': True, 'total_duration': 51696360533, 'load_duration': 160514
82, 'prompt_eval_count': 1104, 'prompt_eval_duration': 50469647000, 'eval_co
unt': 5, 'eval_duration': 980030000}
LLM Response: invoices

invoices

Couldn't run sql:  Execution failed on sql 'invoices
': near "invoices": syntax error
```

In [18]:
```
vn.ask(question="How many customers are there")
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

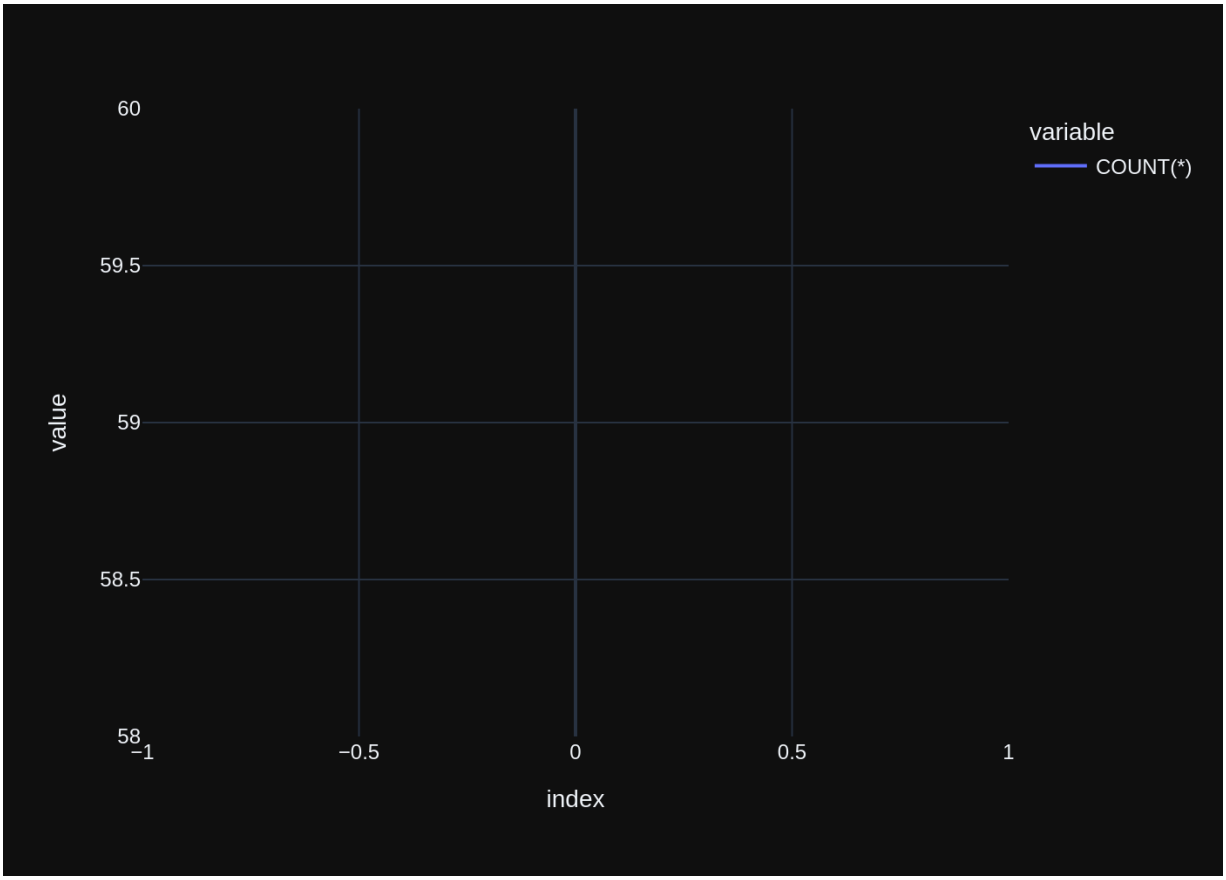
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCRE
ATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREME
NT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCH
AR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(7
0),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVAR
CHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_
InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "invoice_item
s"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPr
ice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREI
GN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks"
(TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IND
EX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE
"albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOR
EIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACT
ION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invo
ice_items" (TrackId)\n\nCREATE TABLE "employees"\r\n(\r\n    EmployeeId INTE
GER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NU
LL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n
ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n
Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r
\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVA
RCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIG
N KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlists"\r\n(\r\n    Playl
istId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)
\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means o
rder\n\n===Response Guidelines \n1. If the provided context is sufficient, p
lease generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can\'t be generated. \n4. Please use the most rele
vant table(s). \n5. If the question has been asked and answered before, plea
se repeat the answer exactly as it was given before. \n'}, {'role': 'user',
'content': 'Can you list all tables in the SQLite database catalog?'}, {'rol
e': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {'role': 'user', 'content': 'How many customers are there'}]
Info: Ollama parameters:
model=gemma2:latest,
```

```
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n
InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    B
illingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
INDEX IFK_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TAB
LE \"customers\"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NO
T NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(2
0)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r\n
City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r
\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHA
R(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n
FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustome
rId ON \"invoices\" (CustomerId)\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n
InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId I
NTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC
(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (Inv
oiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON U
PDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_
InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"albu
ms\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOR
EIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"i
nvoice_items\" (TrackId)\n\nCREATE TABLE \"employees\"\r\n(\r\n    EmployeeI
d INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)
NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(3
0),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DAT
ETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State N
VARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r
\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(6
0),\r\n    FOREIGN KEY (ReportsTo) REFERENCES \"employees\" (EmployeeId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playlis
ts\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook databa
se invoice means order\n\n===Response Guidelines \n1. If the provided contex
t is sufficient, please generate a valid SQL query without any explanations
for the question. \n2. If the provided context is almost sufficient but requ
ires knowledge of a specific string in a particular column, please generate
an intermediate SQL query to find the distinct strings in that column. Prepe
nd the query with a comment saying intermediate_sql \n3. If the provided con
text is insufficient, please explain why it can't be generated. \n4. Please
use the most relevant table(s). \n5. If the question has been asked and answ
ered before, please repeat the answer exactly as it was given before. \n"},
{"role": "user", "content": "Can you list all tables in the SQLite database
catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master
```

```
WHERE type='table'"}, {"role": "user", "content": "How many customers are th
ere"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:39:50.61267405Z', 'm
essage': {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers;
\n'}, 'done_reason': 'stop', 'done': True, 'total_duration': 41008898334, 'l
oad_duration': 24204733, 'prompt_eval_count': 1013, 'prompt_eval_duration':
38820518000, 'eval_count': 9, 'eval_duration': 1932771000}
LLM Response: SELECT COUNT(*) FROM customers;

Info: Output from LLM: SELECT COUNT(*) FROM customers;

Extracted SQL: SELECT COUNT(*) FROM customers
SELECT COUNT(*) FROM customers
   COUNT(*)
0        59
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: 'Ho
w many customers are there'\n\nThe DataFrame was produced using this query:
SELECT COUNT(*) FROM customers\n\nThe following is information about the res
ulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n COUNT(*)    int64
\ndtype: object"}, {"role": "user", "content": "Can you generate the Python
plotly code to chart the results of the dataframe? Assume the data is in a p
andas dataframe called 'df'. If there is only one value in the dataframe, us
e an Indicator. Respond with only Python code. Do not answer with any explan
ations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:40:04.843770134Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nfig = px.indicator(df, \n                    value="COUNT(*)", \n
title="Number of Customers")\n```'}, 'done_reason': 'stop', 'done': True, 't
otal_duration': 14205554037, 'load_duration': 22220565, 'prompt_eval_count':
141, 'prompt_eval_duration': 5730552000, 'eval_count': 38, 'eval_duration':
8364427000}
```



Out[18]:
```
('SELECT COUNT(*) FROM customers',
    COUNT(*)
 0        59,
 Figure({
     'data': [{'hovertemplate': 'variable=COUNT(*)<br>index=%{x}<br>value=%
{y}<extra></extra>',
               'legendgroup': 'COUNT(*)',
               'line': {'color': '#636efa', 'dash': 'solid'},
               'marker': {'symbol': 'circle'},
               'mode': 'lines',
               'name': 'COUNT(*)',
               'orientation': 'v',
               'showlegend': True,
               'type': 'scatter',
               'x': array([0]),
               'xaxis': 'x',
               'y': array([59]),
               'yaxis': 'y'}],
     'layout': {'legend': {'title': {'text': 'variable'}, 'tracegroupgap':
0},
                'margin': {'t': 60},
                'template': '...',
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'index'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'value'}}}
 }))
```

In [ ]:

In [19]:
```
vn.ask(question="what are the top 5 countries that customers come from?")
```

```
Number of requested results 10 is greater than number of elements in index
2, updating n_results = 2
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

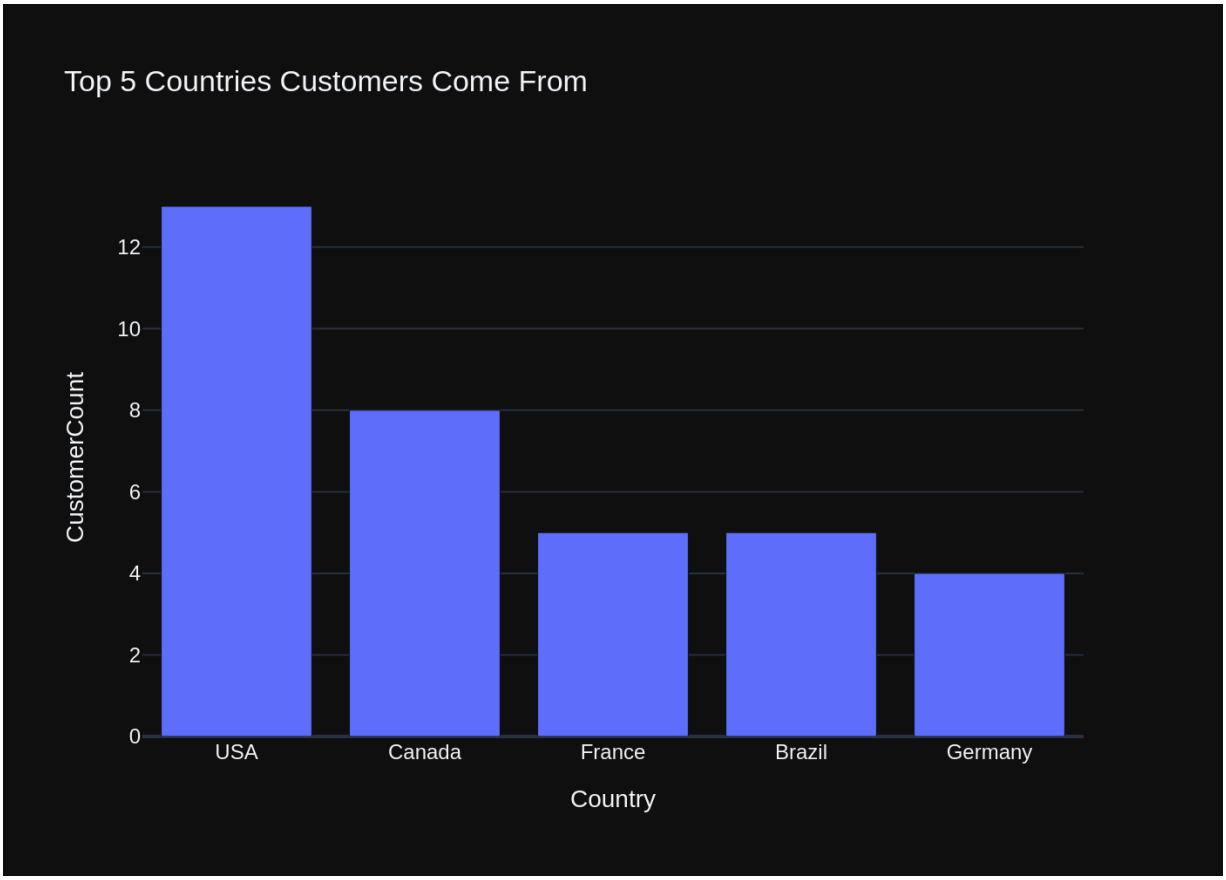
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "media_types"\r\n(\r\n   MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any ex

planations for the question. \n2. If the provided context is almost sufficie
nt but requires knowledge of a specific string in a particular column, pleas
e generate an intermediate SQL query to find the distinct strings in that co
lumn. Prepend the query with a comment saying intermediate_sql \n3. If the p
rovided context is insufficient, please explain why it can\'t be generated.
\n4. Please use the most relevant table(s). \n5. If the question has been as
ked and answered before, please repeat the answer exactly as it was given be
fore. \n'}, {'role': 'user', 'content': 'How many customers are there'}, {'r
ole': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role': 'u
ser', 'content': 'Can you list all tables in the SQLite database catalog?'},
{'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type
='table'"}, {'role': 'user', 'content': 'what are the top 5 countries that c
ustomers come from?'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR
(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r
\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR
(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax
NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEG
ER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (EmployeeId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoi
ce_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r
\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NUL
L,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFE
RENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n   MediaTypeId INTEGER PRIMAR
Y KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX
IFK_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"e
mployees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NO
T NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDat
e DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   Ci
ty NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2
4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e
mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGE
R NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId)

\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playl
ist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, Track
Id),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE \"trac
ks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INT
EGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n
Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (Albu
mId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (Ge
nreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (Medi
aTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additi
onal Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can't be generated. \n4. Please use the most relevant table(s). \n5. If
the question has been asked and answered before, please repeat the answer ex
actly as it was given before. \n"}, {"role": "user", "content": "How many cu
stomers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM
customers"}, {"role": "user", "content": "Can you list all tables in the SQL
ite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM
sqlite_master WHERE type='table'"}, {"role": "user", "content": "what are th
e top 5 countries that customers come from?"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:41:12.771356777Z',
'message': {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS Cus
tomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount
DESC\r\nLIMIT 5; \n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_du
ration': 67824425179, 'load_duration': 27647779, 'prompt_eval_count': 1283,
'prompt_eval_duration': 59493781000, 'eval_count': 33, 'eval_duration': 8006
047000}
LLM Response: SELECT Country, COUNT(*) AS CustomerCount
FROM customers
GROUP BY Country
ORDER BY CustomerCount DESC
LIMIT 5;

Info: Output from LLM: SELECT Country, COUNT(*) AS CustomerCount
FROM customers
GROUP BY Country
ORDER BY CustomerCount DESC
LIMIT 5;

```
Extracted SQL: SELECT Country, COUNT(*) AS CustomerCount
FROM customers
GROUP BY Country
ORDER BY CustomerCount DESC
LIMIT 5
SELECT Country, COUNT(*) AS CustomerCount
FROM customers
GROUP BY Country
ORDER BY CustomerCount DESC
LIMIT 5
   Country  CustomerCount
0      USA             13
1   Canada              8
2   France              5
3   Brazil              5
4  Germany              4
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: 'wh
at are the top 5 countries that customers come from?'\n\nThe DataFrame was p
roduced using this query: SELECT Country, COUNT(*) AS CustomerCount \r\nFROM
customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5\n\nT
he following is information about the resulting pandas DataFrame 'df': \nRun
ning df.dtypes gives:\n Country          object\nCustomerCount     int64\ndt
ype: object"}, {"role": "user", "content": "Can you generate the Python plot
ly code to chart the results of the dataframe? Assume the data is in a panda
s dataframe called 'df'. If there is only one value in the dataframe, use an
Indicator. Respond with only Python code. Do not answer with any explanation
s -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:41:39.3206708Z', 'me
ssage': {'role': 'assistant', 'content': '```python\nimport plotly.express a
s px\n\nif df.shape[0] == 1:\n  fig = px.indicator(df, name="CustomerCount",
value=\'CustomerCount\', title=\'Top Country\')\nelse:\n  fig = px.bar(df, x
=\'Country\', y=\'CustomerCount\', title=\'Top 5 Countries Customers Come Fr
om\')\n\nfig.show() \n```'}, 'done_reason': 'stop', 'done': True, 'total_dur
ation': 26529182088, 'load_duration': 22473492, 'prompt_eval_count': 175, 'p
rompt_eval_duration': 7133568000, 'eval_count': 87, 'eval_duration': 1924320
4000}
```


Top 5 Countries Customers Come From
CustomerCount
0
2
4
6
8
10
12
USA
Canada
France
Brazil
Germany
Country

Out[19]:
```
('SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY
Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5',
    Country  CustomerCount
 0      USA             13
 1   Canada              8
 2   France              5
 3   Brazil              5
 4  Germany              4,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'Country=%{x}<br>CustomerCount=%{y}<extra>
</extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['USA', 'Canada', 'France', 'Brazil', 'Germany'],
dtype=object),
               'xaxis': 'x',
               'y': array([13,  8,  5,  5,  4]),
               'yaxis': 'y'}],
      'layout': {'barmode': 'relative',
                 'legend': {'tracegroupgap': 0},
                 'template': '...',
                 'title': {'text': 'Top 5 Countries Customers Come From'},
                 'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Country'}},
                 'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'CustomerCount'}}}
  }))
```

More SQL questions

see `sample-sql-queries-sqlite-chinook.ipynb`

In [20]:
```
question = """
    List all albums and their corresponding artist names
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
3, updating n_results = 3
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_AlbumArtistId ON "albums" (A
rtistId)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AU
TOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId
INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (Artis
tId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "t
racks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INT
EGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Genr
eId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaType
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK
_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "artists"\r\n(\r\n    Ar
tistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)
\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX
IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE TABLE "play
lists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n
\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\n\n===Additi
onal Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can\'t be generated. \n4. Please use the most relevant table(s). \n5. I
f the question has been asked and answered before, please repeat the answer
exactly as it was given before. \n'}, {'role': 'user', 'content': 'Can you l
ist all tables in the SQLite database catalog?'}, {'role': 'assistant', 'con
tent': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'use
r', 'content': 'what are the top 5 countries that customers come from?'},
{'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS CustomerCount
\r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIM
IT 5'}, {'role': 'user', 'content': 'How many customers are there'}, {'rol
e': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role': 'use
r', 'content': '  \n    List all albums and their corresponding artist names
\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n
\nCREATE TABLE \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER
NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r
```

\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks
\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Na
me NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEG
ER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n
Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (Albu
mId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (Ge
nreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (Medi
aTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDE
X IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n
(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NV
ARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n
\nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nC
REATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINC
REMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"genres\"\r
\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name N
VARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaT
ypeId)\n\n\n===Additional Context \n\nIn the chinook database invoice means
order\n\n===Response Guidelines \n1. If the provided context is sufficient,
please generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can't be generated. \n4. Please use the most relev
ant table(s). \n5. If the question has been asked and answered before, pleas
e repeat the answer exactly as it was given before. \n"}, {"role": "user",
"content": "Can you list all tables in the SQLite database catalog?"}, {"rol
e": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {"role": "user", "content": "what are the top 5 countries that custome
rs come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*)
AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY Customer
Count DESC\r\nLIMIT 5"}, {"role": "user", "content": "How many customers are
there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customers"},
{"role": "user", "content": " \n   List all albums and their corresponding
artist names \n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:42:19.891324561Z',
'message': {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.N
ame\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId; \n'},
'done_reason': 'stop', 'done': True, 'total_duration': 40471677704, 'load_du
ration': 21783928, 'prompt_eval_count': 757, 'prompt_eval_duration': 3157995
3000, 'eval_count': 37, 'eval_duration': 8435409000}
LLM Response: SELECT
    a.Title,
    ar.Name
FROM albums AS a
JOIN artists AS ar ON a.ArtistId = ar.ArtistId;

Info: Output from LLM: SELECT
    a.Title,
    ar.Name
FROM albums AS a
JOIN artists AS ar ON a.ArtistId = ar.ArtistId;

```
Extracted SQL: SELECT
    a.Title,
    ar.Name
FROM albums AS a
JOIN artists AS ar ON a.ArtistId = ar.ArtistId
SELECT
    a.Title,
    ar.Name
FROM albums AS a
JOIN artists AS ar ON a.ArtistId = ar.ArtistId
                                                 Title  \
0                For For Those About To Rock We Salute You
1                                     Balls to the Wall
2                                     Restless and Wild
3                                     Let There Be Rock
4                                              Big Ones
..                                                  ...
342                             Respighi:Pines of Rome
343  Schubert: The Late String Quartets & String Qu...
344                               Monteverdi: L'Orfeo
345                            Mozart: Chamber Music
346  Koyaanisqatsi (Soundtrack from the Motion Pict...

                                                  Name
0                                                AC/DC
1                                              Accept
2                                              Accept
3                                                AC/DC
4                                             Aerosmith
..                                                  ...
342                                    Eugene Ormandy
343                               Emerson String Quartet
344  C. Monteverdi, Nigel Rogers - Chiaroscuro; Lon...
345                                     Nash Ensemble
346                               Philip Glass Ensemble

[347 rows x 2 columns]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all albums and their corresponding artist names  \n'\n\nThe DataF
rame was produced using this query: SELECT \n    a.Title,\n    ar.Name\nFROM
albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId\n\nThe following
is information about the resulting pandas DataFrame 'df': \nRunning df.dtype
s gives:\n Title    object\nName     object\ndtype: object"}, {"role": "use
r", "content": "Can you generate the Python plotly code to chart the results
of the dataframe? Assume the data is in a pandas dataframe called 'df'. If t
here is only one value in the dataframe, use an Indicator. Respond with only
Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:42:48.025921047Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
```

```
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df[\'Title\'].il
oc[0],\n    title="Album and Artist",\n    text=f"{df[\'Title\'].iloc[0]} by
{df[\'Name\'].iloc[0]}"\n  )\nelse:\n  px.bar(df, x=\'Name\', y=\'Title\')\n
```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 28107487780,
'load_duration': 19385367, 'prompt_eval_count': 178, 'prompt_eval_duration':
6847639000, 'eval_count': 96, 'eval_duration': 21195267000}
Couldn't run plotly code: 'NoneType' object has no attribute 'show'
```

```
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1684, in ask
    img_bytes = fig.to_image(format="png", scale=2)
                ^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'to_image'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1687, in ask
    fig.show()
    ^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'show'
```

In [21]:
```
question = """
    Find all tracks with a name containing "What" (case-insensitive)
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
4, updating n_results = 4
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```
```

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON "tracks" (Ge
nreId)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackI
d)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGE
R,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Comp
oser NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTE
GER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId)
REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DEL
ETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENC
ES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON\r\n)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE IND
EX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE TABLE "playlist_
track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT
NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackI
d),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n
\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) RE
FERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n
\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE TABLE "pla
ylists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n
\n\n===Additional Context \n\nIn the chinook database invoice means order\n
\n===Response Guidelines \n1. If the provided context is sufficient, please
generate a valid SQL query without any explanations for the question. \n2. I
f the provided context is almost sufficient but requires knowledge of a spec
ific string in a particular column, please generate an intermediate SQL quer
y to find the distinct strings in that column. Prepend the query with a comm
ent saying intermediate_sql \n3. If the provided context is insufficient, pl
ease explain why it can\'t be generated. \n4. Please use the most relevant t
able(s). \n5. If the question has been asked and answered before, please rep
eat the answer exactly as it was given before. \n'}, {'role': 'user', 'conte
nt': '  \n    List all albums and their corresponding artist names  \n'},
{'role': 'assistant', 'content': 'SELECT \n    a.Title,\n    ar.Name\nFROM a
lbums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'use
r', 'content': 'Can you list all tables in the SQLite database catalog?'},
{'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type
='table'"}, {'role': 'user', 'content': 'what are the top 5 countries that c
ustomers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COU
NT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY Cu
stomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': 'How many custome
rs are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM custo
mers'}, {'role': 'user', 'content': '  \n    Find all tracks with a name con
taining "What" (case-insensitive)\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
```

ons. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\n
CREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCRE
ATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT N
OT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n
MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARC
HAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDEX
IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE TABLE \"playlist_
track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NO
T NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackI
d),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId)
REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackI
d)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\nCREATE TABL
E \"playlists\"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"genres\"\r\n(\r\n
GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(12
0)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means
order\n\n===Response Guidelines \n1. If the provided context is sufficient,
please generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can't be generated. \n4. Please use the most relev
ant table(s). \n5. If the question has been asked and answered before, pleas
e repeat the answer exactly as it was given before. \n"}, {"role": "user",
"content": " \n   List all albums and their corresponding artist names
\n"}, {"role": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\n
FROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role":
"user", "content": "Can you list all tables in the SQLite database catalo
g?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE
type='table'"}, {"role": "user", "content": "what are the top 5 countries th
at customers come from?"}, {"role": "assistant", "content": "SELECT Country,
COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY
CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": "How many custo
mers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM cus
tomers"}, {"role": "user", "content": " \n   Find all tracks with a name c
ontaining \"What\" (case-insensitive)\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:43:28.163143282Z',
'message': {'role': 'assistant', 'content': "SELECT * \r\nFROM tracks\r\nWHE
RE LOWER(Name) LIKE '%what%'; \n\n\n\n"}, 'done_reason': 'stop', 'done': Tru
e, 'total_duration': 40081813525, 'load_duration': 22732934, 'prompt_eval_co
unt': 838, 'prompt_eval_duration': 34812598000, 'eval_count': 21, 'eval_dura
tion': 4706562000}
LLM Response: SELECT *
FROM tracks
WHERE LOWER(Name) LIKE '%what%';

```
Info: Output from LLM: SELECT *
FROM tracks
WHERE LOWER(Name) LIKE '%what%';



Extracted SQL: SELECT *
FROM tracks
WHERE LOWER(Name) LIKE '%what%'
SELECT *
FROM tracks
WHERE LOWER(Name) LIKE '%what%'
    TrackId                                          Name  AlbumId  \
0        26                                What It Takes        5
1        88                                 What You Are       10
2       130                             Do what cha wanna       13
3       342                   What is and Should Never Be       30
4       607                                      So What       48
5       960                                   What A Day       76
6      1000                                What If I Do?       80
7      1039                              What Now My Love       83
8      1145                                   Whatsername       89
9      1440            Whatever It Is, I Just Can't Stop      116
10     1469                         Look What You've Done      119
11     1470                             Get What You Need      119
12     1628              What Is And What Should Never Be      133
13     1778  You're What's Happening (In The World Today)      146
14     1823                                      So What      149
15     2772          I Don't Know What To Do With Myself      223
16     2884                                What Kate Did      231
17     2893                   Whatever the Case May Be      230
18     2992    I Still Haven't Found What I'm Looking for      237
19     3007    I Still Haven't Found What I'm Looking For      238
20     3258              Whatever Gets You Thru the Night      255
21     3475                         What Is It About Men      322

    MediaTypeId  GenreId                                         Composer
\
0             1        1            Steven Tyler, Joe Perry, Desmond Child
1             1        1                          Audioslave/Chris Cornell
2             1        2                                       George Duke
3             1        1                        Jimmy Page/Robert Plant
4             1        2                                       Miles Davis
5             1        1             Mike Bordin, Billy Gould, Mike Patton
6             1        1  Dave Grohl, Taylor Hawkins, Nate Mendel, Chris...
7             1       12         carl sigman/gilbert becaud/pierre leroyer
8             1        4                                         Green Day
9             1        1                                  Jay Kay/Kay, Jay
10            1        4                                         N. Cester
11            1        4                    C. Cester/C. Muncey/N. Cester
12            1        1                         Jimmy Page, Robert Plant
13            1       14             Allen Story/George Gordy/Robert Gordy
```

```
14            1        3                                        Culmer/Exalt
15            1        7                                                None
16            3       19                                                None
17            3       19                                                None
18            1        1      Bono/Clayton, Adam/Mullen Jr., Larry/The Edge
19            1        1                                                  U2
20            2        9                                                None
21            2        9  Delroy "Chris" Cooper, Donovan Jackson, Earl C...

    Milliseconds      Bytes  UnitPrice
0         310622   10144730       0.99
1         249391    5988186       0.99
2         274155    9018565       0.99
3         260675    8497116       0.99
4         564009   18360449       0.99
5         158275    5203430       0.99
6         302994    9929799       0.99
7         149995    4913383       0.99
8         252316    8244843       0.99
9         247222    8249453       0.99
10        230974    7517083       0.99
11        247719    8043765       0.99
12        287973    9369385       0.99
13        142027    4631104       0.99
14        189152    6162894       0.99
15        221387    7251478       0.99
16       2610250  484583988       1.99
17       2616410  183867185       1.99
18        353567   11542247       0.99
19        280764    9306737       0.99
20        215084    3499018       0.99
21        209573    3426106       0.99
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find all tracks with a name containing \"What\" (case-insensitive)
\n'\n\nThe DataFrame was produced using this query: SELECT * \r\nFROM tracks
\r\nWHERE LOWER(Name) LIKE '%what%'\n\nThe following is information about th
e resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n TrackId
int64\nName             object\nAlbumId           int64\nMediaTypeId       i
nt64\nGenreId           int64\nComposer         object\nMilliseconds      in
t64\nBytes             int64\nUnitPrice       float64\ndtype: object"}, {"ro
le": "user", "content": "Can you generate the Python plotly code to chart th
e results of the dataframe? Assume the data is in a pandas dataframe called
'df'. If there is only one value in the dataframe, use an Indicator. Respond
with only Python code. Do not answer with any explanations -- just the cod
e."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:43:53.532271089Z',
'message': {'role': 'assistant', 'content': "```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df['Name'].iloc
[0],\n    title='Track Name',\n    type='number'\n  )\nelse:\n  px.bar(df, x
```

```
='Name', y='UnitPrice')\n```"}, 'done_reason': 'stop', 'done': True, 'total_
duration': 25342129808, 'load_duration': 19970969, 'prompt_eval_count': 214,
'prompt_eval_duration': 8359701000, 'eval_count': 77, 'eval_duration': 16916
807000}
Couldn't run plotly code:  'NoneType' object has no attribute 'show'
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1684, in ask
    img_bytes = fig.to_image(format="png", scale=2)
                ^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'to_image'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1687, in ask
    fig.show()
    ^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'show'
```

In [22]:

```
question = """
    Get the total number of invoices for each customer
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
5, updating n_results = 5
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': 'How many custom

ers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM cust
omers'}, {'role': 'user', 'content': 'what are the top 5 countries that cust
omers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT
(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY Cust
omerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all a
lbums and their corresponding artist names \n'}, {'role': 'assistant', 'con
tent': 'SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists A
S ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   Fin
d all tracks with a name containing "What" (case-insensitive)\n'}, {'role':
'assistant', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE
'%what%'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLi
te database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM s
qlite_master WHERE type='table'"}, {'role': 'user', 'content': ' \n   Get
the total number of invoices for each customer\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK
_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"inv
oice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NUL
L,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NU
LL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n
\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) RE
FERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)
\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AU
TOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastNa
me NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARC
HAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country
NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId I
NTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (Employee
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK
_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"empl
oyees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NU
LL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DA
TETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City N
VARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2
4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e
mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\nCREAT

E TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customers"}, {"role": "user", "content": "what are the top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   Get the total number of invoices for each customer\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:44:56.028585994Z', 'message': {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId; \n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_duration': 62441263807, 'load_duration': 22136263, 'prompt_eval_count': 1287, 'prompt_eval_duration': 56465602000, 'eval_count': 23, 'eval_duration': 5311231000}
LLM Response: SELECT CustomerId, COUNT(*) AS TotalInvoices
FROM invoices
GROUP BY CustomerId;

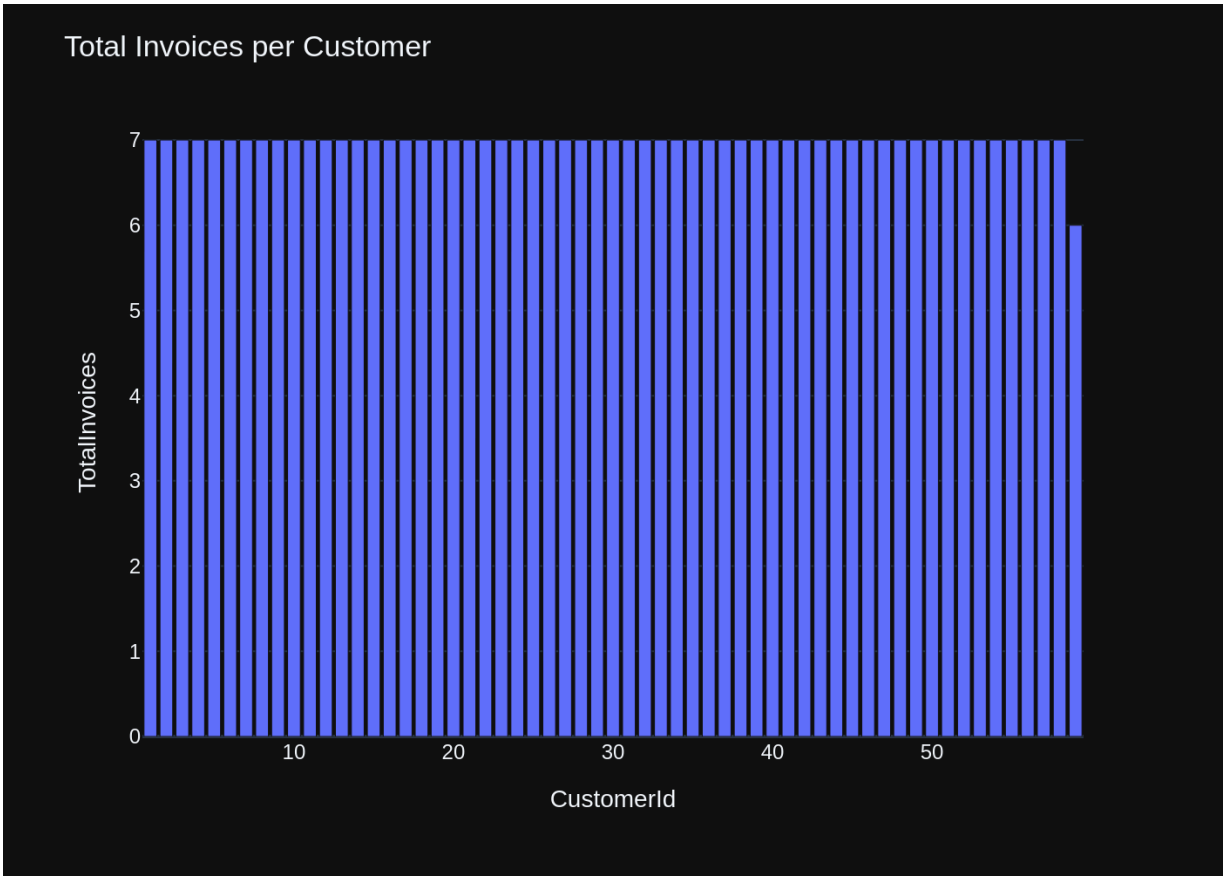
Info: Output from LLM: SELECT CustomerId, COUNT(*) AS TotalInvoices
FROM invoices
GROUP BY CustomerId;

Extracted SQL: SELECT CustomerId, COUNT(*) AS TotalInvoices
FROM invoices
GROUP BY CustomerId
SELECT CustomerId, COUNT(*) AS TotalInvoices

```
FROM invoices
GROUP BY CustomerId
```

	CustomerId	TotalInvoices
0	1	7
1	2	7
2	3	7
3	4	7
4	5	7
5	6	7
6	7	7
7	8	7
8	9	7
9	10	7
10	11	7
11	12	7
12	13	7
13	14	7
14	15	7
15	16	7
16	17	7
17	18	7
18	19	7
19	20	7
20	21	7
21	22	7
22	23	7
23	24	7
24	25	7
25	26	7
26	27	7
27	28	7
28	29	7
29	30	7
30	31	7
31	32	7
32	33	7
33	34	7
34	35	7
35	36	7
36	37	7
37	38	7
38	39	7
39	40	7
40	41	7
41	42	7
42	43	7
43	44	7
44	45	7
45	46	7
46	47	7
47	48	7
48	49	7
49	50	7
50	51	7
51	52	7
52	53	7

```
53          54              7
54          55              7
55          56              7
56          57              7
57          58              7
58          59              6
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Get the total number of invoices for each customer\n'\n\nThe DataFrame
was produced using this query: SELECT CustomerId, COUNT(*) AS TotalInvoices
\r\nFROM invoices\r\nGROUP BY CustomerId\n\nThe following is information abo
ut the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Custome
rId       int64\nTotalInvoices    int64\ndtype: object"}, {"role": "user",
"content": "Can you generate the Python plotly code to chart the results of
the dataframe? Assume the data is in a pandas dataframe called 'df'. If ther
e is only one value in the dataframe, use an Indicator. Respond with only Py
thon code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:45:23.037365275Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df[\'TotalInvoic
es\'].iloc[0],\n    title="Total Invoices",\n  )\nelse:\n  fig = px.bar(df,
x=\'CustomerId\', y=\'TotalInvoices\')\n  fig.update_layout(title="Total Inv
oices per Customer")\n```'}, 'done_reason': 'stop', 'done': True, 'total_dur
ation': 26983254725, 'load_duration': 21100636, 'prompt_eval_count': 170, 'p
rompt_eval_duration': 6541583000, 'eval_count': 92, 'eval_duration': 2037593
0000}
```


Total Invoices per Customer
TotalInvoices
0
1
2
3
4
5
6
7
10
20
30
40
50
CustomerId

Out[22]: ('SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId',

	CustomerId	TotalInvoices
0	1	7
1	2	7
2	3	7
3	4	7
4	5	7
5	6	7
6	7	7
7	8	7
8	9	7
9	10	7
10	11	7
11	12	7
12	13	7
13	14	7
14	15	7
15	16	7
16	17	7
17	18	7
18	19	7
19	20	7
20	21	7
21	22	7
22	23	7
23	24	7
24	25	7
25	26	7
26	27	7
27	28	7
28	29	7
29	30	7
30	31	7
31	32	7
32	33	7
33	34	7
34	35	7
35	36	7
36	37	7
37	38	7
38	39	7
39	40	7
40	41	7
41	42	7
42	43	7
43	44	7
44	45	7
45	46	7
46	47	7
47	48	7
48	49	7
49	50	7
50	51	7
51	52	7
52	53	7

```
 53          54             7
 54          55             7
 55          56             7
 56          57             7
 57          58             7
 58          59             6,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>TotalInvoices=%{y}<extr
a></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12,
13, 14, 15, 16, 17, 18,
                           19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,
31, 32, 33, 34, 35, 36,
                           37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48,
49, 50, 51, 52, 53, 54,
                           55, 56, 57, 58, 59]),
               'xaxis': 'x',
               'y': array([7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,
7, 7, 7, 7, 7, 7, 7, 7,
                           7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,
7, 7, 7, 7, 7, 7, 7, 7,
                           7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 6]),
               'yaxis': 'y'}],
      'layout': {'barmode': 'relative',
                 'legend': {'tracegroupgap': 0},
                 'margin': {'t': 60},
                 'template': '...',
                 'title': {'text': 'Total Invoices per Customer'},
                 'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'CustomerId'}},
                 'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'TotalInvoices'}}}
 }))
```

In [23]:

```
question = """
    Find the total number of invoices per country:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
6, updating n_results = 6
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId I
NTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity
INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (Inv
oiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (Custom
erId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceI
d)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCR
EATE TABLE "employees"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARC
HAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n
BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r
\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR
(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax
NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFE
RENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n
LastName NVARCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address
NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Co
untry NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    Sup
portRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees"
(EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NUL
L,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n
TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(20
0)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NUL
L,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Millisecond
s INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  N
OT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\t
ON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERE
NCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReport
sTo ON "employees" (ReportsTo)\n\n\n===Additional Context \n\nIn the chinook
database invoice means order\n\n===Response Guidelines \n1. If the provided
context is sufficient, please generate a valid SQL query without any explana
tions for the question. \n2. If the provided context is almost sufficient bu
t requires knowledge of a specific string in a particular column, please gen
erate an intermediate SQL query to find the distinct strings in that column.
Prepend the query with a comment saying intermediate_sql \n3. If the provide
d context is insufficient, please explain why it can\'t be generated. \n4. P
```

lease use the most relevant table(s). \n5. If the question has been asked an
d answered before, please repeat the answer exactly as it was given before.
\n'}, {'role': 'user', 'content': ' \n   Get the total number of invoices
for each customer\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId,
COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'rol
e': 'user', 'content': 'what are the top 5 countries that customers come fro
m?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS Customer
Count \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC
\r\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'},
{'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role':
'user', 'content': ' \n   List all albums and their corresponding artist n
ames \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.
Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'},
{'role': 'user', 'content': ' \n   Find all tracks with a name containing
"What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \r
\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'conten
t': 'Can you list all tables in the SQLite database catalog?'}, {'role': 'as
sistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"},
{'role': 'user', 'content': ' \n   Find the total number of invoices per c
ountry:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerI
d)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCR
EATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVA
RCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r
\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR
(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NV
ARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo)
REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n
Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r

\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n
SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT
NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r
\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name N
VARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER
NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Mill
iseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(1
0,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreI
d) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTy
peId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX I
FK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\n\n===Additional Context
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n
1. If the provided context is sufficient, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is a
lmost sufficient but requires knowledge of a specific string in a particular
column, please generate an intermediate SQL query to find the distinct strin
gs in that column. Prepend the query with a comment saying intermediate_sql
\n3. If the provided context is insufficient, please explain why it can't be
generated. \n4. Please use the most relevant table(s). \n5. If the question
has been asked and answered before, please repeat the answer exactly as it w
as given before. \n"}, {"role": "user", "content": " \n   Get the total nu
mber of invoices for each customer\n"}, {"role": "assistant", "content": "SE
LECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY Cust
omerId"}, {"role": "user", "content": "what are the top 5 countries that cus
tomers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT
(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY Cust
omerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": "How many customers
are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customer
s"}, {"role": "user", "content": " \n   List all albums and their correspo
nding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.T
itle,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.
ArtistId"}, {"role": "user", "content": " \n   Find all tracks with a name
containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content":
"SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "us
er", "content": "Can you list all tables in the SQLite database catalog?"},
{"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type
='table'"}, {"role": "user", "content": " \n   Find the total number of in
voices per country:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:46:36.274099532Z',
'message': {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.Invoi
ceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerI
d = i.CustomerId\r\nGROUP BY c.Country; \n\n\n\n'}, 'done_reason': 'stop',
'done': True, 'total_duration': 73115055230, 'load_duration': 20392284, 'pro
mpt_eval_count': 1383, 'prompt_eval_duration': 61836324000, 'eval_count': 4
4, 'eval_duration': 10506086000}
LLM Response: SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId

```
GROUP BY c.Country;

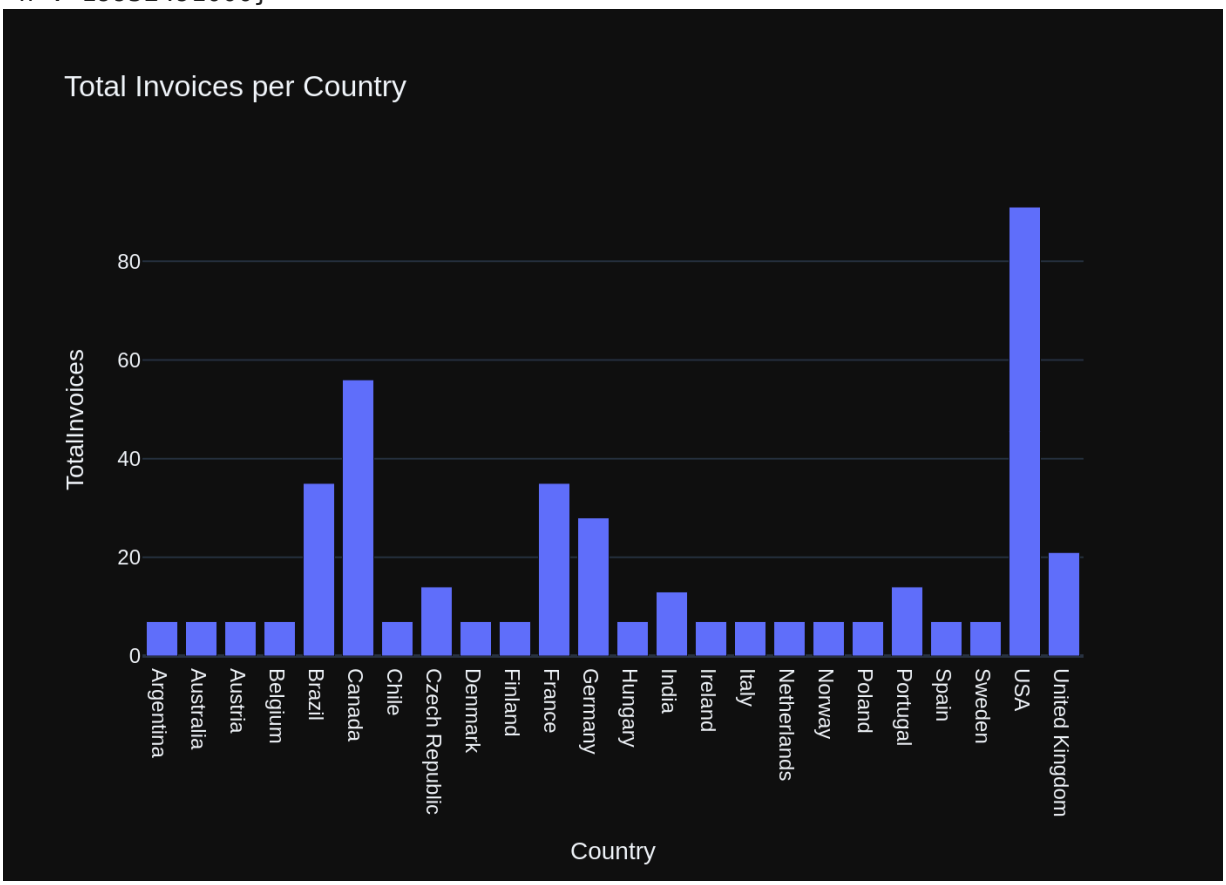


Info: Output from LLM: SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.Country;



Extracted SQL: SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.Country
SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.Country
          Country  TotalInvoices
0       Argentina             7
1       Australia             7
2         Austria             7
3         Belgium             7
4          Brazil            35
5          Canada            56
6           Chile             7
7   Czech Republic            14
8         Denmark             7
9         Finland             7
10         France            35
11        Germany            28
12        Hungary             7
13          India            13
14        Ireland             7
15          Italy             7
16    Netherlands             7
17         Norway             7
18         Poland             7
19       Portugal            14
20          Spain             7
21         Sweden             7
22            USA            91
23  United Kingdom            21
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find the total number of invoices per country:\n'\n\nThe DataFrame was
produced using this query: SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvo
ices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r
```

\nGROUP BY c.Country\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Country object\nTotalInvoices int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the results of the dataframe? Assume the data is in a pandas dataframe called 'df'. If there is only one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:47:03.350328991Z', 'message': {'role': 'assistant', 'content': '```python\nimport plotly.express as px\n\nif df.shape[0] == 1:\n fig = px.indicator(df, name="TotalInvoices", value=\'TotalInvoices\', title= "Total Invoices per Country")\nelse:\n fig = px.bar(df, x=\'Country\', y=\'TotalInvoices\', title= "Total Invoices per Country")\n \nfig.show() \n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 27056402735, 'load_duration': 21108416, 'prompt_eval_count': 188, 'prompt_eval_duration': 7034514000, 'eval_count': 92, 'eval_duration': 19952491000}



Out[23]:
```
('SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c
\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country',
            Country  TotalInvoices
 0         Argentina             7
 1         Australia             7
 2           Austria             7
 3           Belgium             7
 4            Brazil            35
 5            Canada            56
 6             Chile             7
 7    Czech Republic            14
 8           Denmark             7
 9           Finland             7
 10           France            35
 11          Germany            28
 12          Hungary             7
 13            India            13
 14          Ireland             7
 15            Italy             7
 16      Netherlands             7
 17           Norway             7
 18           Poland             7
 19         Portugal            14
 20            Spain             7
 21           Sweden             7
 22              USA            91
 23   United Kingdom            21,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'Country=%{x}<br>TotalInvoices=%{y}<extra>
</extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Argentina', 'Australia', 'Austria', 'Belgium',
'Brazil', 'Canada',
                           'Chile', 'Czech Republic', 'Denmark', 'Finland',
'France', 'Germany',
                           'Hungary', 'India', 'Ireland', 'Italy', 'Netherl
ands', 'Norway',
                           'Poland', 'Portugal', 'Spain', 'Sweden', 'USA',
'United Kingdom'],
                          dtype=object),
               'xaxis': 'x',
               'y': array([ 7,  7,  7,  7, 35, 56,  7, 14,  7,  7, 35, 28,
7, 13,  7,  7,  7,  7,
                            7, 14,  7,  7, 91, 21]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
```

```
                'title': {'text': 'Total Invoices per Country'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Country'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'TotalInvoices'}}}
 }))
```

In [24]:
```
question = """
    List all invoices with a total exceeding $10:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
7, updating n_results = 7
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   Get the total n

umber of invoices for each customer\n'}, {'role': 'assistant', 'content': 'S
ELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY Cus
tomerId'}, {'role': 'user', 'content': ' \n   Find the total number of inv
oices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country,
COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i O
N c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'con
tent': 'How many customers are there'}, {'role': 'assistant', 'content': 'SE
LECT COUNT(*) FROM customers'}, {'role': 'user', 'content': 'what are the to
p 5 countries that customers come from?'}, {'role': 'assistant', 'content':
'SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Co
untry\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'conten
t': ' \n   List all albums and their corresponding artist names \n'}, {'r
ole': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.Name\nFROM albu
ms AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user',
'content': ' \n   Find all tracks with a name containing "What" (case-inse
nsitive)\n'}, {'role': 'assistant', 'content': "SELECT * \r\nFROM tracks\r\n
WHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': 'Can you list
all tables in the SQLite database catalog?'}, {'role': 'assistant', 'conten
t': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user',
'content': ' \n   List all invoices with a total exceeding $10:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NU
LL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NU
LL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERE
NCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON D
ELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvo
iceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n
InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTE
GER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress
NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR
(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR
(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId)
REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\"
(TrackId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)
\n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGE
R,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Comp
oser NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTE
GER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId)
REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFER
ENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsT
o)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   Last

Name NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVA
RCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Count
ry NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(2
4),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   Sup
portRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees
\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREA
TE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARC
HAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n
BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r
\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR
(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax
NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFE
RENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO A
CTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (Suppor
tRepId)\n\n\n===Additional Context \n\nIn the chinook database invoice means
order\n\n===Response Guidelines \n1. If the provided context is sufficient,
please generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can't be generated. \n4. Please use the most relev
ant table(s). \n5. If the question has been asked and answered before, pleas
e repeat the answer exactly as it was given before. \n"}, {"role": "user",
"content": " \n   Get the total number of invoices for each customer\n"},
{"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoic
es\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": "
\n   Find the total number of invoices per country:\n"}, {"role": "assistan
t", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFRO
M customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY
c.Country"}, {"role": "user", "content": "How many customers are there"},
{"role": "assistant", "content": "SELECT COUNT(*) FROM customers"}, {"role":
"user", "content": "what are the top 5 countries that customers come fro
m?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) AS Customer
Count \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC
\r\nLIMIT 5"}, {"role": "user", "content": " \n   List all albums and thei
r corresponding artist names \n"}, {"role": "assistant", "content": "SELECT
\n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.Artis
tId = ar.ArtistId"}, {"role": "user", "content": " \n   Find all tracks wi
th a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant",
"content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"},
{"role": "user", "content": "Can you list all tables in the SQLite database
catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master
WHERE type='table'"}, {"role": "user", "content": " \n   List all invoices
with a total exceeding $10:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:48:07.452444868Z',
'message': {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nW
HERE Total > 10;\n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_dura
tion': 63988670940, 'load_duration': 21220382, 'prompt_eval_count': 1394, 'p
rompt_eval_duration': 59181531000, 'eval_count': 17, 'eval_duration': 384702
0000}
LLM Response: SELECT *
FROM invoices
WHERE Total > 10;

```
Info: Output from LLM: SELECT *
FROM invoices
WHERE Total > 10;



Extracted SQL: SELECT *
FROM invoices
WHERE Total > 10
SELECT *
FROM invoices
WHERE Total > 10
    InvoiceId  CustomerId          InvoiceDate             BillingAddress  \
0           5          23  2009-01-11 00:00:00            69 Salem Street
1          12           2  2009-02-11 00:00:00    Theodor-Heuss-Straße 34
2          19          40  2009-03-14 00:00:00             8, Rue Hanovre
3          26          19  2009-04-14 00:00:00            1 Infinite Loop
4          33          57  2009-05-15 00:00:00            Calle Lira, 198
..        ...         ...                  ...                        ...
59        383          10  2013-08-12 00:00:00  Rua Dr. Falcão Filho, 155
60        390          48  2013-09-12 00:00:00      Lijnbaansgracht 120bg
61        397          27  2013-10-13 00:00:00            1033 N Park Ave
62        404           6  2013-11-13 00:00:00              Rilská 3174/6
63        411          44  2013-12-14 00:00:00            Porthaninkatu 9

   BillingCity BillingState  BillingCountry BillingPostalCode  Total
0       Boston           MA             USA              2113  13.86
1    Stuttgart         None         Germany             70174  13.86
2        Paris         None          France             75002  13.86
3    Cupertino           CA             USA             95014  13.86
4     Santiago         None           Chile              None  13.86
..         ...          ...             ...               ...    ...
59   São Paulo           SP          Brazil         01007-010  13.86
60   Amsterdam           VV     Netherlands              1016  13.86
61      Tucson           AZ             USA             85719  13.86
62      Prague         None  Czech Republic             14300  25.86
63    Helsinki         None         Finland             00530  13.86

[64 rows x 9 columns]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all invoices with a total exceeding $10:\n'\n\nThe DataFrame was
produced using this query: SELECT * \r\nFROM invoices\r\nWHERE Total > 10\n
\nThe following is information about the resulting pandas DataFrame 'df': \n
Running df.dtypes gives:\n InvoiceId              int64\nCustomerId
int64\nInvoiceDate           object\nBillingAddress        object\nBillingCi
```

```
ty           object\nBillingState          object\nBillingCountry        obj
ect\nBillingPostalCode     object\nTotal                float64\ndtype: obje
ct"}, {"role": "user", "content": "Can you generate the Python plotly code t
o chart the results of the dataframe? Assume the data is in a pandas datafra
me called 'df'. If there is only one value in the dataframe, use an Indicato
r. Respond with only Python code. Do not answer with any explanations -- jus
t the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:48:39.761153441Z',
'message': {'role': 'assistant', 'content': "```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n      value=df['Total'].il
oc[0],\n      title='Invoice Total',\n      mode='number',\n      color_disc
rete_sequence=['green']\n  )\nelse:\n  px.scatter(df, x='InvoiceId', y='Tota
l', \n              size='Total', color='Total',\n              hover_name='In
voiceId', title='Invoice Totals') \n```"}, 'done_reason': 'stop', 'done': Tr
ue, 'total_duration': 32277699794, 'load_duration': 30346332, 'prompt_eval_c
ount': 202, 'prompt_eval_duration': 7389000000, 'eval_count': 114, 'eval_dur
ation': 24728002000}
Couldn't run plotly code:  'NoneType' object has no attribute 'show'
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1684, in ask
    img_bytes = fig.to_image(format="png", scale=2)
                ^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'to_image'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1687, in ask
    fig.show()
    ^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'show'
```

In [25]:
```
question = """
    Find all invoices since 2010 and the total amount invoiced:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
8, updating n_results = 8
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the

provided context is almost sufficient but requires knowledge of a specific s
tring in a particular column, please generate an intermediate SQL query to f
ind the distinct strings in that column. Prepend the query with a comment sa
ying intermediate_sql \n3. If the provided context is insufficient, please e
xplain why it can\'t be generated. \n4. Please use the most relevant table
(s). \n5. If the question has been asked and answered before, please repeat
the answer exactly as it was given before. \n'}, {'role': 'user', 'content':
' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assist
ant', 'content': 'SELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role':
'user', 'content': ' \n   Find the total number of invoices per countr
y:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceI
d) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId =
i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': ' \n   G
et the total number of invoices for each customer\n'}, {'role': 'assistant',
'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r
\nGROUP BY CustomerId'}, {'role': 'user', 'content': 'How many customers are
there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'},
{'role': 'user', 'content': 'what are the top 5 countries that customers com
e from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS Cus
tomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount
DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all albums and
their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SE
LECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.
ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   Find all trac
ks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistan
t', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%wha
t%'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLite da
tabase catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite
_master WHERE type='table'"}, {'role': 'user', 'content': ' \n   Find all
invoices since 2010 and the total amount invoiced:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (I
nvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCR
EATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVA

RCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r
\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR
(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NV
ARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo)
REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n
Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r
\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n
SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n
MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARC
HAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOI
NCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INT
EGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (Artist
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"p
laylist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INT
EGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId,
TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (Playlist
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (Tra
ckId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoic
e means order\n\n===Response Guidelines \n1. If the provided context is suff
icient, please generate a valid SQL query without any explanations for the q
uestion. \n2. If the provided context is almost sufficient but requires know
ledge of a specific string in a particular column, please generate an interm
ediate SQL query to find the distinct strings in that column. Prepend the qu
ery with a comment saying intermediate_sql \n3. If the provided context is i
nsufficient, please explain why it can't be generated. \n4. Please use the m
ost relevant table(s). \n5. If the question has been asked and answered befo
re, please repeat the answer exactly as it was given before. \n"}, {"role":
"user", "content": " \n   List all invoices with a total exceeding $1
0:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM invoices\r\nWHER
E Total > 10"}, {"role": "user", "content": " \n   Find the total number o
f invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Coun
try, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoice
s i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country"}, {"role": "user",
"content": " \n   Get the total number of invoices for each customer\n"},
{"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoic
es\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": "H
ow many customers are there"}, {"role": "assistant", "content": "SELECT COUN
T(*) FROM customers"}, {"role": "user", "content": "what are the top 5 count
ries that customers come from?"}, {"role": "assistant", "content": "SELECT C
ountry, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\n
ORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n

```
List all albums and their corresponding artist names  \n"}, {"role": "assist
ant", "content": "SELECT \n    a.Title,\n    ar.Name\nFROM albums AS a\nJOIN
artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "
\n    Find all tracks with a name containing \"What\" (case-insensitive)
\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LO
WER(Name) LIKE '%what%'"}, {"role": "user", "content": "Can you list all tab
les in the SQLite database catalog?"}, {"role": "assistant", "content": "SEL
ECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "conten
t": "  \n    Find all invoices since 2010 and the total amount invoice
d:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:49:57.655386885Z',
'message': {'role': 'assistant', 'content': "SELECT SUM(Total) AS TotalInvoi
ced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01';\n\n\n\n"}, 'done
_reason': 'stop', 'done': True, 'total_duration': 77831418315, 'load_duratio
n': 23050174, 'prompt_eval_count': 1579, 'prompt_eval_duration': 6897769700
0, 'eval_count': 33, 'eval_duration': 7768587000}
LLM Response: SELECT SUM(Total) AS TotalInvoiced
FROM invoices
WHERE InvoiceDate >= '2010-01-01';




Info: Output from LLM: SELECT SUM(Total) AS TotalInvoiced
FROM invoices
WHERE InvoiceDate >= '2010-01-01';




Extracted SQL: SELECT SUM(Total) AS TotalInvoiced
FROM invoices
WHERE InvoiceDate >= '2010-01-01'
SELECT SUM(Total) AS TotalInvoiced
FROM invoices
WHERE InvoiceDate >= '2010-01-01'
   TotalInvoiced
0        1879.14
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find all invoices since 2010 and the total amount invoiced:\n'\n\nThe
DataFrame was produced using this query: SELECT SUM(Total) AS TotalInvoiced
\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'\n\nThe following is
information about the resulting pandas DataFrame 'df': \nRunning df.dtypes g
ives:\n TotalInvoiced    float64\ndtype: object"}, {"role": "user", "conten
t": "Can you generate the Python plotly code to chart the results of the dat
aframe? Assume the data is in a pandas dataframe called 'df'. If there is on
ly one value in the dataframe, use an Indicator. Respond with only Python co
de. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
```

{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:50:29.544861332Z',
'message': {'role': 'assistant', 'content': "```python\nimport plotly.graph_
objects as go\n\nif df.shape[0] == 1:\n fig = go.Figure(data=go.Indicator(v
alue=df['TotalInvoiced'].iloc[0],\n   mode
='number',\n   title={'text': 'Total Invoic
ed Since 2010'}))\nelse:\n fig = go.Figure(data=[go.Bar(x=['Total Invoice
d'], y=[df['TotalInvoiced'].iloc[0]])])\n```"}, 'done_reason': 'stop', 'don
e': True, 'total_duration': 31865680630, 'load_duration': 24738624, 'prompt_
eval_count': 183, 'prompt_eval_duration': 6910117000, 'eval_count': 114, 'ev
al_duration': 24882314000}

Out[25]:
```
("SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate
>= '2010-01-01'",
    TotalInvoiced
 0        1879.14,
 Figure({
     'data': [{'mode': 'number', 'title': {'text': 'Total Invoiced Since 20
10'}, 'type': 'indicator', 'value': 1879.14}],
     'layout': {'template': '...'}
 }))
```

In [26]:
```
question = """
    List all employees and their reporting manager's name (if any):
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
9, updating n_results = 9
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

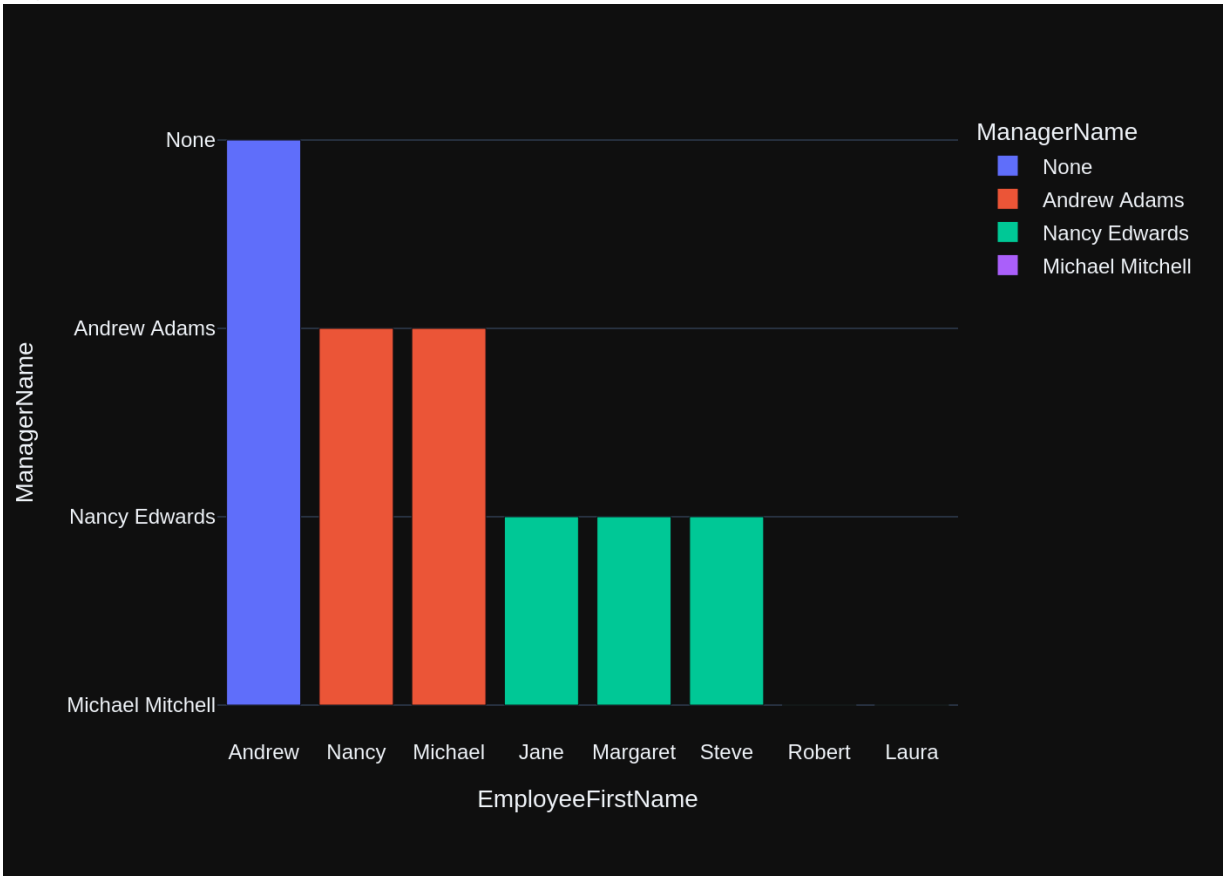
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "artists"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t

be generated. \n4. Please use the most relevant table(s). \n5. If the questi
on has been asked and answered before, please repeat the answer exactly as i
t was given before. \n'}, {'role': 'user', 'content': 'what are the top 5 co
untries that customers come from?'}, {'role': 'assistant', 'content': 'SELEC
T Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country
\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': '
\n   Get the total number of invoices for each customer\n'}, {'role': 'assi
stant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM inv
oices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   Find al
l invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistan
t', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHER
E InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   Find t
he total number of invoices per country:\n'}, {'role': 'assistant', 'conten
t': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers
c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'},
{'role': 'user', 'content': ' \n   List all albums and their corresponding
artist names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Titl
e,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.Art
istId'}, {'role': 'user', 'content': ' \n   List all invoices with a total
exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invo
ices\r\nWHERE Total > 10'}, {'role': 'user', 'content': 'How many customers
are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customer
s'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite data
base catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_m
aster WHERE type='table'"}, {'role': 'user', 'content': ' \n   Find all tr
acks with a name containing "What" (case-insensitive)\n'}, {'role': 'assista
nt', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%wha
t%'"}, {'role': 'user', 'content': " \n   List all employees and their rep
orting manager's name (if any):\n"}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (Repo
rtsTo)\n\nCREATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   F
irstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo
INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address
NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Co
untry NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(2
4),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY
(ReportsTo) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   Custome
rId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(4
0) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHA
R(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State N
VARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r
\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60)
NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REF
ERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (Suppo
rtRepId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY

KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   Invo
iceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   Billi
ngCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry
NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(1
0,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (Cu
stomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TAB
LE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCRE
MENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER N
OT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE
\"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaType
Id INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(22
0),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   Unit
Price NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"alb
ums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FORE
IGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_t
ypes\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER
NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_st
at1(tbl,idx,stat)\n\n\n===Additional Context \n\nIn the chinook database inv
oice means order\n\n===Response Guidelines \n1. If the provided context is s
ufficient, please generate a valid SQL query without any explanations for th
e question. \n2. If the provided context is almost sufficient but requires k
nowledge of a specific string in a particular column, please generate an int
ermediate SQL query to find the distinct strings in that column. Prepend the
query with a comment saying intermediate_sql \n3. If the provided context is
insufficient, please explain why it can't be generated. \n4. Please use the
most relevant table(s). \n5. If the question has been asked and answered bef
ore, please repeat the answer exactly as it was given before. \n"}, {"role":
"user", "content": "what are the top 5 countries that customers come fro
m?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) AS Customer
Count \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC
\r\nLIMIT 5"}, {"role": "user", "content": " \n   Get the total number of
invoices for each customer\n"}, {"role": "assistant", "content": "SELECT Cus
tomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerI
d"}, {"role": "user", "content": " \n   Find all invoices since 2010 and t
he total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM
(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-
01'"}, {"role": "user", "content": " \n   Find the total number of invoice
s per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUN
T(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.
CustomerId = i.CustomerId\r\nGROUP BY c.Country"}, {"role": "user", "conten
t": " \n   List all albums and their corresponding artist names \n"}, {"r
ole": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albu
ms AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user",
"content": " \n   List all invoices with a total exceeding $10:\n"}, {"rol
e": "assistant", "content": "SELECT * \r\nFROM invoices\r\nWHERE Total > 1
0"}, {"role": "user", "content": "How many customers are there"}, {"role":

"assistant", "content": "SELECT COUNT(*) FROM customers"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": " \n   List all employees and their reporting manager's name (if any):\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:51:53.124651661Z', 'message': {'role': 'assistant', 'content': "SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e;"}, 'done_reason': 'stop', 'done': True, 'total_duration': 83503041243, 'load_duration': 18986747, 'prompt_eval_count': 1561, 'prompt_eval_duration': 67693667000, 'eval_count': 61, 'eval_duration': 14651173000}
LLM Response: SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName,
       CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName
FROM employees e;
Info: Output from LLM: SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName,
       CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName
FROM employees e;
Extracted SQL: SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName,
       CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName
FROM employees e
SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName,
       CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName
FROM employees e
  EmployeeFirstName EmployeeLastName   ManagerName
0   Andrew   Adams   None
1   Nancy   Edwards   Andrew Adams
2   Jane   Peacock   Nancy Edwards
3   Margaret   Park   Nancy Edwards
4   Steve   Johnson   Nancy Edwards
5   Michael   Mitchell   Andrew Adams
6   Robert   King Michael Mitchell
7   Laura   Callahan Michael Mitchell
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that contains the results of the query that answers the question the user asked: '

\n   List all employees and their reporting manager's name (if any):\n'\n\n
The DataFrame was produced using this query: SELECT e.FirstName AS EmployeeF
irstName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.ReportsTo IS
NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WH
ERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e\n\nThe
following is information about the resulting pandas DataFrame 'df': \nRunnin
g df.dtypes gives:\n EmployeeFirstName   object\nEmployeeLastName   objec
t\nManagerName   object\ndtype: object"}, {"role": "user", "content":
"Can you generate the Python plotly code to chart the results of the datafra
me? Assume the data is in a pandas dataframe called 'df'. If there is only o
ne value in the dataframe, use an Indicator. Respond with only Python code.
Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:52:18.89328394Z', 'm
essage': {'role': 'assistant', 'content': '```python\nimport plotly.express
as px\n\nif df.shape[0] == 1:\n fig = px.indicator(df, \n
name="Employee", \n   value="ManagerName")\nelse:\n fig =
px.bar(df, x="EmployeeFirstName", y="ManagerName", \n   color="M
anagerName") \n\n```'}, 'done_reason': 'stop', 'done': True, 'total_duratio
n': 25748876313, 'load_duration': 58508001, 'prompt_eval_count': 216, 'promp
t_eval_duration': 8063539000, 'eval_count': 81, 'eval_duration': 1758177700
0}



Out[26]:
```
("SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName,
\n       CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName |
| ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS Man
agerName\r\nFROM employees e",
   EmployeeFirstName EmployeeLastName       ManagerName
 0            Andrew            Adams              None
 1             Nancy          Edwards      Andrew Adams
 2              Jane          Peacock     Nancy Edwards
 3          Margaret             Park     Nancy Edwards
 4             Steve          Johnson     Nancy Edwards
 5           Michael         Mitchell      Andrew Adams
 6            Robert             King  Michael Mitchell
 7             Laura         Callahan  Michael Mitchell,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'ManagerName=%{y}<br>EmployeeFirstName=%{x}
<extra></extra>',
               'legendgroup': 'None',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': 'None',
               'offsetgroup': 'None',
               'orientation': 'v',
               'showlegend': True,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Andrew'], dtype=object),
               'xaxis': 'x',
               'y': array(['None'], dtype=object),
               'yaxis': 'y'},
              {'alignmentgroup': 'True',
               'hovertemplate': 'ManagerName=%{y}<br>EmployeeFirstName=%{x}
<extra></extra>',
               'legendgroup': 'Andrew Adams',
               'marker': {'color': '#EF553B', 'pattern': {'shape': ''}},
               'name': 'Andrew Adams',
               'offsetgroup': 'Andrew Adams',
               'orientation': 'v',
               'showlegend': True,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Nancy', 'Michael'], dtype=object),
               'xaxis': 'x',
               'y': array(['Andrew Adams', 'Andrew Adams'], dtype=object),
               'yaxis': 'y'},
              {'alignmentgroup': 'True',
               'hovertemplate': 'ManagerName=%{y}<br>EmployeeFirstName=%{x}
<extra></extra>',
               'legendgroup': 'Nancy Edwards',
               'marker': {'color': '#00cc96', 'pattern': {'shape': ''}},
               'name': 'Nancy Edwards',
               'offsetgroup': 'Nancy Edwards',
               'orientation': 'v',
               'showlegend': True,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Jane', 'Margaret', 'Steve'], dtype=object),
```

```
                'xaxis': 'x',
                'y': array(['Nancy Edwards', 'Nancy Edwards', 'Nancy Edward
s'], dtype=object),
                'yaxis': 'y'},
               {'alignmentgroup': 'True',
                'hovertemplate': 'ManagerName=%{y}<br>EmployeeFirstName=%{x}
<extra></extra>',
                'legendgroup': 'Michael Mitchell',
                'marker': {'color': '#ab63fa', 'pattern': {'shape': ''}},
                'name': 'Michael Mitchell',
                'offsetgroup': 'Michael Mitchell',
                'orientation': 'v',
                'showlegend': True,
                'textposition': 'auto',
                'type': 'bar',
                'x': array(['Robert', 'Laura'], dtype=object),
                'xaxis': 'x',
                'y': array(['Michael Mitchell', 'Michael Mitchell'], dtype=o
bject),
                'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'title': {'text': 'ManagerName'}, 'tracegroupga
p': 0},
                'margin': {'t': 60},
                'template': '...',
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'EmployeeFirstName'}},
                'yaxis': {'anchor': 'x',
                          'categoryarray': [Michael Mitchell, Nancy Edward
s, Andrew
                                            Adams, None],
                          'categoryorder': 'array',
                          'domain': [0.0, 1.0],
                          'title': {'text': 'ManagerName'}}}
 }))
```

In [27]:
```
question = """
    Get the average invoice total for each customer:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHER

E InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   List a
ll invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'conten
t': 'SELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'con
tent': 'How many customers are there'}, {'role': 'assistant', 'content': 'SE
LECT COUNT(*) FROM customers'}, {'role': 'user', 'content': 'what are the to
p 5 countries that customers come from?'}, {'role': 'assistant', 'content':
'SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Co
untry\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'conten
t': " \n   List all employees and their reporting manager's name (if an
y):\n"}, {'role': 'assistant', 'content': "SELECT e.FirstName AS EmployeeFir
stName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.ReportsTo IS NU
LL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHER
E EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e"}, {'rol
e': 'user', 'content': ' \n   List all albums and their corresponding arti
st names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n
ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'},
{'role': 'user', 'content': ' \n   Find all tracks with a name containing
"What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \r
\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'conten
t': 'Can you list all tables in the SQLite database catalog?'}, {'role': 'as
sistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"},
{'role': 'user', 'content': ' \n   Get the average invoice total for each
customer:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK
_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"inv
oice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NUL
L,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NU
LL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n
\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) RE
FERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)
\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE INDEX IFK_CustomerSupp
ortRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"customers\"\r\n
(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   First
Name NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n
Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(4
0),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCod
e NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   E
mail NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY
(SupportRepId) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO AC

TION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\nCREATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": " \n   Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n   List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customers"}, {"role": "user", "content": "what are the top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   List all employees and their reporting manager's name (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   Get the average invoice total for each customer:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:53:26.434932376Z', 'message': {'role': 'assistant', 'content': 'SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId; \n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_duration': 67425285207, 'load_durat

```
ion': 22370574, 'prompt_eval_count': 1412, 'prompt_eval_duration': 601466250
00, 'eval_count': 26, 'eval_duration': 6008293000}
LLM Response: SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal
FROM invoices
GROUP BY CustomerId;

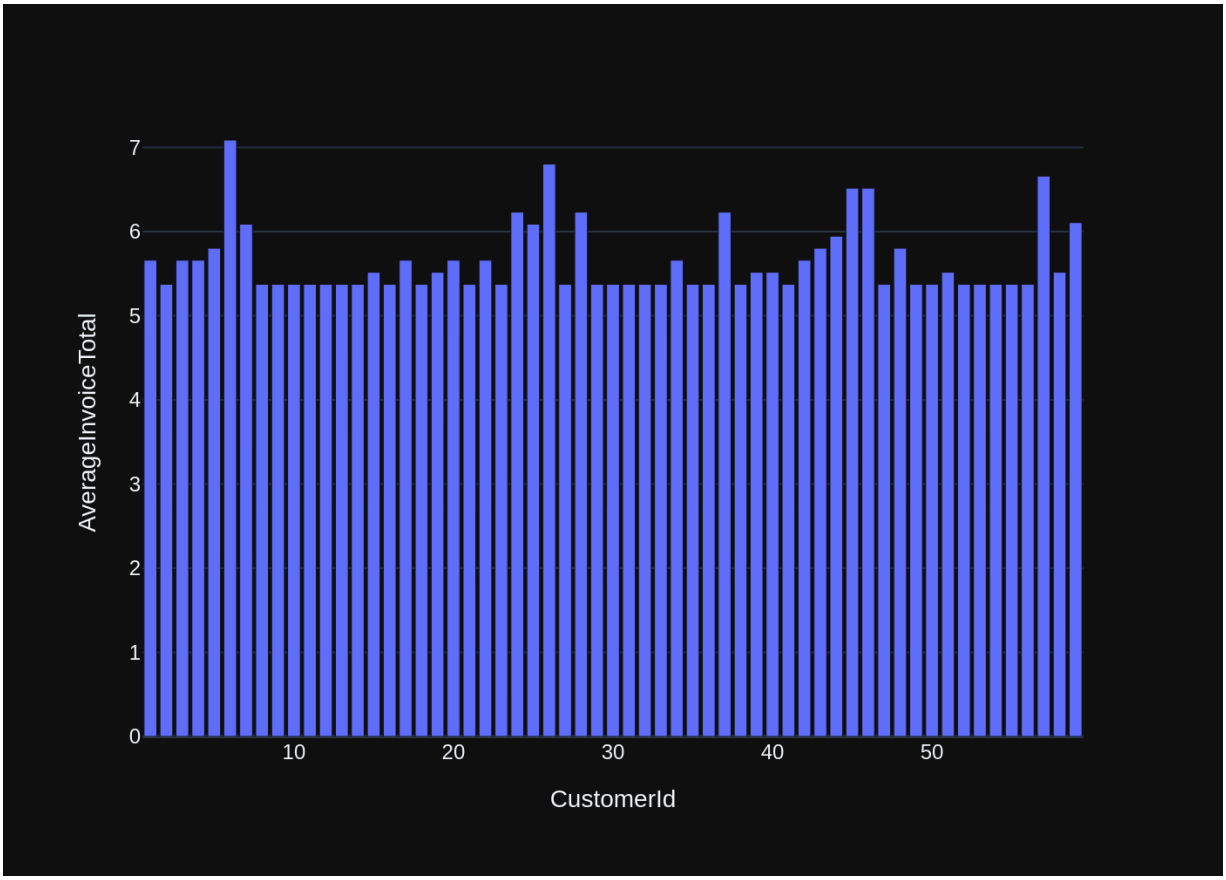


Info: Output from LLM: SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal
FROM invoices
GROUP BY CustomerId;



Extracted SQL: SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal
FROM invoices
GROUP BY CustomerId
SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal
FROM invoices
GROUP BY CustomerId
    CustomerId  AverageInvoiceTotal
0            1             5.660000
1            2             5.374286
2            3             5.660000
3            4             5.660000
4            5             5.802857
5            6             7.088571
6            7             6.088571
7            8             5.374286
8            9             5.374286
9           10             5.374286
10          11             5.374286
11          12             5.374286
12          13             5.374286
13          14             5.374286
14          15             5.517143
15          16             5.374286
16          17             5.660000
17          18             5.374286
18          19             5.517143
19          20             5.660000
20          21             5.374286
21          22             5.660000
22          23             5.374286
23          24             6.231429
24          25             6.088571
25          26             6.802857
26          27             5.374286
27          28             6.231429
28          29             5.374286
29          30             5.374286
30          31             5.374286
31          32             5.374286
32          33             5.374286
33          34             5.660000
34          35             5.374286
```

```
35          36             5.374286
36          37             6.231429
37          38             5.374286
38          39             5.517143
39          40             5.517143
40          41             5.374286
41          42             5.660000
42          43             5.802857
43          44             5.945714
44          45             6.517143
45          46             6.517143
46          47             5.374286
47          48             5.802857
48          49             5.374286
49          50             5.374286
50          51             5.517143
51          52             5.374286
52          53             5.374286
53          54             5.374286
54          55             5.374286
55          56             5.374286
56          57             6.660000
57          58             5.517143
58          59             6.106667
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Get the average invoice total for each customer:\n'\n\nThe DataFrame w
as produced using this query: SELECT CustomerId, AVG(Total) AS AverageInvoic
eTotal\r\nFROM invoices\r\nGROUP BY CustomerId\n\nThe following is informati
on about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n C
ustomerId               int64\nAverageInvoiceTotal    float64\ndtype: objec
t"}, {"role": "user", "content": "Can you generate the Python plotly code to
chart the results of the dataframe? Assume the data is in a pandas dataframe
called 'df'. If there is only one value in the dataframe, use an Indicator.
Respond with only Python code. Do not answer with any explanations -- just t
he code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:53:48.142612611Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  fig = px.indicator(df, name="AverageInvoi
ceTotal", value="AverageInvoiceTotal")\nelse:\n  fig = px.bar(df, x="Custome
rId", y="AverageInvoiceTotal") \n```'}, 'done_reason': 'stop', 'done': True,
'total_duration': 21687776892, 'load_duration': 60067142, 'prompt_eval_coun
t': 174, 'prompt_eval_duration': 6496595000, 'eval_count': 70, 'eval_duratio
n': 15086013000}
```


AverageInvoiceTotal
0
1
2
3
4
5
6
7
10
20
30
40
50
CustomerId

Out[27]: ('SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\n
GROUP BY CustomerId',

	CustomerId	AverageInvoiceTotal
0	1	5.660000
1	2	5.374286
2	3	5.660000
3	4	5.660000
4	5	5.802857
5	6	7.088571
6	7	6.088571
7	8	5.374286
8	9	5.374286
9	10	5.374286
10	11	5.374286
11	12	5.374286
12	13	5.374286
13	14	5.374286
14	15	5.517143
15	16	5.374286
16	17	5.660000
17	18	5.374286
18	19	5.517143
19	20	5.660000
20	21	5.374286
21	22	5.660000
22	23	5.374286
23	24	6.231429
24	25	6.088571
25	26	6.802857
26	27	5.374286
27	28	6.231429
28	29	5.374286
29	30	5.374286
30	31	5.374286
31	32	5.374286
32	33	5.374286
33	34	5.660000
34	35	5.374286
35	36	5.374286
36	37	6.231429
37	38	5.374286
38	39	5.517143
39	40	5.517143
40	41	5.374286
41	42	5.660000
42	43	5.802857
43	44	5.945714
44	45	6.517143
45	46	6.517143
46	47	5.374286
47	48	5.802857
48	49	5.374286
49	50	5.374286
50	51	5.517143
51	52	5.374286
52	53	5.374286

```
 53          54             5.374286
 54          55             5.374286
 55          56             5.374286
 56          57             6.660000
 57          58             5.517143
 58          59             6.106667,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>AverageInvoiceTotal=%
{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12,
13, 14, 15, 16, 17, 18,
                           19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,
31, 32, 33, 34, 35, 36,
                           37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48,
49, 50, 51, 52, 53, 54,
                           55, 56, 57, 58, 59]),
               'xaxis': 'x',
               'y': array([5.66      , 5.37428571, 5.66      , 5.66      ,
5.80285714, 7.08857143,
                           6.08857143, 5.37428571, 5.37428571, 5.37428571,
5.37428571, 5.37428571,
                           5.37428571, 5.37428571, 5.51714286, 5.37428571,
5.66      , 5.37428571,
                           5.51714286, 5.66      , 5.37428571, 5.66      ,
5.37428571, 6.23142857,
                           6.08857143, 6.80285714, 5.37428571, 6.23142857,
5.37428571, 5.37428571,
                           5.37428571, 5.37428571, 5.37428571, 5.66      ,
5.37428571, 5.37428571,
                           6.23142857, 5.37428571, 5.51714286, 5.51714286,
5.37428571, 5.66      ,
                           5.80285714, 5.94571429, 6.51714286, 6.51714286,
5.37428571, 5.80285714,
                           5.37428571, 5.37428571, 5.51714286, 5.37428571,
5.37428571, 5.37428571,
                           5.37428571, 5.37428571, 6.66      , 5.51714286,
6.10666667]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'CustomerId'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'AverageInvoiceTotal'}}}
 }))
```

In [28]:
```
question = """
    Find the top 5 most expensive tracks (based on unit price):
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n    TrackId INT
EGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NUL
L,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    Genr
eId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT
NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n
FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACT
ION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (G
enreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks"
(AlbumId)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE I
NDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE INDEX
IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE INDEX IFK_Trac
kMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE TABLE "invoice_items"\r\n
(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    In
voiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPric
e NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN
KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (Tra
ckId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE
"playlist_track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId IN
TEGER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistI
d, TrackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlists" (Playlis
tId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Tr
ackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE N
O ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCR
EATE TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NO
T NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT N
ULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nI
n the chinook database invoice means order\n\n===Response Guidelines \n1. If
the provided context is sufficient, please generate a valid SQL query withou
t any explanations for the question. \n2. If the provided context is almost
sufficient but requires knowledge of a specific string in a particular colum
n, please generate an intermediate SQL query to find the distinct strings in
that column. Prepend the query with a comment saying intermediate_sql \n3. I
f the provided context is insufficient, please explain why it can\'t be gene
rated. \n4. Please use the most relevant table(s). \n5. If the question has
been asked and answered before, please repeat the answer exactly as it was g
iven before. \n'}, {'role': 'user', 'content': '  \n    Find all tracks with
a name containing "What" (case-insensitive)\n'}, {'role': 'assistant', 'cont
ent': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {'rol
e': 'user', 'content': '  \n    List all invoices with a total exceeding $1
0:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHER
E Total > 10'}, {'role': 'user', 'content': '  \n    List all albums and the
ir corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELEC
T \n    a.Title,\n    ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.Art
istId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    Find all invoice
s since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'cont
ent': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE Invoice
Date >= '2010-01-01'"}, {'role': 'user', 'content': '  \n    Get the average
invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELE
CT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP B
```

Y CustomerId'}, {'role': 'user', 'content': 'what are the top 5 countries th
at customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country,
COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY
CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find t
he total number of invoices per country:\n'}, {'role': 'assistant', 'conten
t': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers
c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'},
{'role': 'user', 'content': 'Can you list all tables in the SQLite database
catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master
WHERE type='table'"}, {'role': 'user', 'content': ' \n   Get the total num
ber of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SEL
ECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY Custo
merId'}, {'role': 'user', 'content': 'How many customers are there'}, {'rol
e': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role': 'use
r', 'content': ' \n   Find the top 5 most expensive tracks (based on unit
price):\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (Albu
mId)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE INDE
X IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE INDEX I
FK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCREATE INDEX IFK_Tra
ckMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE TABLE \"invoice_items
\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPr
ice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREI
GN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO
ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks
\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   Tr
ackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (Pl
aylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\"
(PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN
KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON
UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (Artis
tId)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTO
INCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId IN
TEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (Artis
tId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional
Context \n\nIn the chinook database invoice means order\n\n===Response Guide
lines \n1. If the provided context is sufficient, please generate a valid SQ

L query without any explanations for the question. \n2. If the provided cont
ext is almost sufficient but requires knowledge of a specific string in a pa
rticular column, please generate an intermediate SQL query to find the disti
nct strings in that column. Prepend the query with a comment saying intermed
iate_sql \n3. If the provided context is insufficient, please explain why it
can't be generated. \n4. Please use the most relevant table(s). \n5. If the
question has been asked and answered before, please repeat the answer exactl
y as it was given before. \n"}, {"role": "user", "content": " \n   Find al
l tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "a
ssistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%
what%'"}, {"role": "user", "content": " \n   List all invoices with a tota
l exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM in
voices\r\nWHERE Total > 10"}, {"role": "user", "content": " \n   List all
albums and their corresponding artist names \n"}, {"role": "assistant", "co
ntent": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists
AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Fi
nd all invoices since 2010 and the total amount invoiced:\n"}, {"role": "ass
istant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r
\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n
Get the average invoice total for each customer:\n"}, {"role": "assistant",
"content": "SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM inv
oices\r\nGROUP BY CustomerId"}, {"role": "user", "content": "what are the to
p 5 countries that customers come from?"}, {"role": "assistant", "content":
"SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Co
untry\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "conten
t": " \n   Find the total number of invoices per country:\n"}, {"role": "a
ssistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGR
OUP BY c.Country"}, {"role": "user", "content": "Can you list all tables in
the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT nam
e FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n
Get the total number of invoices for each customer\n"}, {"role": "assistan
t", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoice
s\r\nGROUP BY CustomerId"}, {"role": "user", "content": "How many customers
are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customer
s"}, {"role": "user", "content": " \n   Find the top 5 most expensive trac
ks (based on unit price):\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:54:48.001329999Z',
'message': {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM
tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5;\n\n\n\n'}, 'done_reason': 'sto
p', 'done': True, 'total_duration': 59765313922, 'load_duration': 22342527,
'prompt_eval_count': 1264, 'prompt_eval_duration': 53060790000, 'eval_coun
t': 24, 'eval_duration': 5436377000}
LLM Response: SELECT Name, UnitPrice
FROM tracks
ORDER BY UnitPrice DESC
LIMIT 5;

Info: Output from LLM: SELECT Name, UnitPrice
FROM tracks
ORDER BY UnitPrice DESC
LIMIT 5;

```
Extracted SQL: SELECT Name, UnitPrice
FROM tracks
ORDER BY UnitPrice DESC
LIMIT 5
SELECT Name, UnitPrice
FROM tracks
ORDER BY UnitPrice DESC
LIMIT 5
                                    Name  UnitPrice
0  Battlestar Galactica: The Story So Far       1.99
1                  Occupation / Precipice       1.99
2                           Exodus, Pt. 1       1.99
3                           Exodus, Pt. 2       1.99
4                           Collaborators       1.99
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find the top 5 most expensive tracks (based on unit price):\n'\n\nThe
DataFrame was produced using this query: SELECT Name, UnitPrice\r\nFROM trac
ks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5\n\nThe following is information abo
ut the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Name
object\nUnitPrice    float64\ndtype: object"}, {"role": "user", "content":
"Can you generate the Python plotly code to chart the results of the datafra
me? Assume the data is in a pandas dataframe called 'df'. If there is only o
ne value in the dataframe, use an Indicator. Respond with only Python code.
Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:55:14.916472962Z',
'message': {'role': 'assistant', 'content': "```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df['UnitPrice'].
iloc[0],\n    name='Most Expensive Track Unit Price',\n    title='Top Tracks
by Unit Price'\n  )\nelse:\n  px.bar(df, x='Name', y='UnitPrice', title='Top
5 Most Expensive Tracks') \n```"}, 'done_reason': 'stop', 'done': True, 'tot
al_duration': 26888365288, 'load_duration': 17891522, 'prompt_eval_count': 1
73, 'prompt_eval_duration': 6454648000, 'eval_count': 94, 'eval_duration': 2
0369192000}
Couldn't run plotly code:  'NoneType' object has no attribute 'show'
```

```
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1684, in ask
    img_bytes = fig.to_image(format="png", scale=2)
                ^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'to_image'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1687, in ask
    fig.show()
    ^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'show'
```

In [29]:
```
question = """
    List all genres and the number of tracks in each genre:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE TABLE "genres"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "playlists"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC \r\nLIMIT 5'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Count

ry'}, {'role': 'user', 'content': ' \n   List all invoices with a total ex
ceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoic
es\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Find all invo
ices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'c
ontent': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE Invo
iceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   Get the tota
l number of invoices for each customer\n'}, {'role': 'assistant', 'content':
'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY C
ustomerId'}, {'role': 'user', 'content': 'How many customers are there'},
{'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role':
'user', 'content': ' \n   List all genres and the number of tracks in each
genre:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (Genr
eId)\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTO
INCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_Playl
istTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE INDEX IFK_TrackAlb
umId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   Playl
istId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)
\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCR
EATE TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n
TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY
(PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlist
s\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOR
EIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT N
ULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERE
NCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\n\n===Add
itional Context \n\nIn the chinook database invoice means order\n\n===Respon
se Guidelines \n1. If the provided context is sufficient, please generate a
valid SQL query without any explanations for the question. \n2. If the provi
ded context is almost sufficient but requires knowledge of a specific string
in a particular column, please generate an intermediate SQL query to find th
e distinct strings in that column. Prepend the query with a comment saying i
ntermediate_sql \n3. If the provided context is insufficient, please explain
why it can't be generated. \n4. Please use the most relevant table(s). \n5.
If the question has been asked and answered before, please repeat the answer
exactly as it was given before. \n"}, {"role": "user", "content": " \n   F

ind the top 5 most expensive tracks (based on unit price):\n"}, {"role": "as
sistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY Unit
Price DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   List all album
s and their corresponding artist names \n"}, {"role": "assistant", "conten
t": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS a
r ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Find a
ll tracks with a name containing \"What\" (case-insensitive)\n"}, {"role":
"assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE
'%what%'"}, {"role": "user", "content": "what are the top 5 countries that c
ustomers come from?"}, {"role": "assistant", "content": "SELECT Country, COU
NT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY Cu
stomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": "Can you list all
tables in the SQLite database catalog?"}, {"role": "assistant", "content":
"SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "cont
ent": " \n   Find the total number of invoices per country:\n"}, {"role":
"assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoic
es\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\n
GROUP BY c.Country"}, {"role": "user", "content": " \n   List all invoices
with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *
\r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user", "content": " \n
Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "a
ssistant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices
\r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n
Get the total number of invoices for each customer\n"}, {"role": "assistan
t", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoice
s\r\nGROUP BY CustomerId"}, {"role": "user", "content": "How many customers
are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customer
s"}, {"role": "user", "content": " \n   List all genres and the number of
tracks in each genre:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:56:15.985079001Z',
'message': {'role': 'assistant', 'content': 'SELECT g.Name, COUNT(t.GenreId)
AS TrackCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\n
GROUP BY g.Name;\n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_dura
tion': 61014714035, 'load_duration': 20493479, 'prompt_eval_count': 1188, 'p
rompt_eval_duration': 49319869000, 'eval_count': 45, 'eval_duration': 104451
70000}
LLM Response: SELECT g.Name, COUNT(t.GenreId) AS TrackCount
FROM genres g
JOIN tracks t ON g.GenreId = t.GenreId
GROUP BY g.Name;

Info: Output from LLM: SELECT g.Name, COUNT(t.GenreId) AS TrackCount
FROM genres g
JOIN tracks t ON g.GenreId = t.GenreId
GROUP BY g.Name;

Extracted SQL: SELECT g.Name, COUNT(t.GenreId) AS TrackCount
FROM genres g
JOIN tracks t ON g.GenreId = t.GenreId

```
GROUP BY g.Name
SELECT g.Name, COUNT(t.GenreId) AS TrackCount
FROM genres g
JOIN tracks t ON g.GenreId = t.GenreId
GROUP BY g.Name
                   Name  TrackCount
0           Alternative          40
1   Alternative & Punk         332
2                Blues          81
3            Bossa Nova          15
4             Classical          74
5                Comedy          17
6                 Drama          64
7       Easy Listening          24
8    Electronica/Dance          30
9           Heavy Metal          28
10          Hip Hop/Rap          35
11                 Jazz         130
12                Latin         579
13                Metal         374
14                Opera           1
15                  Pop          48
16            R&B/Soul          61
17              Reggae          58
18                Rock        1297
19       Rock And Roll          12
20    Sci Fi & Fantasy          26
21     Science Fiction          13
22          Soundtrack          43
23            TV Shows          93
24               World          28
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all genres and the number of tracks in each genre:\n'\n\nThe Data
Frame was produced using this query: SELECT g.Name, COUNT(t.GenreId) AS Trac
kCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP B
Y g.Name\n\nThe following is information about the resulting pandas DataFram
e 'df': \nRunning df.dtypes gives:\n Name          object\nTrackCount     in
t64\ndtype: object"}, {"role": "user", "content": "Can you generate the Pyth
on plotly code to chart the results of the dataframe? Assume the data is in
a pandas dataframe called 'df'. If there is only one value in the dataframe,
use an Indicator. Respond with only Python code. Do not answer with any expl
anations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:56:44.779674515Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    df,\n    name="TrackCo
unt",\n    value=\'TrackCount\',\n    title=f"Number of Tracks per Genre ({d
f[\'Name\'].iloc[0]})"\n  )\nelse:\n  px.bar(df, x="Name", y="TrackCount", t
itle="Tracks per Genre") \n```'}, 'done_reason': 'stop', 'done': True, 'tota
l_duration': 28768138600, 'load_duration': 23605651, 'prompt_eval_count': 19
```

```
3, 'prompt_eval_duration': 7278135000, 'eval_count': 99, 'eval_duration': 21
421473000}
Couldn't run plotly code:  'NoneType' object has no attribute 'show'
```

```
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1684, in ask
    img_bytes = fig.to_image(format="png", scale=2)
                ^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'to_image'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1687, in ask
    fig.show()
    ^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'show'
```

In [30]:
```
question = """
    Get all genres that do not have any tracks associated with them:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON "tracks" (Ge
nreId)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTO
INCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTE
GER,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Co
mposer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes IN
TEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumI
d) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO AC
TION,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFER
ENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO A
CTION\r\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (Tra
ckId)\n\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREA
TE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "genres"\r\n
(\r\n    GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVA
RCHAR(120)\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    Art
istId INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists"
(ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TA
BLE "playlist_track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackI
d INTEGER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (Playli
stId, TrackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlists" (Play
listId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n
\nCREATE TABLE "playlists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Additional Context
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n
1. If the provided context is sufficient, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is a
lmost sufficient but requires knowledge of a specific string in a particular
column, please generate an intermediate SQL query to find the distinct strin
gs in that column. Prepend the query with a comment saying intermediate_sql
\n3. If the provided context is insufficient, please explain why it can\'t b
e generated. \n4. Please use the most relevant table(s). \n5. If the questio
n has been asked and answered before, please repeat the answer exactly as it
was given before. \n'}, {'role': 'user', 'content': '  \n    List all genres
and the number of tracks in each genre:\n'}, {'role': 'assistant', 'conten
t': 'SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN
tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name'}, {'role': 'user', 'co
ntent': '  \n    Find all tracks with a name containing "What" (case-insensi
tive)\n'}, {'role': 'assistant', 'content': "SELECT * \r\nFROM tracks\r\nWHE
RE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': '  \n    Find th
e top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistan
t', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice
DESC\r\nLIMIT 5'}, {'role': 'user', 'content': '  \n    List all albums and
their corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SE
LECT \n    a.Title,\n    ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.
ArtistId = ar.ArtistId'}, {'role': 'user', 'content': 'Can you list all tabl
es in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELE
CT name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content':
"  \n    List all employees and their reporting manager's name (if an
y):\n"}, {'role': 'assistant', 'content': "SELECT e.FirstName AS EmployeeFir
stName, e.LastName AS EmployeeLastName, \n       CASE WHEN e.ReportsTo IS NU
```

LL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHER
E EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e"}, {'rol
e': 'user', 'content': ' \n   Find all invoices since 2010 and the total a
mount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Total) AS
TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {'ro
le': 'user', 'content': ' \n   List all invoices with a total exceeding $1
0:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHER
E Total > 10'}, {'role': 'user', 'content': 'what are the top 5 countries th
at customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country,
COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY
CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': 'How many custo
mers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM cus
tomers'}, {'role': 'user', 'content': ' \n   Get all genres that do not ha
ve any tracks associated with them:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\n
CREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n
MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARC
HAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)
\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE
INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"genres\"\r
\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name N
VARCHAR(120)\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRI
MARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n
ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artis
ts\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREA
TE TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n
TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY
(PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlist
s\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOR
EIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\"
(ArtistId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER PRIM
ARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Addit
ional Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can't be generated. \n4. Please use the most relevant table(s). \n5. If

the question has been asked and answered before, please repeat the answer ex
actly as it was given before. \n"}, {"role": "user", "content": " \n   Lis
t all genres and the number of tracks in each genre:\n"}, {"role": "assistan
t", "content": "SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres
g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name"}, {"role":
"user", "content": " \n   Find all tracks with a name containing \"What\"
(case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM
tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "
\n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"rol
e": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER
BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   List al
l albums and their corresponding artist names \n"}, {"role": "assistant",
"content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artis
ts AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "Can you
list all tables in the SQLite database catalog?"}, {"role": "assistant", "co
ntent": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "use
r", "content": " \n   List all employees and their reporting manager's nam
e (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstName AS Emp
loyeeFirstName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.Reports
To IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employ
ees WHERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees
e"}, {"role": "user", "content": " \n   Find all invoices since 2010 and t
he total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM
(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-
01'"}, {"role": "user", "content": " \n   List all invoices with a total e
xceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM invoi
ces\r\nWHERE Total > 10"}, {"role": "user", "content": "what are the top 5 c
ountries that customers come from?"}, {"role": "assistant", "content": "SELE
CT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country
\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": "H
ow many customers are there"}, {"role": "assistant", "content": "SELECT COUN
T(*) FROM customers"}, {"role": "user", "content": " \n   Get all genres t
hat do not have any tracks associated with them:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:57:46.259942483Z',
'message': {'role': 'assistant', 'content': 'SELECT * \r\nFROM genres g\r\nL
EFT JOIN tracks t ON g.GenreId = t.GenreId\r\nWHERE t.TrackId IS NULL;\n\n\n
\n\n'}, 'done_reason': 'stop', 'done': True, 'total_duration': 61426991862,
'load_duration': 19051354, 'prompt_eval_count': 1239, 'prompt_eval_duratio
n': 52134014000, 'eval_count': 35, 'eval_duration': 8041991000}
LLM Response: SELECT *
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
WHERE t.TrackId IS NULL;

Info: Output from LLM: SELECT *
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
WHERE t.TrackId IS NULL;

```
Extracted SQL: SELECT *
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
WHERE t.TrackId IS NULL
SELECT *
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
WHERE t.TrackId IS NULL
Empty DataFrame
Columns: [GenreId, Name, TrackId, Name, AlbumId, MediaTypeId, GenreId, Compo
ser, Milliseconds, Bytes, UnitPrice]
Index: []
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Get all genres that do not have any tracks associated with them:\n'\n
\nThe DataFrame was produced using this query: SELECT * \r\nFROM genres g\r
\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\r\nWHERE t.TrackId IS NULL\n\n
The following is information about the resulting pandas DataFrame 'df': \nRu
nning df.dtypes gives:\n GenreId         object\nName            object\nTra
ckId         object\nName            object\nAlbumId         object\nMediaTy
peId     object\nGenreId         object\nComposer        object\nMillisecond
s    object\nBytes           object\nUnitPrice       object\ndtype: objec
t"}, {"role": "user", "content": "Can you generate the Python plotly code to
chart the results of the dataframe? Assume the data is in a pandas dataframe
called 'df'. If there is only one value in the dataframe, use an Indicator.
Respond with only Python code. Do not answer with any explanations -- just t
he code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:58:10.829065073Z',
'message': {'role': 'assistant', 'content': "```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  fig = px.indicator(df, value='Name', titl
e='Genres with No Tracks')\nelse:\n  fig = px.bar(df, x='Name', y='GenreId',
title='Genres with No Tracks') \n```"}, 'done_reason': 'stop', 'done': True,
'total_duration': 24566561361, 'load_duration': 64425116, 'prompt_eval_coun
t': 222, 'prompt_eval_duration': 8315958000, 'eval_count': 75, 'eval_duratio
n': 16137255000}
Couldn't run plotly code:  The truth value of a Series is ambiguous. Use a.e
mpty, a.bool(), a.item(), a.any() or a.all().
```

```
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1999, in get_plotly_figure
    exec(plotly_code, globals(), ldict)
  File "<string>", line 6, in <module>
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/plotl
y/express/_chart_types.py", line 373, in bar
    return make_figure(
           ^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/plotl
y/express/_core.py", line 2090, in make_figure
    args = build_dataframe(args, constructor)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/plotl
y/express/_core.py", line 1492, in build_dataframe
    df_output, wide_id_vars = process_args_into_dataframe(
                              ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/plotl
y/express/_core.py", line 1228, in process_args_into_dataframe
    df_output[col_name] = to_unindexed_series(
                          ^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/plotl
y/express/_core.py", line 1076, in to_unindexed_series
    return pd.Series(x, name=name).reset_index(drop=True)
           ^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/panda
s/core/series.py", line 584, in __init__
    data = sanitize_array(data, index, dtype, copy)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/panda
s/core/construction.py", line 633, in sanitize_array
    return sanitize_array(
           ^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/panda
s/core/construction.py", line 659, in sanitize_array
    subarr = _sanitize_ndim(subarr, data, dtype, index, allow_2d=allow_2d)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/panda
s/core/construction.py", line 718, in _sanitize_ndim
    raise ValueError(
ValueError: Data must be 1-dimensional, got ndarray of shape (0, 2) instead

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1675, in ask
    fig = self.get_plotly_figure(plotly_code=plotly_code, df=df)
          ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 2016, in get_plotly_figure
    elif len(categorical_cols) >= 1 and df[categorical_cols[0]].nunique() <
10:
                                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^^
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/panda
```

```
s/core/generic.py", line 1577, in __nonzero__
    raise ValueError(
ValueError: The truth value of a Series is ambiguous. Use a.empty, a.bool(),
a.item(), a.any() or a.all().
```

In [31]:
```
question = """
    List all customers who have not placed any orders:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "playlists"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a val

id SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': " \n   List all employees and their reporting manager's name (if any):\n"}, {'role': 'assistant', 'content': "SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e"}, {'role': 'user', 'content': ' \n   Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all customers who have not placed any orders:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT

NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR
(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r
\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR
(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax
NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEG
ER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (EmployeeId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoi
ce_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r
\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NUL
L,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFE
RENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY K
EY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Fi
rstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo
INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address
NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Co
untry NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(2
4),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY
(ReportsTo) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n   Pl
aylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRA
INT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY
(PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\"
(TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TAB
LE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NUL
L,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupp
ortRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"playlists\"\r\n
(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name
NVARCHAR(120)\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PR
IMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n
AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices
\" (CustomerId)\n\n\n===Additional Context \n\nIn the chinook database invoi
ce means order\n\n===Response Guidelines \n1. If the provided context is suf
ficient, please generate a valid SQL query without any explanations for the
question. \n2. If the provided context is almost sufficient but requires kno
wledge of a specific string in a particular column, please generate an inter
mediate SQL query to find the distinct strings in that column. Prepend the q
uery with a comment saying intermediate_sql \n3. If the provided context is
insufficient, please explain why it can't be generated. \n4. Please use the
most relevant table(s). \n5. If the question has been asked and answered bef
ore, please repeat the answer exactly as it was given before. \n"}, {"role":
"user", "content": "How many customers are there"}, {"role": "assistant", "c
ontent": "SELECT COUNT(*) FROM customers"}, {"role": "user", "content": "wha
t are the top 5 countries that customers come from?"}, {"role": "assistant",

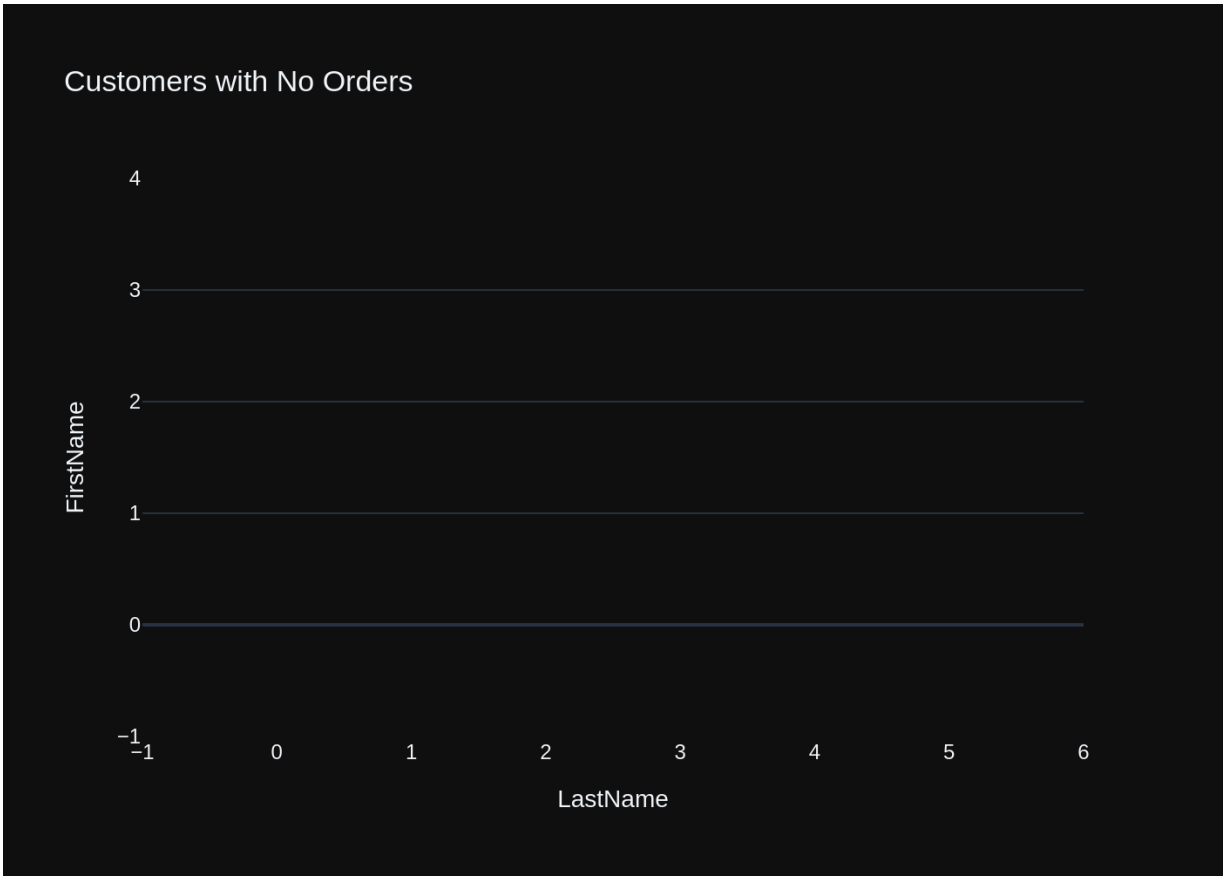
"content": "SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\n
GROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "use
r", "content": " \n   Get the total number of invoices for each customer
\n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS Total
Invoices\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "conten
t": " \n   Find the total number of invoices per country:\n"}, {"role": "a
ssistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGR
OUP BY c.Country"}, {"role": "user", "content": " \n   List all employees
and their reporting manager's name (if any):\n"}, {"role": "assistant", "con
tent": "SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastN
ame, \n   CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstNa
me || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS
ManagerName\r\nFROM employees e"}, {"role": "user", "content": " \n   Get
the average invoice total for each customer:\n"}, {"role": "assistant", "con
tent": "SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoice
s\r\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   List all in
voices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SE
LECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user", "content":
" \n   Find all invoices since 2010 and the total amount invoiced:\n"},
{"role": "assistant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM
invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "conten
t": " \n   List all albums and their corresponding artist names \n"}, {"r
ole": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albu
ms AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user",
"content": " \n   Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tr
acks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": "
\n   List all customers who have not placed any orders:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T18:59:35.734136301Z',
'message': {'role': 'assistant', 'content': 'SELECT * \r\nFROM customers\r\n
LEFT JOIN invoices ON customers.CustomerId = invoices.CustomerId\r\nWHERE In
voiceId IS NULL;\n\n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_du
ration': 84853475977, 'load_duration': 24125406, 'prompt_eval_count': 1748,
'prompt_eval_duration': 76677415000, 'eval_count': 29, 'eval_duration': 6856
619000}
LLM Response: SELECT *
FROM customers
LEFT JOIN invoices ON customers.CustomerId = invoices.CustomerId
WHERE InvoiceId IS NULL;

Info: Output from LLM: SELECT *
FROM customers
LEFT JOIN invoices ON customers.CustomerId = invoices.CustomerId
WHERE InvoiceId IS NULL;

Extracted SQL: SELECT *

```
FROM customers
LEFT JOIN invoices ON customers.CustomerId = invoices.CustomerId
WHERE InvoiceId IS NULL
SELECT *
FROM customers
LEFT JOIN invoices ON customers.CustomerId = invoices.CustomerId
WHERE InvoiceId IS NULL
Empty DataFrame
Columns: [CustomerId, FirstName, LastName, Company, Address, City, State, Co
untry, PostalCode, Phone, Fax, Email, SupportRepId, InvoiceId, CustomerId, I
nvoiceDate, BillingAddress, BillingCity, BillingState, BillingCountry, Billi
ngPostalCode, Total]
Index: []

[0 rows x 22 columns]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all customers who have not placed any orders:\n'\n\nThe DataFrame
was produced using this query: SELECT * \r\nFROM customers\r\nLEFT JOIN invo
ices ON customers.CustomerId = invoices.CustomerId\r\nWHERE InvoiceId IS NUL
L\n\nThe following is information about the resulting pandas DataFrame 'df':
\nRunning df.dtypes gives:\n CustomerId           object\nFirstName
object\nLastName             object\nCompany              object\nAddress
object\nCity                 object\nState                object\nCountry
object\nPostalCode           object\nPhone                object\nFax
object\nEmail                object\nSupportRepId         object\nInvoiceId
object\nCustomerId           object\nInvoiceDate          object\nBillingAdd
ress       object\nBillingCity          object\nBillingState         object
\nBillingCountry       object\nBillingPostalCode    object\nTotal
object\ndtype: object"}, {"role": "user", "content": "Can you generate the P
ython plotly code to chart the results of the dataframe? Assume the data is
in a pandas dataframe called 'df'. If there is only one value in the datafra
me, use an Indicator. Respond with only Python code. Do not answer with any
explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:00:11.202632726Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df.shape[0],\n
title="Customers with No Orders",\n    mode=\'number\',\n    color={\'value
\': \'lightgreen\'} if df.shape[0] > 0 else \'red\'\n  )\nelse:\n  fig = px.
bar(df, x=\'LastName\', y=\'FirstName\', title="Customers with No Orders")\n
fig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 35
464939073, 'load_duration': 61336950, 'prompt_eval_count': 262, 'prompt_eval
_duration': 10029894000, 'eval_count': 116, 'eval_duration': 25324844000}
```


Customers with No Orders
FirstName
LastName
4
3
2
1
0
−1
−1
0
1
2
3
4
5
6

Out[31]:
```
('SELECT * \r\nFROM customers\r\nLEFT JOIN invoices ON customers.CustomerId
= invoices.CustomerId\r\nWHERE InvoiceId IS NULL',
 Empty DataFrame
 Columns: [CustomerId, FirstName, LastName, Company, Address, City, State,
Country, PostalCode, Phone, Fax, Email, SupportRepId, InvoiceId, CustomerI
d, InvoiceDate, BillingAddress, BillingCity, BillingState, BillingCountry,
BillingPostalCode, Total]
 Index: []

 [0 rows x 22 columns],
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'LastName=%{x}<br>FirstName=%{y}<extra></ex
tra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([], dtype=object),
               'xaxis': 'x',
               'y': array([], dtype=object),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Customers with No Orders'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'LastName'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'FirstName'}}}
 }))
```

In [32]:
```
question = """
    There are 3 tables: artists, albums and tracks, where albums and artists
    Can you find the top 10 most popular artists based on the number of trac
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "artists"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "playlists"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   List all genres and the number of tracks in each genre:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT

c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN i
nvoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role':
'user', 'content': 'Can you list all tables in the SQLite database catalo
g?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE
type='table'"}, {'role': 'user', 'content': ' \n   List all invoices with
a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nF
ROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Get
the total number of invoices for each customer\n'}, {'role': 'assistant', 'c
ontent': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nG
ROUP BY CustomerId'}, {'role': 'user', 'content': 'How many customers are th
ere'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'},
{'role': 'user', 'content': ' \n   There are 3 tables: artists, albums and
tracks, where albums and artists are linked by ArtistId, albums and tracks a
re linked by AlbumId,\n   Can you find the top 10 most popular artists base
d on the number of tracks\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEG
ER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NUL
L,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENC
ES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUT
OINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_Albu
mArtistId ON \"albums\" (ArtistId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tra
cks\" (AlbumId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE
TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n
(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n
CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREI
GN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE
NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tra
cks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREA
TE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE INDEX IFK_Playli
stTrackTrackId ON \"playlist_track\" (TrackId)\n\n\n===Additional Context \n
\nIn the chinook database invoice means order\n\n===Response Guidelines \n1.
If the provided context is sufficient, please generate a valid SQL query wit
hout any explanations for the question. \n2. If the provided context is almo
st sufficient but requires knowledge of a specific string in a particular co
lumn, please generate an intermediate SQL query to find the distinct strings
in that column. Prepend the query with a comment saying intermediate_sql \n

3. If the provided context is insufficient, please explain why it can't be g
enerated. \n4. Please use the most relevant table(s). \n5. If the question h
as been asked and answered before, please repeat the answer exactly as it wa
s given before. \n"}, {"role": "user", "content": " \n   List all genres a
nd the number of tracks in each genre:\n"}, {"role": "assistant", "content":
"SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN trac
ks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name"}, {"role": "user", "conten
t": " \n   Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tr
acks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": "
\n   List all albums and their corresponding artist names \n"}, {"role":
"assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS
a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "conte
nt": "what are the top 5 countries that customers come from?"}, {"role": "as
sistant", "content": "SELECT Country, COUNT(*) AS CustomerCount \r\nFROM cus
tomers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"ro
le": "user", "content": " \n   Find all tracks with a name containing \"Wh
at\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \r\n
FROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "conten
t": " \n   Find the total number of invoices per country:\n"}, {"role": "a
ssistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices
\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGR
OUP BY c.Country"}, {"role": "user", "content": "Can you list all tables in
the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT nam
e FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n
List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "co
ntent": "SELECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user",
"content": " \n   Get the total number of invoices for each customer\n"},
{"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoic
es\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": "H
ow many customers are there"}, {"role": "assistant", "content": "SELECT COUN
T(*) FROM customers"}, {"role": "user", "content": " \n   There are 3 table
s: artists, albums and tracks, where albums and artists are linked by Artist
Id, albums and tracks are linked by AlbumId,\n   Can you find the top 10 mo
st popular artists based on the number of tracks\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:01:21.797568128Z',
'message': {'role': 'assistant', 'content': 'SELECT a.Name, COUNT(t.TrackId)
AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistI
d\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER BY
TrackCount DESC\r\nLIMIT 10;'}, 'done_reason': 'stop', 'done': True, 'total_
duration': 70492173679, 'load_duration': 20138838, 'prompt_eval_count': 124
8, 'prompt_eval_duration': 52265996000, 'eval_count': 72, 'eval_duration': 1
6955433000}
LLM Response: SELECT a.Name, COUNT(t.TrackId) AS TrackCount
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
ORDER BY TrackCount DESC
LIMIT 10;
Info: Output from LLM: SELECT a.Name, COUNT(t.TrackId) AS TrackCount
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name

```
ORDER BY TrackCount DESC
LIMIT 10;
Extracted SQL: SELECT a.Name, COUNT(t.TrackId) AS TrackCount
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
ORDER BY TrackCount DESC
LIMIT 10
SELECT a.Name, COUNT(t.TrackId) AS TrackCount
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
ORDER BY TrackCount DESC
LIMIT 10
              Name  TrackCount
0      Iron Maiden         213
1               U2         135
2     Led Zeppelin         114
3        Metallica         112
4             Lost          92
5      Deep Purple          92
6        Pearl Jam          67
7    Lenny Kravitz          57
8  Various Artists          56
9       The Office          53
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    There are 3 tables: artists, albums and tracks, where albums and artis
ts are linked by ArtistId, albums and tracks are linked by AlbumId,\n    Can
you find the top 10 most popular artists based on the number of tracks\n'\n
\nThe DataFrame was produced using this query: SELECT a.Name, COUNT(t.TrackI
d) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.Arti
stId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER
BY TrackCount DESC\r\nLIMIT 10\n\nThe following is information about the res
ulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Name          obj
ect\nTrackCount     int64\ndtype: object"}, {"role": "user", "content": "Can
you generate the Python plotly code to chart the results of the dataframe? A
ssume the data is in a pandas dataframe called 'df'. If there is only one va
lue in the dataframe, use an Indicator. Respond with only Python code. Do no
t answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:02:00.054277539Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  px.indicator(\n    value=df[\'TrackCount
\'].iloc[0],\n    title="Top Artist by Track Count",\n    mode=\'number\',\n
color_title="Artist Tracks",\n    text=f"{df[\'Name\'].iloc[0]} has {df[\'Tr
ackCount\'].iloc[0]} tracks"\n  )\nelse:\n  px.bar(df, x=\'Name\', y=\'Track
Count\', title="Top 10 Most Popular Artists") \n```'}, 'done_reason': 'sto
p', 'done': True, 'total_duration': 38229193817, 'load_duration': 22124229,
```

```
'prompt_eval_count': 259, 'prompt_eval_duration': 9946620000, 'eval_count':
129, 'eval_duration': 28216316000}
Couldn't run plotly code:  'NoneType' object has no attribute 'show'
```

```
Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1684, in ask
    img_bytes = fig.to_image(format="png", scale=2)
                ^^^^^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'to_image'

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/gongai/anaconda3/envs/vanna/lib/python3.11/site-packages/vann
a/base/base.py", line 1687, in ask
    fig.show()
    ^^^^^^^^
AttributeError: 'NoneType' object has no attribute 'show'
```

In [33]:
```
question = """
     List all customers from Canada and their email addresses:
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Countr

y, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': " \n   List all employees and their reporting manager's name (if any):\n"}, {'role': 'assistant', 'content': "SELECT e.FirstName AS EmployeeFirstName, e.LastName AS EmployeeLastName, \n   CASE WHEN e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FROM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM employees e"}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all customers from Canada and their email addresses:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Las

tName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n
Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n
HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r
\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVA
RCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email
NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"employees\" (Emplo
yeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE
\"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT
NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT
NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId)
REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE \"playlis
t_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, Track
Id),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsT
o)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTE
GER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional C
ontext \n\nIn the chinook database invoice means order\n\n===Response Guidel
ines \n1. If the provided context is sufficient, please generate a valid SQL
query without any explanations for the question. \n2. If the provided contex
t is almost sufficient but requires knowledge of a specific string in a part
icular column, please generate an intermediate SQL query to find the distinc
t strings in that column. Prepend the query with a comment saying intermedia
te_sql \n3. If the provided context is insufficient, please explain why it c
an't be generated. \n4. Please use the most relevant table(s). \n5. If the q
uestion has been asked and answered before, please repeat the answer exactly
as it was given before. \n"}, {"role": "user", "content": "what are the top
5 countries that customers come from?"}, {"role": "assistant", "content": "S
ELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Coun
try\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content":
" \n   Find the total number of invoices per country:\n"}, {"role": "assis
tant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\n
FROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP
BY c.Country"}, {"role": "user", "content": "How many customers are there"},
{"role": "assistant", "content": "SELECT COUNT(*) FROM customers"}, {"role":
"user", "content": " \n   Get the total number of invoices for each custom
er\n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS Tot
alInvoices\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "cont
ent": " \n   List all invoices with a total exceeding $10:\n"}, {"role":
"assistant", "content": "SELECT * \r\nFROM invoices\r\nWHERE Total > 10"},
{"role": "user", "content": " \n   Get the average invoice total for each
customer:\n"}, {"role": "assistant", "content": "SELECT CustomerId, AVG(Tota
l) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"rol
e": "user", "content": " \n   List all employees and their reporting manag
er's name (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstNam
e AS EmployeeFirstName, e.LastName AS EmployeeLastName, \n   CASE WHEN
e.ReportsTo IS NULL THEN 'None' ELSE (SELECT FirstName || ' ' || LastName FR
OM employees WHERE EmployeeId = e.ReportsTo) END AS ManagerName\r\nFROM empl
oyees e"}, {"role": "user", "content": " \n   Find all invoices since 2010

and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   List all customers from Canada and their email addresses:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:03:12.793718447Z', 'message': {'role': 'assistant', 'content': "SELECT FirstName, LastName, Email \r\nFROM customers \r\nWHERE Country = 'Canada';\n\n\n\n"}, 'done_reason': 'stop', 'done': True, 'total_duration': 72685938705, 'load_duration': 23631580, 'prompt_eval_count': 1539, 'prompt_eval_duration': 66853048000, 'eval_count': 20, 'eval_duration': 4588507000}
LLM Response: SELECT FirstName, LastName, Email
FROM customers
WHERE Country = 'Canada';

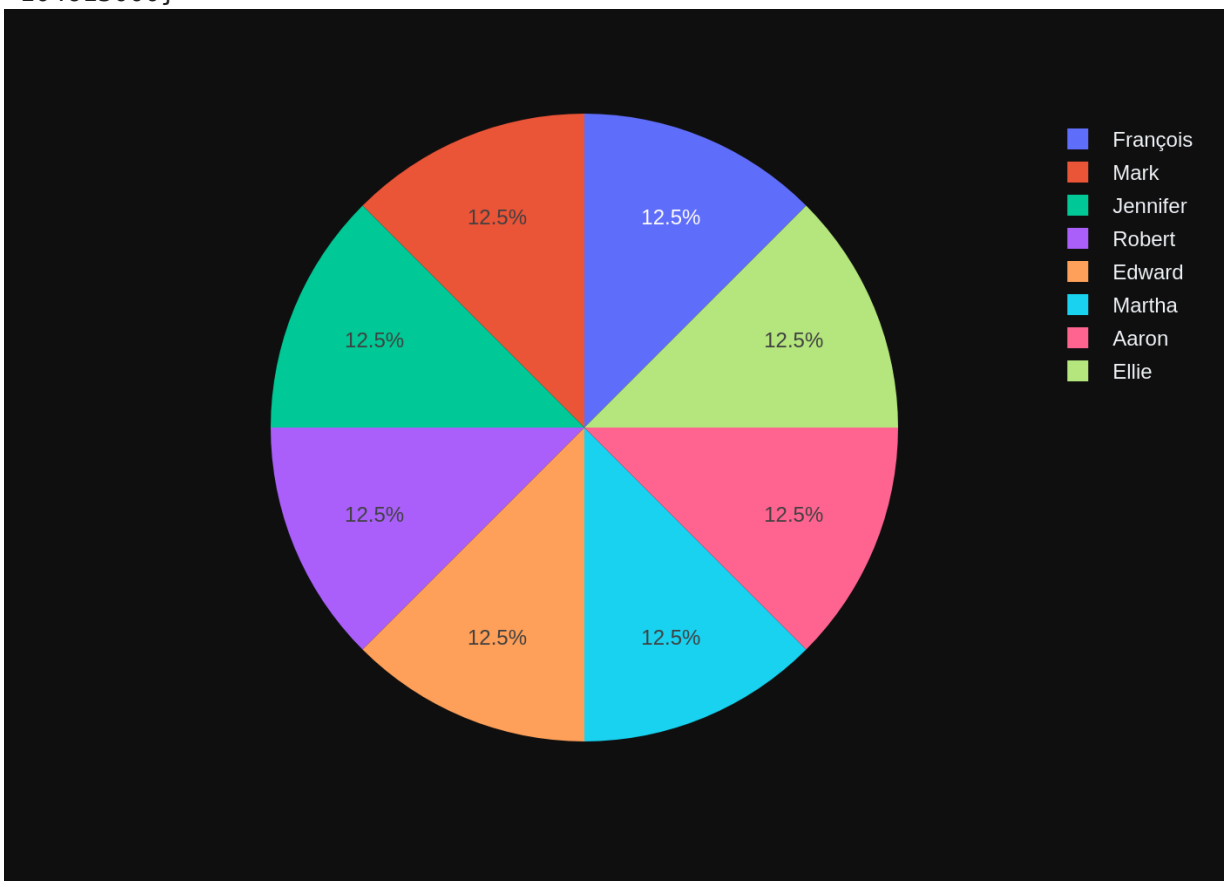
Info: Output from LLM: SELECT FirstName, LastName, Email
FROM customers
WHERE Country = 'Canada';

Extracted SQL: SELECT FirstName, LastName, Email
FROM customers
WHERE Country = 'Canada'
SELECT FirstName, LastName, Email
FROM customers
WHERE Country = 'Canada'

	FirstName	LastName	Email
0	François	Tremblay	ftremblay@gmail.com
1	Mark	Philips	mphilips12@shaw.ca
2	Jennifer	Peterson	jenniferp@rogers.ca
3	Robert	Brown	robbrown@shaw.ca
4	Edward	Francis	edfrancis@yachoo.ca
5	Martha	Silk	marthasilk@gmail.com
6	Aaron	Mitchell	aaronmitchell@yahoo.ca
7	Ellie	Sullivan	ellie.sullivan@shaw.ca

Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that contains the results of the query that answers the question the user asked: '\n   List all customers from Canada and their email addresses:\n'\n\nThe DataFrame was produced using this query: SELECT FirstName, LastName, Email \r\nFROM customers \r\nWHERE Country = 'Canada'\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Fi

rstName object\nLastName object\nEmail object\ndtype: objec
t"}, {"role": "user", "content": "Can you generate the Python plotly code to
chart the results of the dataframe? Assume the data is in a pandas dataframe
called 'df'. If there is only one value in the dataframe, use an Indicator.
Respond with only Python code. Do not answer with any explanations -- just t
he code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:03:39.107968875Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n px.indicator(\n value=df[\'Email\'].il
oc[0],\n title="Customer Email from Canada",\n)\nelse:\n fig = px.scat
ter(df, x="FirstName", y="LastName", size="Email")\n fig.update_layout(titl
e="Canadian Customers") \n```'}, 'done_reason': 'stop', 'done': True, 'total
_duration': 26294315090, 'load_duration': 19992287, 'prompt_eval_count': 16
8, 'prompt_eval_duration': 6122782000, 'eval_count': 93, 'eval_duration': 20
104615000}



Out[33]:
```
("SELECT FirstName, LastName, Email \r\nFROM customers \r\nWHERE Country =
'Canada'",
   FirstName  LastName                  Email
 0  François  Tremblay     ftremblay@gmail.com
 1      Mark   Philips      mphilips12@shaw.ca
 2  Jennifer  Peterson     jenniferp@rogers.ca
 3    Robert     Brown        robbrown@shaw.ca
 4    Edward   Francis     edfrancis@yachoo.ca
 5    Martha      Silk    marthasilk@gmail.com
 6     Aaron  Mitchell  aaronmitchell@yahoo.ca
 7     Ellie  Sullivan  ellie.sullivan@shaw.ca,
 Figure({
     'data': [{'domain': {'x': [0.0, 1.0], 'y': [0.0, 1.0]},
               'hovertemplate': 'FirstName=%{label}<extra></extra>',
               'labels': array(['François', 'Mark', 'Jennifer', 'Robert',
'Edward', 'Martha', 'Aaron',
                                'Ellie'], dtype=object),
               'legendgroup': '',
               'name': '',
               'showlegend': True,
               'type': 'pie'}],
     'layout': {'legend': {'tracegroupgap': 0}, 'margin': {'t': 60}, 'templ
ate': '...'}
 }))
```

In [34]:
```
question = """
     Find the customer with the most invoices
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   Get the

total number of invoices for each customer\n'}, {'role': 'assistant', 'conte
nt': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP
BY CustomerId'}, {'role': 'user', 'content': ' \n   Find the total number
of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Cou
ntry, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoic
es i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'use
r', 'content': ' \n   List all invoices with a total exceeding $10:\n'},
{'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHERE Total
> 10'}, {'role': 'user', 'content': ' \n   Get the average invoice total f
or each customer:\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId,
AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId'},
{'role': 'user', 'content': ' \n   Find all invoices since 2010 and the to
tal amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Tota
l) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-0
1'"}, {'role': 'user', 'content': 'what are the top 5 countries that custome
rs come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*)
AS CustomerCount \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY Customer
Count DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find the top 5
most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'con
tent': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\n
LIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'}, {'ro
le': 'assistant', 'content': 'SELECT COUNT(*) FROM customers'}, {'role': 'us
er', 'content': ' \n   List all customers from Canada and their email add
resses:\n'}, {'role': 'assistant', 'content': "SELECT FirstName, LastName, E
mail \r\nFROM customers \r\nWHERE Country = 'Canada'"}, {'role': 'user', 'co
ntent': ' \n   There are 3 tables: artists, albums and tracks, where albums
and artists are linked by ArtistId, albums and tracks are linked by AlbumI
d,\n   Can you find the top 10 most popular artists based on the number of
tracks\n'}, {'role': 'assistant', 'content': 'SELECT a.Name, COUNT(t.TrackI
d) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.Arti
stId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER
BY TrackCount DESC\r\nLIMIT 10'}, {'role': 'user', 'content': ' \n   Find
the customer with the most invoices \n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK
_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"inv
oice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NUL
L,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NU
LL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n
\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) RE
FERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO

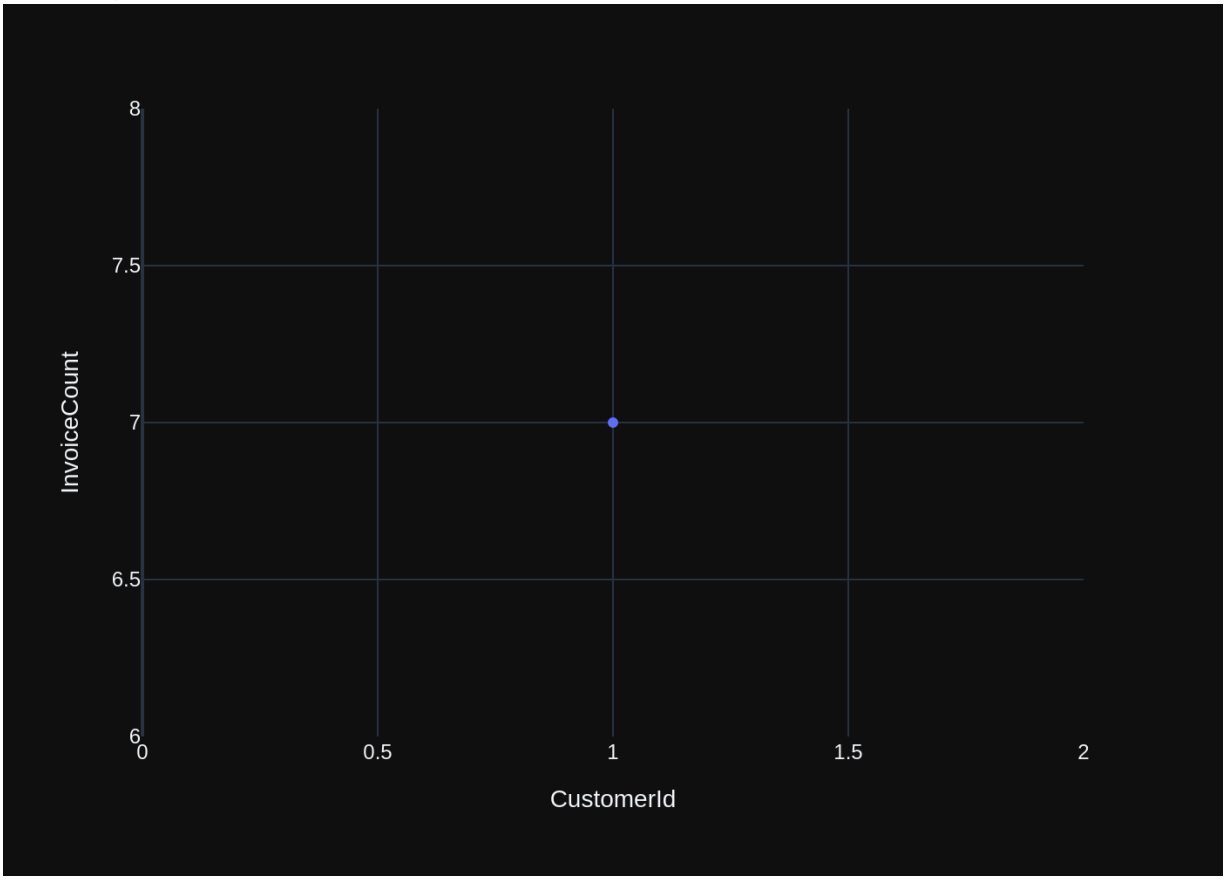
N\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)
\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AU
TOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastNa
me NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARC
HAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country
NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId I
NTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (Employee
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK
_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"empl
oyees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NU
LL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DA
TETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City N
VARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2
4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e
mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\nCREAT
E TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   Me
diaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHA
R(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means ord
er\n\n===Response Guidelines \n1. If the provided context is sufficient, ple
ase generate a valid SQL query without any explanations for the question. \n
2. If the provided context is almost sufficient but requires knowledge of a
specific string in a particular column, please generate an intermediate SQL
query to find the distinct strings in that column. Prepend the query with a
comment saying intermediate_sql \n3. If the provided context is insufficien
t, please explain why it can't be generated. \n4. Please use the most releva
nt table(s). \n5. If the question has been asked and answered before, please
repeat the answer exactly as it was given before. \n"}, {"role": "user", "co
ntent": " \n   Get the total number of invoices for each customer\n"}, {"r
ole": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoices
\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": "
\n   Find the total number of invoices per country:\n"}, {"role": "assistan
t", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFRO
M customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY
c.Country"}, {"role": "user", "content": " \n   List all invoices with a t
otal exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM
invoices\r\nWHERE Total > 10"}, {"role": "user", "content": " \n   Get the
average invoice total for each customer:\n"}, {"role": "assistant", "conten
t": "SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r
\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   Find all invoi
ces since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "co
ntent": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE Invoi
ceDate >= '2010-01-01'"}, {"role": "user", "content": "what are the top 5 co
untries that customers come from?"}, {"role": "assistant", "content": "SELEC
T Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGROUP BY Country
\r\nORDER BY CustomerCount DESC\r\nLIMIT 5"}, {"role": "user", "content": "

\n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"rol
e": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER
BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": "How many custom
ers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM cust
omers"}, {"role": "user", "content": " \n   List all customers from Canad
a and their email addresses:\n"}, {"role": "assistant", "content": "SELECT F
irstName, LastName, Email \r\nFROM customers \r\nWHERE Country = 'Canada'"},
{"role": "user", "content": " \n   There are 3 tables: artists, albums and
tracks, where albums and artists are linked by ArtistId, albums and tracks a
re linked by AlbumId,\n   Can you find the top 10 most popular artists base
d on the number of tracks\n"}, {"role": "assistant", "content": "SELECT a.Na
me, COUNT(t.TrackId) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.
ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP B
Y a.Name\r\nORDER BY TrackCount DESC\r\nLIMIT 10"}, {"role": "user", "conten
t": " \n   Find the customer with the most invoices \n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:04:59.515075199Z',
'message': {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS
InvoiceCount\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCoun
t DESC\r\nLIMIT 1;\n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_du
ration': 80317013233, 'load_duration': 21441883, 'prompt_eval_count': 1627,
'prompt_eval_duration': 70993161000, 'eval_count': 34, 'eval_duration': 8049
760000}
LLM Response: SELECT CustomerId, COUNT(*) AS InvoiceCount
FROM invoices
GROUP BY CustomerId
ORDER BY InvoiceCount DESC
LIMIT 1;

Info: Output from LLM: SELECT CustomerId, COUNT(*) AS InvoiceCount
FROM invoices
GROUP BY CustomerId
ORDER BY InvoiceCount DESC
LIMIT 1;

Extracted SQL: SELECT CustomerId, COUNT(*) AS InvoiceCount
FROM invoices
GROUP BY CustomerId
ORDER BY InvoiceCount DESC
LIMIT 1
SELECT CustomerId, COUNT(*) AS InvoiceCount
FROM invoices
GROUP BY CustomerId
ORDER BY InvoiceCount DESC
LIMIT 1
   CustomerId InvoiceCount
0   1   7
Info: Ollama parameters:
model=gemma2:latest,
options={},

keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n   Find the customer with the most invoices \n'\n\nThe DataFrame was pro
duced using this query: SELECT CustomerId, COUNT(*) AS InvoiceCount\r\nFROM
invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCount DESC\r\nLIMIT 1\n\n
The following is information about the resulting pandas DataFrame 'df': \nRu
nning df.dtypes gives:\n CustomerId   int64\nInvoiceCount   int64\ndtyp
e: object"}, {"role": "user", "content": "Can you generate the Python plotly
code to chart the results of the dataframe? Assume the data is in a pandas d
ataframe called 'df'. If there is only one value in the dataframe, use an In
dicator. Respond with only Python code. Do not answer with any explanations
-- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:05:25.026346189Z',
'message': {'role': 'assistant', 'content': "```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n fig = px.indicator(df, name='InvoiceCoun
t', value='InvoiceCount', title='Customer with Most Invoices')\nelse:\n fig
= px.bar(df, x='CustomerId', y='InvoiceCount', title='Customers by Invoice C
ount')\n\nfig.show() \n```"}, 'done_reason': 'stop', 'done': True, 'total_du
ration': 25486242299, 'load_duration': 17626442, 'prompt_eval_count': 181,
'prompt_eval_duration': 6661457000, 'eval_count': 87, 'eval_duration': 18761
146000}



Out[34]:
```
('SELECT CustomerId, COUNT(*) AS InvoiceCount\r\nFROM invoices\r\nGROUP BY
CustomerId\r\nORDER BY InvoiceCount DESC\r\nLIMIT 1',
    CustomerId  InvoiceCount
 0           1             7,
 Figure({
     'data': [{'hovertemplate': 'CustomerId=%{x}<br>InvoiceCount=%{y}<extra
></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'symbol': 'circle'},
               'mode': 'markers',
               'name': '',
               'orientation': 'v',
               'showlegend': False,
               'type': 'scatter',
               'x': array([1]),
               'xaxis': 'x',
               'y': array([7]),
               'yaxis': 'y'}],
      'layout': {'legend': {'tracegroupgap': 0},
                 'margin': {'t': 60},
                 'template': '...',
                 'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'CustomerId'}},
                 'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'InvoiceCount'}}}
 }))
```

In [ ]:
```
```

Advanced SQL questions

In [35]:
```
question = """
     Find the customer who bought the most albums in total quantity (across
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "artists"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   Find the customer with the most invoices \n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS InvoiceCount\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCount DESC\r\nLIMIT 1'}, {'role': 'user', 'content': ' \n   There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks are linked by AlbumId,\n Can you find the top 10 most popular artists based on the number of tracks \n'}, {'role': 'assistant', 'content': 'SELECT a.Name, COUNT(t.TrackId) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER BY TrackCount DESC\r\nLIMIT 10'}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'conten

t': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   List all genres and the number of tracks in each genre:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name'}, {'role': 'user', 'content': ' \n   Find the customer who bought the most albums in total quantity (across all invoices):\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   Billin

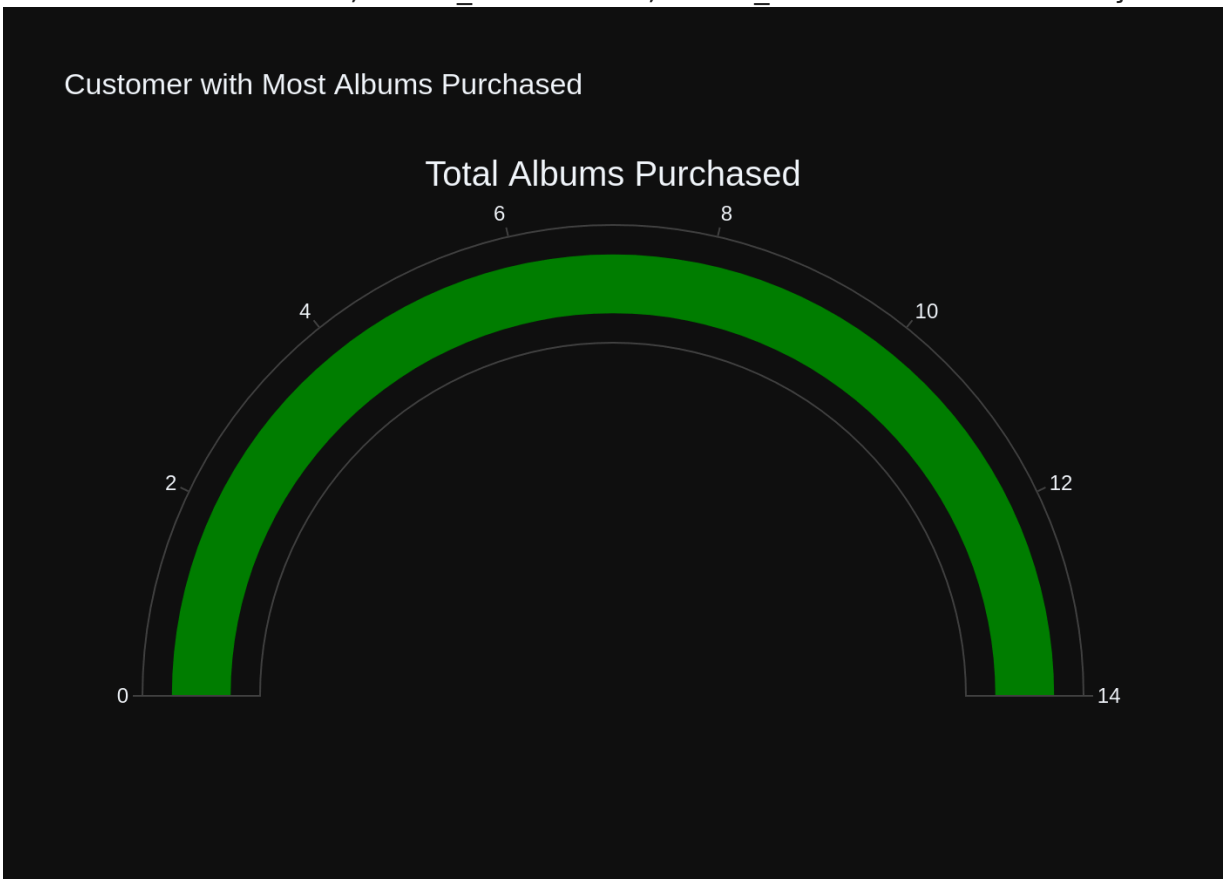
gCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": " \n   Find the customer with the most invoices \n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS InvoiceCount\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCount DESC\r\nLIMIT 1"}, {"role": "user", "content": " \n   There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks are linked by AlbumId,\n   Can you find the top 10 most popular artists based on the number of tracks\n"}, {"role": "assistant", "content": "SELECT a.Name, COUNT(t.TrackId) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER BY TrackCount DESC\r\nLIMIT 10"}, {"role": "user", "content": " \n   Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n   Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user", "content": " \n   Get the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   List all genres and the number of tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name"}, {"role": "user", "content": " \n   Find the customer who bought the most albums in total quantity (across all invoices): \n"}]
Info: Ollama Response:

{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:06:43.02491497Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 1;\n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_duration': 77912153353, 'load_duration': 23256244, 'prompt_eval_count': 1462, 'prompt_eval_duration': 62595037000, 'eval_count': 59, 'eval_duration': 14018099000}
LLM Response: SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased
FROM customers c
JOIN invoice_items ii ON c.CustomerId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 1;

Info: Output from LLM: SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased
FROM customers c
JOIN invoice_items ii ON c.CustomerId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 1;

Extracted SQL: SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased
FROM customers c
JOIN invoice_items ii ON c.CustomerId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 1
SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased
FROM customers c
JOIN invoice_items ii ON c.CustomerId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 1
   CustomerId TotalAlbumsPurchased
0   5   14
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that contains the results of the query that answers the question the user asked: '
\n   Find the customer who bought the most albums in total quantity (across all invoices): \n'\n\nThe DataFrame was produced using this query: SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 1\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes give

s:\n CustomerId   int64\nTotalAlbumsPurchased   int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the results of the dataframe? Assume the data is in a pandas dataframe called 'df'. If there is only one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:07:26.726665637Z', 'message': {'role': 'assistant', 'content': "```python\nimport plotly.graph_objects as go\n\nfig = go.Figure()\n\nif df.shape[0] == 1:\n fig.add_trace(go.Indicator(\n   mode='gauge',\n   value=df['TotalAlbumsPurchased'].iloc[0],\n   title={'text': 'Total Albums Purchased'},\n   gauge={'axis': {'range': [0, df['TotalAlbumsPurchased'].iloc[0]]}}\n)) \nelse:\n fig.add_trace(go.Bar(x=[df['CustomerId'].iloc[0]], y=[df['TotalAlbumsPurchased'].iloc[0]]))\n\nfig.update_layout(title='Customer with Most Albums Purchased')\nfig.show() \n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 43674938810, 'load_duration': 20925811, 'prompt_eval_count': 216, 'prompt_eval_duration': 7994250000, 'eval_count': 162, 'eval_duration': 35616673000}



Out[35]:
```
('SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM cus
tomers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP B
Y c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 1',
    CustomerId  TotalAlbumsPurchased
 0           5                   14,
 Figure({
     'data': [{'gauge': {'axis': {'range': [0, 14]}},
               'mode': 'gauge',
               'title': {'text': 'Total Albums Purchased'},
               'type': 'indicator',
               'value': 14}],
     'layout': {'template': '...', 'title': {'text': 'Customer with Most Al
bums Purchased'}}
 }))
```

In [36]:
```
question = """
    Hint: album quantity is found in invoice_items,

    Find the top 5 customers who bought the most albums in total quantity (a
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoice_items"\r\n(\r\n    Invo
iceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEG
ER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,
2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (Invoice
Id) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n
(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVA
RCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NO
T NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Millis
econds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,
2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId)
REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "album
s"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Ti
tle NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREI
GN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (Ar
tistId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (Invoice
Id)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nC
REATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCREME
NT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIM
E  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR
(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCountry NVARCHAR(4
0),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT N
ULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_Invoi
ceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_TrackAlbumId ON
"tracks" (AlbumId)\n\nCREATE TABLE "artists"\r\n(\r\n    ArtistId INTEGER PR
IMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Add
itional Context \n\nIn the chinook database invoice means order\n\n===Respon
se Guidelines \n1. If the provided context is sufficient, please generate a
valid SQL query without any explanations for the question. \n2. If the provi
ded context is almost sufficient but requires knowledge of a specific string
in a particular column, please generate an intermediate SQL query to find th
e distinct strings in that column. Prepend the query with a comment saying i
ntermediate_sql \n3. If the provided context is insufficient, please explain
why it can\'t be generated. \n4. Please use the most relevant table(s). \n5.
If the question has been asked and answered before, please repeat the answer
exactly as it was given before. \n'}, {'role': 'user', 'content': '  \n
Find the customer who bought the most albums in total quantity (across all i
nvoices): \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(i
i.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_item
s ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY Tot
alAlbumsPurchased DESC\r\nLIMIT 1'}, {'role': 'user', 'content': ' \n    The
re are 3 tables: artists, albums and tracks, where albums and artists are li
nked by ArtistId, albums and tracks are linked by AlbumId,\n    Can you find
the top 10 most popular artists based on the number of tracks\n'}, {'role':
'assistant', 'content': 'SELECT a.Name, COUNT(t.TrackId) AS TrackCount\r\nFR
OM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t
ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER BY TrackCount DESC\r\n
```

LIMIT 10'}, {'role': 'user', 'content': ' \n   Find the customer with the most invoices \n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS InvoiceCount\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCount DESC\r\nLIMIT 1'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': ' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   Hint: album quantity is found in invoice_items, \n   \n   Find the top 5 customers who bought the most albums in total quantity (across all invoices):\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\"

(InvoiceId)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order \n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": " \n   Find the customer who bought the most albums in total quantity (across all invoices): \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 1"}, {"role": "user", "content": " \n   There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks are linked by AlbumId,\n Can you find the top 10 most popular artists based on the number of tracks \n"}, {"role": "assistant", "content": "SELECT a.Name, COUNT(t.TrackId) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER BY TrackCount DESC\r\nLIMIT 10"}, {"role": "user", "content": " \n   Find the customer with the most invoices \n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS InvoiceCount\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCount DESC\r\nLIMIT 1"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country"}, {"role": "user", "content": " \n   List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user", "content": " \n   Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n   Get the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.Art

```
istId"}, {"role": "user", "content": "  \n    Hint: album quantity is found
in invoice_items, \n    \n    Find the top 5 customers who bought the most a
lbums in total quantity (across all invoices):\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:08:46.857550971Z',
'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(ii.Qua
ntity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii
ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlb
umsPurchased DESC\r\nLIMIT 5  \n\n\n\n'}, 'done_reason': 'stop', 'done': Tru
e, 'total_duration': 80053309009, 'load_duration': 21446675, 'prompt_eval_co
unt': 1499, 'prompt_eval_duration': 64732391000, 'eval_count': 59, 'eval_dur
ation': 14037128000}
LLM Response: SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased
FROM customers c
JOIN invoice_items ii ON c.CustomerId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 5

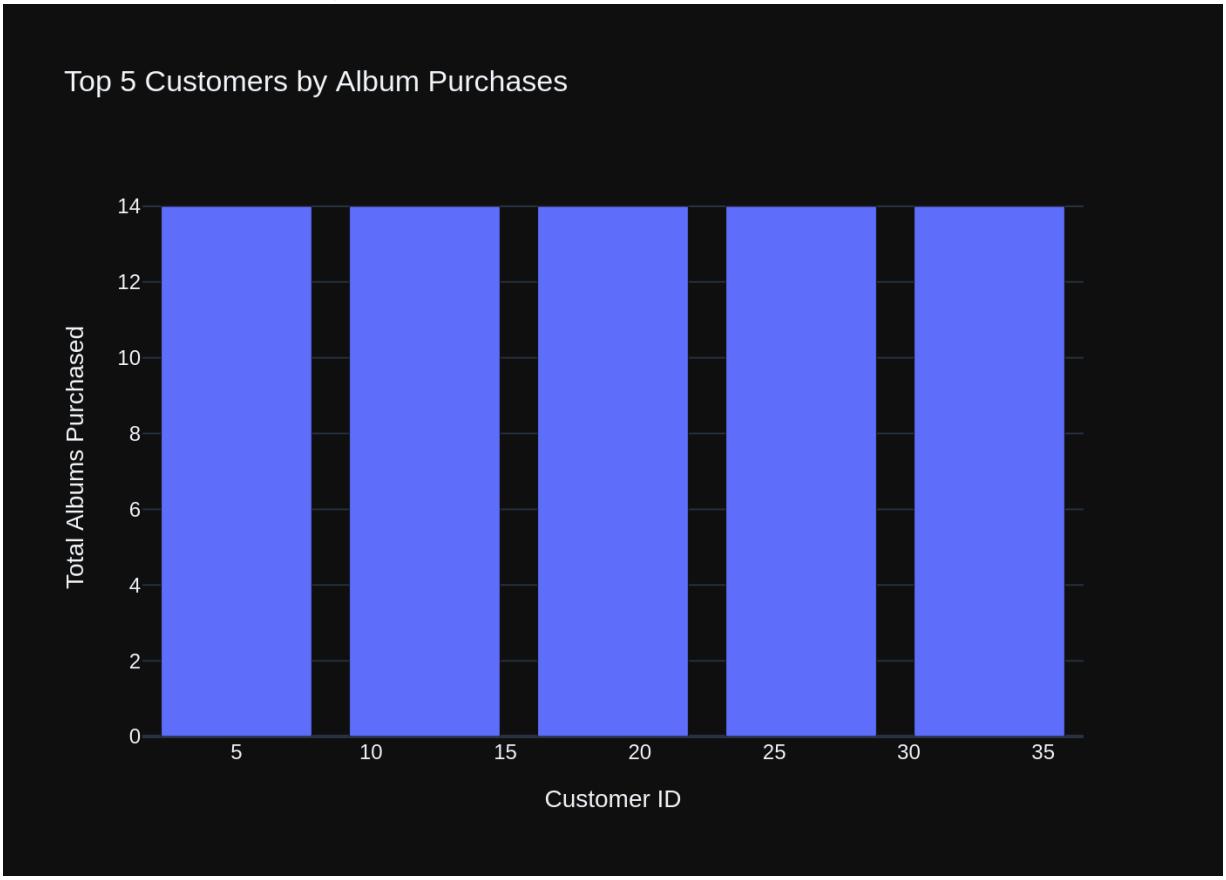


SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased
FROM customers c
JOIN invoice_items ii ON c.CustomerId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 5



   CustomerId  TotalAlbumsPurchased
0           5                    14
1          12                    14
2          19                    14
3          26                    14
4          33                    14
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Hint: album quantity is found in invoice_items, \n    \n    Find the t
op 5 customers who bought the most albums in total quantity (across all invo
ices):\n'\n\nThe DataFrame was produced using this query: SELECT c.CustomerI
d, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invo
ice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDE
R BY TotalAlbumsPurchased DESC\r\nLIMIT 5  \n\n\n\n\n\nThe following is info
rmation about the resulting pandas DataFrame 'df': \nRunning df.dtypes give
s:\n CustomerId              int64\nTotalAlbumsPurchased    int64\ndtype: ob
ject"}, {"role": "user", "content": "Can you generate the Python plotly code
to chart the results of the dataframe? Assume the data is in a pandas datafr
ame called 'df'. If there is only one value in the dataframe, use an Indicat
```

or. Respond with only Python code. Do not answer with any explanations -- ju
st the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:09:25.188096103Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n px.indicator(\n   value=df[\'TotalAlbums
Purchased\'].iloc[0],\n   title="Top Customer Album Purchases",\n   mode=
\'number\',\n   textfont_size=30,\n   color="gray"\n)\nelse:\n fig = p
x.bar(df, x=\'CustomerId\', y=\'TotalAlbumsPurchased\', title="Top 5 Custome
rs by Album Purchases")\n fig.update_layout(xaxis_title="Customer ID", yaxi
s_title="Total Albums Purchased")\n\n```'}, 'done_reason': 'stop', 'done': T
rue, 'total_duration': 38310559890, 'load_duration': 20861076, 'prompt_eval_
count': 235, 'prompt_eval_duration': 8693891000, 'eval_count': 135, 'eval_du
ration': 29550677000}



Out[36]:
```
('SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM cus
tomers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP B
Y c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 5  \n\n\n\n',
    CustomerId  TotalAlbumsPurchased
 0           5                   14
 1          12                   14
 2          19                   14
 3          26                   14
 4          33                   14,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>TotalAlbumsPurchased=%
{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([ 5, 12, 19, 26, 33]),
               'xaxis': 'x',
               'y': array([14, 14, 14, 14, 14]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Top 5 Customers by Album Purchases'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Customer ID'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Total Albums Purchased'}}}
 }))
```

```sql
SELECT c.CustomerId, SUM(il.Quantity) AS TotalAlbums
FROM Customers c
JOIN invoices i  ON c.CustomerId = i.CustomerId
JOIN invoice_items il ON i.InvoiceId = il.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbums DESC
LIMIT 5
```

In [37]:
```
question = """
     Find the top 5 customers who spent the most money overall,

     Hint: order total can be found on invoices table, calculation using inv
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "customers"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "employees"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the qu

ery with a comment saying intermediate_sql \n3. If the provided context is i
nsufficient, please explain why it can\'t be generated. \n4. Please use the
most relevant table(s). \n5. If the question has been asked and answered bef
ore, please repeat the answer exactly as it was given before. \n'}, {'role':
'user', 'content': ' \n   Hint: album quantity is found in invoice_items,
\n   \n   Find the top 5 customers who bought the most albums in total qua
ntity (across all invoices):\n'}, {'role': 'assistant', 'content': 'SELECT
c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r
\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.Custome
rId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 5 \n\n\n\n'}, {'role':
'user', 'content': ' \n   Find the customer who bought the most albums in
total quantity (across all invoices): \n'}, {'role': 'assistant', 'content':
'SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM custo
mers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY
c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 1'}, {'role': 'u
ser', 'content': ' \n   Find the customer with the most invoices \n'},
{'role': 'assistant', 'content': 'SELECT CustomerId, COUNT(*) AS InvoiceCoun
t\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY InvoiceCount DESC\r\nL
IMIT 1'}, {'role': 'user', 'content': ' \n   Get the average invoice total
for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT CustomerId,
AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGROUP BY CustomerId'},
{'role': 'user', 'content': ' \n   Find the total number of invoices per c
ountry:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.Inv
oiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN invoices i ON c.Custome
rId = i.CustomerId\r\nGROUP BY c.Country'}, {'role': 'user', 'content': '
\n   Get the total number of invoices for each customer\n'}, {'role': 'assi
stant', 'content': 'SELECT CustomerId, COUNT(*) AS TotalInvoices\r\nFROM inv
oices\r\nGROUP BY CustomerId'}, {'role': 'user', 'content': ' \n   Find th
e top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistan
t', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice
DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all invoices wi
th a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \r
\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n
Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'a
ssistant', 'content': "SELECT SUM(Total) AS TotalInvoiced\r\nFROM invoices
\r\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': 'what a
re the top 5 countries that customers come from?'}, {'role': 'assistant', 'c
ontent': 'SELECT Country, COUNT(*) AS CustomerCount \r\nFROM customers\r\nGR
OUP BY Country\r\nORDER BY CustomerCount DESC\r\nLIMIT 5'}, {'role': 'user',
'content': ' \n   Find the top 5 customers who spent the most money overa
ll, \n   \n   Hint: order total can be found on invoices table, calculat
ion using invoice_items detail table is unnecessary \n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"

(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOIN CREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (I nvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId) \n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCR EATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCR EMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVA RCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(7 0),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVAR CHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId I NTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (Employee Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"e mployees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NO T NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDat e DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   Ci ty NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2 4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n) \n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCR EMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGE R,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Comp oser NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTE GER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT ION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFER ENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGE R NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_Playlist Track PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) RE FERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t \tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeR eportsTo ON \"employees\" (ReportsTo)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the p rovided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost suffic ient but requires knowledge of a specific string in a particular column, ple ase generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been as ked and answered before, please repeat the answer exactly as it was given be fore. \n"}, {"role": "user", "content": " \n   Hint: album quantity is fou nd in invoice_items, \n   \n   Find the top 5 customers who bought the mos t albums in total quantity (across all invoices):\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r \nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r

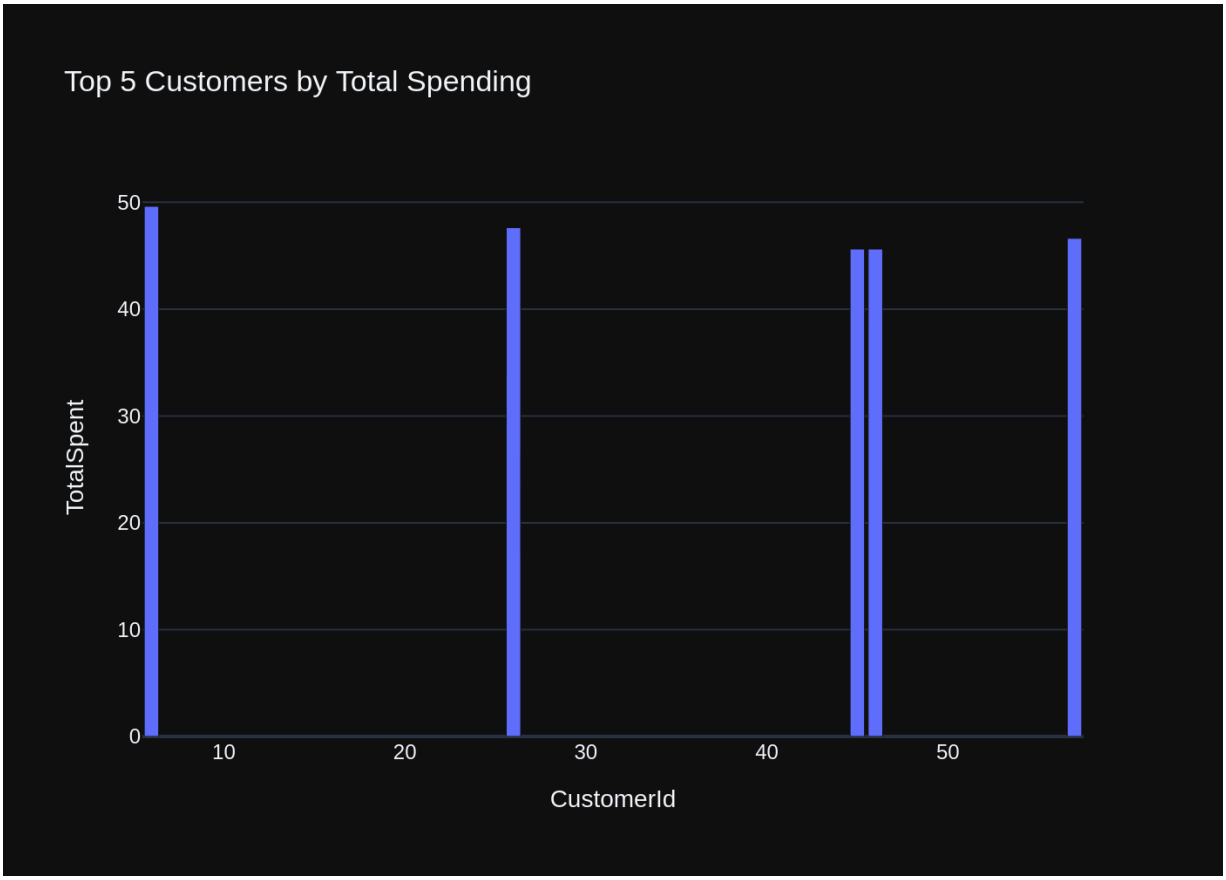
\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 5 \n
\n\n\n"}, {"role": "user", "content": " \n   Find the customer who bought
the most albums in total quantity (across all invoices): \n"}, {"role": "ass
istant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPur
chased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.Inv
oiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIM
IT 1"}, {"role": "user", "content": " \n   Find the customer with the mos
t invoices \n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT
(*) AS InvoiceCount\r\nFROM invoices\r\nGROUP BY CustomerId\r\nORDER BY Invo
iceCount DESC\r\nLIMIT 1"}, {"role": "user", "content": " \n   Get the ave
rage invoice total for each customer:\n"}, {"role": "assistant", "content":
"SELECT CustomerId, AVG(Total) AS AverageInvoiceTotal\r\nFROM invoices\r\nGR
OUP BY CustomerId"}, {"role": "user", "content": " \n   Find the total num
ber of invoices per country:\n"}, {"role": "assistant", "content": "SELECT
c.Country, COUNT(i.InvoiceId) AS TotalInvoices\r\nFROM customers c\r\nJOIN i
nvoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.Country"}, {"role":
"user", "content": " \n   Get the total number of invoices for each custom
er\n"}, {"role": "assistant", "content": "SELECT CustomerId, COUNT(*) AS Tot
alInvoices\r\nFROM invoices\r\nGROUP BY CustomerId"}, {"role": "user", "cont
ent": " \n   Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tr
acks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": "
\n   List all invoices with a total exceeding $10:\n"}, {"role": "assistan
t", "content": "SELECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role":
"user", "content": " \n   Find all invoices since 2010 and the total amoun
t invoiced:\n"}, {"role": "assistant", "content": "SELECT SUM(Total) AS Tota
lInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-01'"}, {"role":
"user", "content": "what are the top 5 countries that customers come fro
m?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) AS Customer
Count \r\nFROM customers\r\nGROUP BY Country\r\nORDER BY CustomerCount DESC
\r\nLIMIT 5"}, {"role": "user", "content": " \n   Find the top 5 customer
s who spent the most money overall, \n   \n   Hint: order total can be f
ound on invoices table, calculation using invoice_items detail table is unne
cessary \n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:10:58.534649086Z',
'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(i.Tota
l) AS TotalSpent\r\nFROM customers c\r\nJOIN invoices i ON c.CustomerId = i.
CustomerId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalSpent DESC\r\nLIMIT 5
\n\n\n\n'}, 'done_reason': 'stop', 'done': True, 'total_duration': 932159342
34, 'load_duration': 18048479, 'prompt_eval_count': 1794, 'prompt_eval_durat
ion': 78916901000, 'eval_count': 54, 'eval_duration': 13041725000}
LLM Response: SELECT c.CustomerId, SUM(i.Total) AS TotalSpent
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalSpent DESC
LIMIT 5

```
SELECT c.CustomerId, SUM(i.Total) AS TotalSpent
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
```

```
ORDER BY TotalSpent DESC
LIMIT 5
```

```
   CustomerId  TotalSpent
0           6       49.62
1          26       47.62
2          57       46.62
3          45       45.62
4          46       45.62
```

```
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n     Find the top 5 customers who spent the most money overall, \n     \n
Hint: order total can be found on invoices table, calculation using invoice_
items detail table is unnecessary \n'\n\nThe DataFrame was produced using th
is query: SELECT c.CustomerId, SUM(i.Total) AS TotalSpent\r\nFROM customers
c\r\nJOIN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.CustomerId
\r\nORDER BY TotalSpent DESC\r\nLIMIT 5  \n\n\n\n\n\nThe following is inform
ation about the resulting pandas DataFrame 'df': \nRunning df.dtypes give
s:\n CustomerId      int64\nTotalSpent    float64\ndtype: object"}, {"role":
"user", "content": "Can you generate the Python plotly code to chart the res
ults of the dataframe? Assume the data is in a pandas dataframe called 'df'.
If there is only one value in the dataframe, use an Indicator. Respond with
only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:11:24.032860693Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n  fig = px.indicator(df, name="TotalSpent",
value="TotalSpent")\nelse:\n  fig = px.bar(df, x="CustomerId", y="TotalSpen
t", title="Top 5 Customers by Total Spending")\n```'}, 'done_reason': 'sto
p', 'done': True, 'total_duration': 25471059838, 'load_duration': 21027175,
'prompt_eval_count': 233, 'prompt_eval_duration': 8911870000, 'eval_count':
76, 'eval_duration': 16492416000}
```


Top 5 Customers by Total Spending
TotalSpent
0
10
20
30
40
50
10
20
30
40
50
CustomerId

Out[37]:
```
('SELECT c.CustomerId, SUM(i.Total) AS TotalSpent\r\nFROM customers c\r\nJO
IN invoices i ON c.CustomerId = i.CustomerId\r\nGROUP BY c.CustomerId\r\nOR
DER BY TotalSpent DESC\r\nLIMIT 5  \n\n\n\n',
    CustomerId  TotalSpent
 0           6       49.62
 1          26       47.62
 2          57       46.62
 3          45       45.62
 4          46       45.62,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>TotalSpent=%{y}<extra>
</extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([ 6, 26, 57, 45, 46]),
               'xaxis': 'x',
               'y': array([49.62, 47.62, 46.62, 45.62, 45.62]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Top 5 Customers by Total Spending'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'CustomerId'}},
               'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'TotalSpent'}}}
 }))
```

In [38]:
```
question = """
     Get all playlists containing at least 10 tracks and the total duration
"""

vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

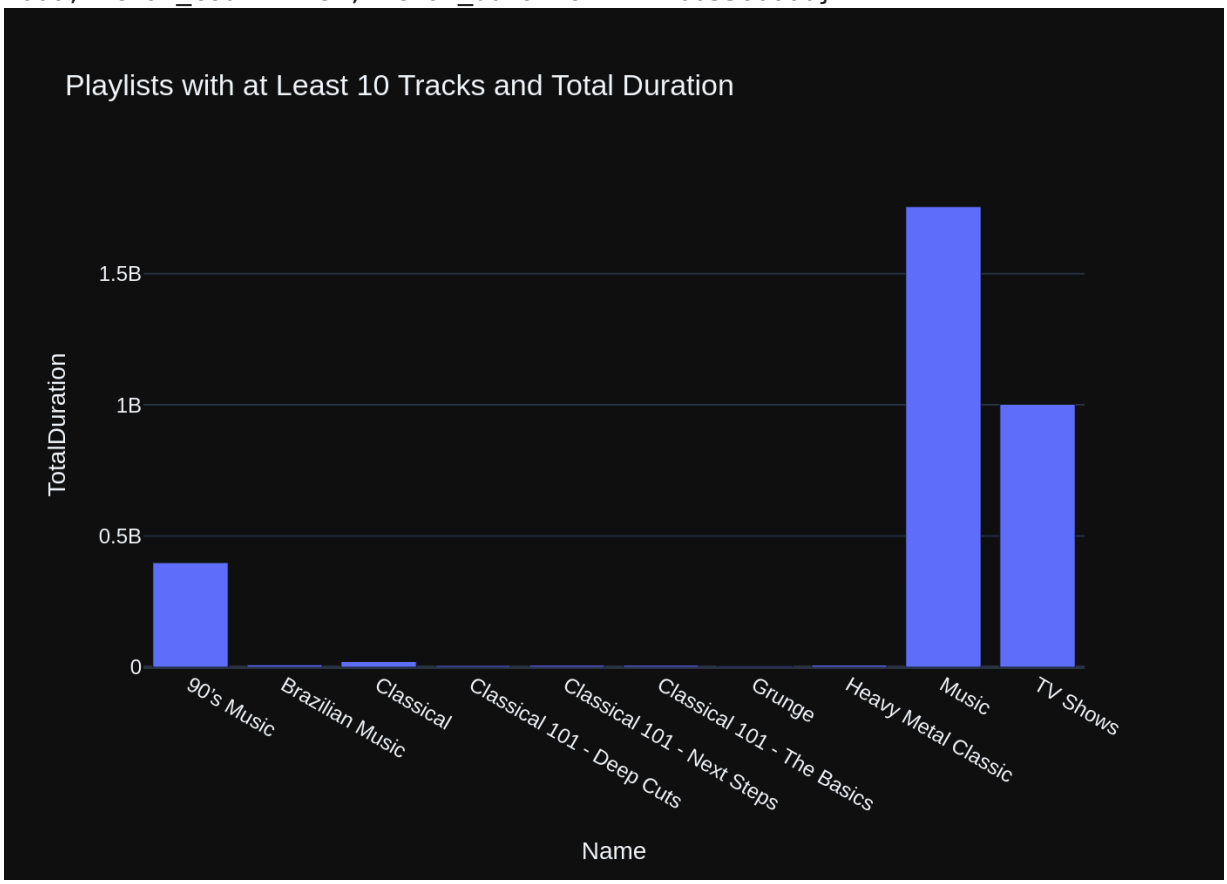
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_PlaylistTrackTrackId ON "pla
ylist_track" (TrackId)\n\nCREATE TABLE "playlists"\r\n(\r\n    PlaylistId IN
TEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n
CREATE TABLE "playlist_track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n
TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY
(PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlist
s" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FORE
IGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION
ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGE
R PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r
\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    GenreId
INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NUL
L,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FO
REIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (Gen
reId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (M
ediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (Ge
nreId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE INDE
X IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE INDEX IFK_AlbumAr
tistId ON "albums" (ArtistId)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT N
ULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERE
NCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINC
REMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Additional Context
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n
1. If the provided context is sufficient, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is a
lmost sufficient but requires knowledge of a specific string in a particular
column, please generate an intermediate SQL query to find the distinct strin
gs in that column. Prepend the query with a comment saying intermediate_sql
\n3. If the provided context is insufficient, please explain why it can\'t b
e generated. \n4. Please use the most relevant table(s). \n5. If the questio
n has been asked and answered before, please repeat the answer exactly as it
was given before. \n'}, {'role': 'user', 'content': '  \n    List all genres
and the number of tracks in each genre:\n'}, {'role': 'assistant', 'conten
t': 'SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN
tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name'}, {'role': 'user', 'co
ntent': ' \n    There are 3 tables: artists, albums and tracks, where albums
and artists are linked by ArtistId, albums and tracks are linked by AlbumI
d,\n    Can you find the top 10 most popular artists based on the number of
tracks\n'}, {'role': 'assistant', 'content': 'SELECT a.Name, COUNT(t.TrackI
d) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.Arti
stId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER
BY TrackCount DESC\r\nLIMIT 10'}, {'role': 'user', 'content': '  \n    Find
all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'a
ssistant', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%
what%'"}, {'role': 'user', 'content': '  \n     Find the customer who bought
the most albums in total quantity (across all invoices): \n'}, {'role': 'ass
istant', 'content': 'SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPur
chased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.Inv
oiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIM
```

IT 1'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive t
racks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT N
ame, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5'}, {'rol
e': 'user', 'content': ' \n   List all albums and their corresponding arti
st names \n'}, {'role': 'assistant', 'content': 'SELECT \n   a.Title,\n
ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId'},
{'role': 'user', 'content': ' \n   Hint: album quantity is found in invoic
e_items, \n   \n   Find the top 5 customers who bought the most albums in
total quantity (across all invoices):\n'}, {'role': 'assistant', 'content':
'SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM custo
mers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY
c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 5 \n\n\n\n'},
{'role': 'user', 'content': ' \n   Find all invoices since 2010 and the to
tal amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT SUM(Tota
l) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-01-0
1'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLite dat
abase catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_
master WHERE type='table'"}, {'role': 'user', 'content': ' \n   List all i
nvoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'S
ELECT * \r\nFROM invoices\r\nWHERE Total > 10'}, {'role': 'user', 'content':
' \n   Get all playlists containing at least 10 tracks and the total dura
tion of those tracks:\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track
\" (TrackId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER PR
IMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE T
ABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   Tra
ckId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (Pla
ylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\"
(PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN
KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON
UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n
AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (Genr
eId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDE
X IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE INDEX IFK_Album
ArtistId ON \"albums\" (ArtistId)\n\nCREATE TABLE \"albums\"\r\n(\r\n   Alb
umId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160)
NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) R
EFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO AC
TION\r\n)\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Additional Co

ntext \n\nIn the chinook database invoice means order\n\n===Response Guideli
nes \n1. If the provided context is sufficient, please generate a valid SQL
query without any explanations for the question. \n2. If the provided contex
t is almost sufficient but requires knowledge of a specific string in a part
icular column, please generate an intermediate SQL query to find the distinc
t strings in that column. Prepend the query with a comment saying intermedia
te_sql \n3. If the provided context is insufficient, please explain why it c
an't be generated. \n4. Please use the most relevant table(s). \n5. If the q
uestion has been asked and answered before, please repeat the answer exactly
as it was given before. \n"}, {"role": "user", "content": " \n   List all
genres and the number of tracks in each genre:\n"}, {"role": "assistant", "c
ontent": "SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\n
JOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name"}, {"role": "use
r", "content": " \n   There are 3 tables: artists, albums and tracks, where
albums and artists are linked by ArtistId, albums and tracks are linked by A
lbumId,\n   Can you find the top 10 most popular artists based on the numbe
r of tracks\n"}, {"role": "assistant", "content": "SELECT a.Name, COUNT(t.Tr
ackId) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.
ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nOR
DER BY TrackCount DESC\r\nLIMIT 10"}, {"role": "user", "content": " \n   F
ind all tracks with a name containing \"What\" (case-insensitive)\n"}, {"rol
e": "assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) L
IKE '%what%'"}, {"role": "user", "content": " \n   Find the customer who
bought the most albums in total quantity (across all invoices): \n"}, {"rol
e": "assistant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalA
lbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId
= ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DES
C\r\nLIMIT 1"}, {"role": "user", "content": " \n   Find the top 5 most exp
ensive tracks (based on unit price):\n"}, {"role": "assistant", "content":
"SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT
5"}, {"role": "user", "content": " \n   List all albums and their correspo
nding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.T
itle,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.
ArtistId"}, {"role": "user", "content": " \n   Hint: album quantity is fou
nd in invoice_items, \n   \n   Find the top 5 customers who bought the mos
t albums in total quantity (across all invoices):\n"}, {"role": "assistant",
"content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r
\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r
\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 5 \n
\n\n\n"}, {"role": "user", "content": " \n   Find all invoices since 2010
and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT
SUM(Total) AS TotalInvoiced\r\nFROM invoices \r\nWHERE InvoiceDate >= '2010-
01-01'"}, {"role": "user", "content": "Can you list all tables in the SQLite
database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqli
te_master WHERE type='table'"}, {"role": "user", "content": " \n   List al
l invoices with a total exceeding $10:\n"}, {"role": "assistant", "content":
"SELECT * \r\nFROM invoices\r\nWHERE Total > 10"}, {"role": "user", "conten
t": " \n   Get all playlists containing at least 10 tracks and the total
duration of those tracks:\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:12:43.742551136Z',
'message': {'role': 'assistant', 'content': '```sql\nSELECT p.Name, SUM(t.Mi
lliseconds) AS TotalDuration\r\nFROM playlists p\r\nJOIN playlist_track pt O
N p.PlaylistId = pt.PlaylistId\r\nJOIN tracks t ON pt.TrackId = t.TrackId\r
\nGROUP BY p.Name\r\nHAVING COUNT(*) >= 10;\n```'}, 'done_reason': 'stop',
'done': True, 'total_duration': 79610219023, 'load_duration': 23521360, 'pro

```
mpt_eval_count': 1414, 'prompt_eval_duration': 60501757000, 'eval_count': 7
5, 'eval_duration': 17827914000}
LLM Response: ```sql
SELECT p.Name, SUM(t.Milliseconds) AS TotalDuration
FROM playlists p
JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId
JOIN tracks t ON pt.TrackId = t.TrackId
GROUP BY p.Name
HAVING COUNT(*) >= 10;
```
Info: Output from LLM: ```sql
SELECT p.Name, SUM(t.Milliseconds) AS TotalDuration
FROM playlists p
JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId
JOIN tracks t ON pt.TrackId = t.TrackId
GROUP BY p.Name
HAVING COUNT(*) >= 10;
```
Extracted SQL: SELECT p.Name, SUM(t.Milliseconds) AS TotalDuration
FROM playlists p
JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId
JOIN tracks t ON pt.TrackId = t.TrackId
GROUP BY p.Name
HAVING COUNT(*) >= 10
SELECT p.Name, SUM(t.Milliseconds) AS TotalDuration
FROM playlists p
JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId
JOIN tracks t ON pt.TrackId = t.TrackId
GROUP BY p.Name
HAVING COUNT(*) >= 10
                        Name  TotalDuration
0                  90’s Music      398705153
1             Brazilian Music        9486559
2                   Classical       21770592
3   Classical 101 - Deep Cuts        6755730
4  Classical 101 - Next Steps        7575051
5  Classical 101 - The Basics        7439811
6                      Grunge        4122018
7         Heavy Metal Classic        8206312
8                       Music     1755366166
9                   TV Shows     1002189914
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n     Get all playlists containing at least 10 tracks and the total duratio
n of those tracks:\n'\n\nThe DataFrame was produced using this query: SELECT
p.Name, SUM(t.Milliseconds) AS TotalDuration\r\nFROM playlists p\r\nJOIN pla
ylist_track pt ON p.PlaylistId = pt.PlaylistId\r\nJOIN tracks t ON pt.TrackI
d = t.TrackId\r\nGROUP BY p.Name\r\nHAVING COUNT(*) >= 10\n\nThe following i
s information about the resulting pandas DataFrame 'df': \nRunning df.dtypes
gives:\n Name             object\nTotalDuration     int64\ndtype: object"},
{"role": "user", "content": "Can you generate the Python plotly code to char
```

t the results of the dataframe? Assume the data is in a pandas dataframe cal
led 'df'. If there is only one value in the dataframe, use an Indicator. Res
pond with only Python code. Do not answer with any explanations -- just the
code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:13:09.862733063Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n fig = px.indicator(df, name="TotalDuratio
n", value="TotalDuration")\nelse:\n fig = px.bar(df, x="Name", y="TotalDura
tion", title="Playlists with at Least 10 Tracks and Total Duration")\n```'},
'done_reason': 'stop', 'done': True, 'total_duration': 26092631224, 'load_du
ration': 22991544, 'prompt_eval_count': 225, 'prompt_eval_duration': 8416880
000, 'eval_count': 81, 'eval_duration': 17605560000}



Out[38]:
```
('SELECT p.Name, SUM(t.Milliseconds) AS TotalDuration\r\nFROM playlists p\r
\nJOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId\r\nJOIN tracks t O
N pt.TrackId = t.TrackId\r\nGROUP BY p.Name\r\nHAVING COUNT(*) >= 10',
                          Name  TotalDuration
 0                 90’s Music      398705153
 1             Brazilian Music        9486559
 2                   Classical       21770592
 3   Classical 101 - Deep Cuts        6755730
 4  Classical 101 - Next Steps        7575051
 5  Classical 101 - The Basics        7439811
 6                      Grunge        4122018
 7         Heavy Metal Classic        8206312
 8                       Music     1755366166
 9                   TV Shows     1002189914,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'Name=%{x}<br>TotalDuration=%{y}<extra></ex
tra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['90’s Music', 'Brazilian Music', 'Classical',
                           'Classical 101 - Deep Cuts', 'Classical 101 - Ne
xt Steps',
                           'Classical 101 - The Basics', 'Grunge', 'Heavy M
etal Classic', 'Music',
                           'TV Shows'], dtype=object),
               'xaxis': 'x',
               'y': array([ 398705153,    9486559,   21770592,    6755730,
7575051,    7439811,
                              4122018,    8206312, 1755366166, 100218991
4]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Playlists with at Least 10 Tracks and To
tal Duration'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Name'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'TotalDuration'}}}
 }))
```

In [39]:
```
question = """
     Identify artists who have albums with tracks appearing in multiple genr


"""
```

```
vn.ask(question=question)
```

```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

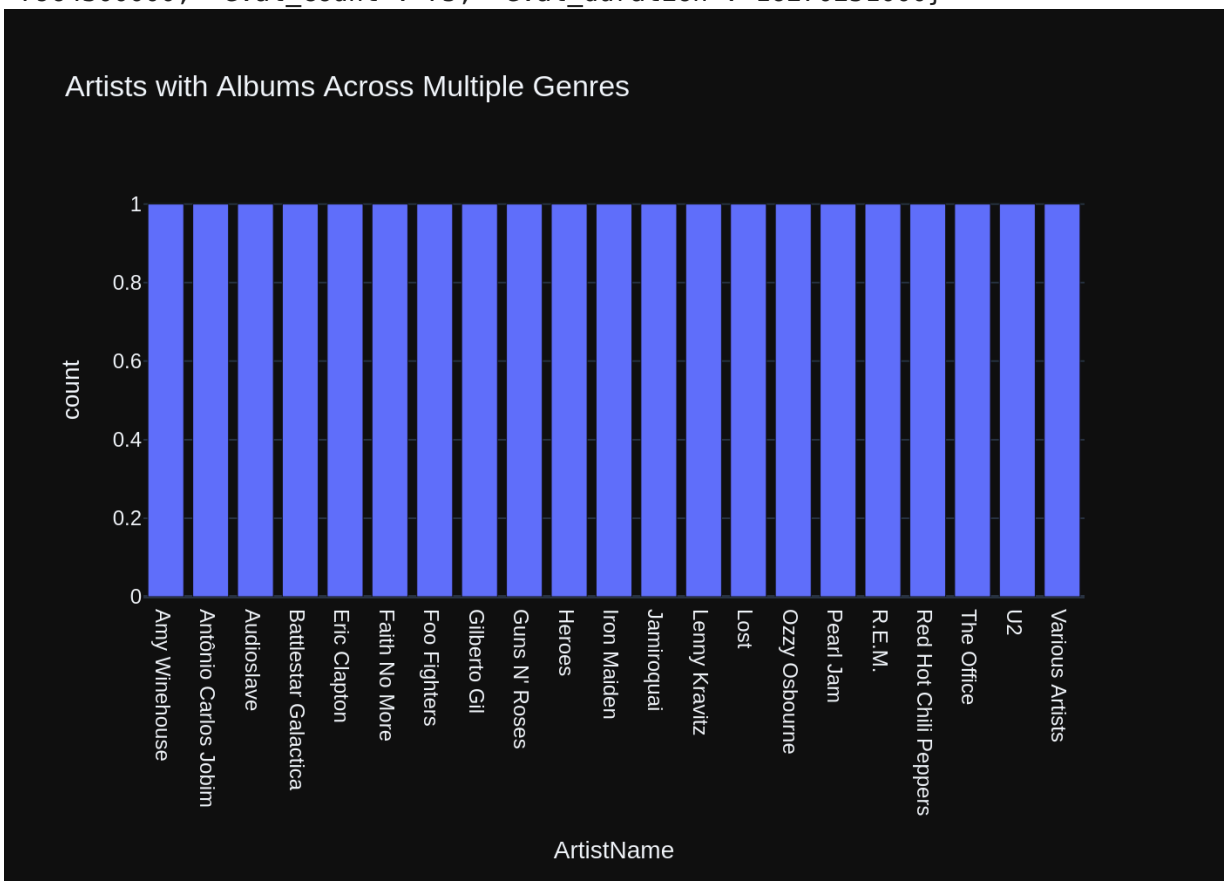
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n    TrackId INT
EGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NUL
L,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    Genr
eId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT
NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n
FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACT
ION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (G
enreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums"
(ArtistId)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE
INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "albums"\r\n(\r
\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARC
HAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (A
rtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDA
TE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTyp
eId)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_Playlis
tTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE TABLE "artists"\r\n(\r
\n    ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARC
HAR(120)\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n    PlaylistId INTEGE
R  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_Playlist
Track PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) RE
FERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO A
CTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn
the chinook database invoice means order\n\n===Response Guidelines \n1. If t
he provided context is sufficient, please generate a valid SQL query without
any explanations for the question. \n2. If the provided context is almost su
fficient but requires knowledge of a specific string in a particular column,
please generate an intermediate SQL query to find the distinct strings in th
at column. Prepend the query with a comment saying intermediate_sql \n3. If
the provided context is insufficient, please explain why it can\'t be genera
ted. \n4. Please use the most relevant table(s). \n5. If the question has be
en asked and answered before, please repeat the answer exactly as it was giv
en before. \n'}, {'role': 'user', 'content': ' \n    There are 3 tables: art
ists, albums and tracks, where albums and artists are linked by ArtistId, al
bums and tracks are linked by AlbumId,\n    Can you find the top 10 most pop
ular artists based on the number of tracks\n'}, {'role': 'assistant', 'conte
nt': 'SELECT a.Name, COUNT(t.TrackId) AS TrackCount\r\nFROM artists a\r\nJOI
N albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.A
lbumId\r\nGROUP BY a.Name\r\nORDER BY TrackCount DESC\r\nLIMIT 10'}, {'rol
e': 'user', 'content': '  \n    List all genres and the number of tracks in
each genre:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name, COUNT(t.Ge
nreId) AS TrackCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.Genr
eId\r\nGROUP BY g.Name'}, {'role': 'user', 'content': '  \n    List all albu
ms and their corresponding artist names  \n'}, {'role': 'assistant', 'conten
t': 'SELECT \n    a.Title,\n    ar.Name\nFROM albums AS a\nJOIN artists AS a
r ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': '  \n     Get a
ll playlists containing at least 10 tracks and the total duration of those t
racks:\n'}, {'role': 'assistant', 'content': 'SELECT p.Name, SUM(t.Milliseco
nds) AS TotalDuration\r\nFROM playlists p\r\nJOIN playlist_track pt ON p.Pla
ylistId = pt.PlaylistId\r\nJOIN tracks t ON pt.TrackId = t.TrackId\r\nGROUP
```

BY p.Name\r\nHAVING COUNT(*) >= 10'}, {'role': 'user', 'content': ' \n
Find the customer who bought the most albums in total quantity (across all i
nvoices): \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(i
i.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_item
s ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY Tot
alAlbumsPurchased DESC\r\nLIMIT 1'}, {'role': 'user', 'content': ' \n   Hi
nt: album quantity is found in invoice_items, \n   \n   Find the top 5 cus
tomers who bought the most albums in total quantity (across all invoice
s):\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(ii.Quant
ity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON
c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbums
Purchased DESC\r\nLIMIT 5 \n\n\n\n'}, {'role': 'user', 'content': ' \n
Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'a
ssistant', 'content': 'SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY Uni
tPrice DESC\r\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find all trac
ks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistan
t', 'content': "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%wha
t%'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLite da
tabase catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite
_master WHERE type='table'"}, {'role': 'user', 'content': 'How many customer
s are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM custom
ers'}, {'role': 'user', 'content': ' \n   Identify artists who have album
s with tracks appearing in multiple genres:\n\n\n'}]
Info: Ollama parameters:
model=gemma2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (Art
istId)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE IN
DEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"albums\"\r\n
(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NV
ARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY
(ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON
UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (Me
diaTypeId)\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK
_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE TABLE \"arti
sts\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n   Playl
istId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT
PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (Pla
ylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (Trac

kId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": " \n   There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks are linked by AlbumId,\n   Can you find the top 10 most popular artists based on the number of tracks\n"}, {"role": "assistant", "content": "SELECT a.Name, COUNT(t.TrackId) AS TrackCount\r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nORDER BY TrackCount DESC\r\nLIMIT 10"}, {"role": "user", "content": " \n   List all genres and the number of tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name, COUNT(t.GenreId) AS TrackCount\r\nFROM genres g\r\nJOIN tracks t ON g.GenreId = t.GenreId\r\nGROUP BY g.Name"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT \n   a.Title,\n   ar.Name\nFROM albums AS a\nJOIN artists AS ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Get all playlists containing at least 10 tracks and the total duration of those tracks:\n"}, {"role": "assistant", "content": "SELECT p.Name, SUM(t.Milliseconds) AS TotalDuration\r\nFROM playlists p\r\nJOIN playlist_trackpt ON p.PlaylistId = pt.PlaylistId\r\nJOIN tracks t ON pt.TrackId = t.TrackId\r\nGROUP BY p.Name\r\nHAVING COUNT(*) >= 10"}, {"role": "user", "content": " \n   Find the customer who bought the most albums in total quantity (across all invoices): \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 1"}, {"role": "user", "content": " \n   Hint: album quantity is found in invoice_items, \n   \n   Find the top 5 customers who bought the most albums in total quantity (across all invoices):\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) AS TotalAlbumsPurchased\r\nFROM customers c\r\nJOIN invoice_items ii ON c.CustomerId = ii.InvoiceId\r\nGROUP BY c.CustomerId\r\nORDER BY TotalAlbumsPurchased DESC\r\nLIMIT 5 \n\n\n\n"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\r\nFROM tracks\r\nORDER BY UnitPrice DESC\r\nLIMIT 5"}, {"role": "user", "content": " \n   Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \r\nFROM tracks\r\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM customers"}, {"role": "user", "content": " \n   Identify artists who have albums with tracks appearing in multiple genres:\n\n\n"}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:14:30.193048982Z', 'message': {'role': 'assistant', 'content': '```sql\r\nSELECT DISTINCT a.Name AS ArtistName \r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\nHAVING

```
COUNT(DISTINCT t.GenreId) > 1; \r\n```'}, 'done_reason': 'stop', 'done': Tru
e, 'total_duration': 80230431672, 'load_duration': 21627877, 'prompt_eval_co
unt': 1428, 'prompt_eval_duration': 61083512000, 'eval_count': 75, 'eval_dur
ation': 17881358000}
LLM Response: ```sql
SELECT DISTINCT a.Name AS ArtistName
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
HAVING COUNT(DISTINCT t.GenreId) > 1;
```
Info: Output from LLM: ```sql
SELECT DISTINCT a.Name AS ArtistName
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
HAVING COUNT(DISTINCT t.GenreId) > 1;
```
Extracted SQL: SELECT DISTINCT a.Name AS ArtistName
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
HAVING COUNT(DISTINCT t.GenreId) > 1
SELECT DISTINCT a.Name AS ArtistName
FROM artists a
JOIN albums al ON a.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY a.Name
HAVING COUNT(DISTINCT t.GenreId) > 1
               ArtistName
0           Amy Winehouse
1    Antônio Carlos Jobim
2              Audioslave
3    Battlestar Galactica
4            Eric Clapton
5           Faith No More
6            Foo Fighters
7            Gilberto Gil
8           Guns N' Roses
9                  Heroes
10           Iron Maiden
11             Jamiroquai
12          Lenny Kravitz
13                   Lost
14          Ozzy Osbourne
15              Pearl Jam
16                 R.E.M.
17  Red Hot Chili Peppers
18             The Office
19                     U2
20        Various Artists
Info: Ollama parameters:
model=gemma2:latest,
```

options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n   Identify artists who have albums with tracks appearing in multiple ge
nres:\n\n\n'\n\nThe DataFrame was produced using this query: SELECT DISTINCT
a.Name AS ArtistName \r\nFROM artists a\r\nJOIN albums al ON a.ArtistId = a
l.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nGROUP BY a.Name\r\n
HAVING COUNT(DISTINCT t.GenreId) > 1\n\nThe following is information about t
he resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n ArtistName
object\ndtype: object"}, {"role": "user", "content": "Can you generate the P
ython plotly code to chart the results of the dataframe? Assume the data is
in a pandas dataframe called 'df'. If there is only one value in the datafra
me, use an Indicator. Respond with only Python code. Do not answer with any
explanations -- just the code."}]
Info: Ollama Response:
{'model': 'gemma2:latest', 'created_at': '2024-08-01T19:14:54.518290578Z',
'message': {'role': 'assistant', 'content': '```python\nimport plotly.expres
s as px\n\nif df.shape[0] == 1:\n fig = px.indicator(df, value="ArtistNam
e", title="Artists with Albums Across Multiple Genres")\nelse:\n fig = px.b
ar(df, x="ArtistName", title="Artists with Albums Across Multiple Genres")\n
\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 24300679867,
'load_duration': 20795847, 'prompt_eval_count': 211, 'prompt_eval_duration':
7964300000, 'eval_count': 75, 'eval_duration': 16270251000}



Out[39]:
```
('SELECT DISTINCT a.Name AS ArtistName \r\nFROM artists a\r\nJOIN albums al
ON a.ArtistId = al.ArtistId\r\nJOIN tracks t ON al.AlbumId = t.AlbumId\r\nG
ROUP BY a.Name\r\nHAVING COUNT(DISTINCT t.GenreId) > 1',
               ArtistName
 0           Amy Winehouse
 1    Antônio Carlos Jobim
 2             Audioslave
 3    Battlestar Galactica
 4            Eric Clapton
 5           Faith No More
 6            Foo Fighters
 7            Gilberto Gil
 8           Guns N' Roses
 9                  Heroes
 10            Iron Maiden
 11             Jamiroquai
 12          Lenny Kravitz
 13                   Lost
 14          Ozzy Osbourne
 15              Pearl Jam
 16                 R.E.M.
 17  Red Hot Chili Peppers
 18             The Office
 19                     U2
 20        Various Artists,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'ArtistName=%{x}<br>count=%{y}<extra></extr
a>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Amy Winehouse', 'Antônio Carlos Jobim', 'Audios
lave',
                           'Battlestar Galactica', 'Eric Clapton', 'Faith N
o More', 'Foo Fighters',
                           'Gilberto Gil', "Guns N' Roses", 'Heroes', 'Iron
Maiden', 'Jamiroquai',
                           'Lenny Kravitz', 'Lost', 'Ozzy Osbourne', 'Pearl
Jam', 'R.E.M.',
                           'Red Hot Chili Peppers', 'The Office', 'U2', 'Va
rious Artists'],
                          dtype=object),
               'xaxis': 'x',
               'y': array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Artists with Albums Across Multiple Genr
```

```
es'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'ArtistName'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'count'}}}
 }))
```

Check completion time

In [4]:
```
from datetime import datetime
import os
hostname = os.uname().nodename
print("Hostname:", hostname)
```

```
Hostname: ducklover1
```

In [40]:
```
ts_stop = time()

elapsed_time = ts_stop - ts_start
print(f"test running on '{hostname}' with '{model_name}' LLM took : {elapsed
```

```
test running on 'ducklover1' with 'gemma2' LLM took : 2267.60 sec
```

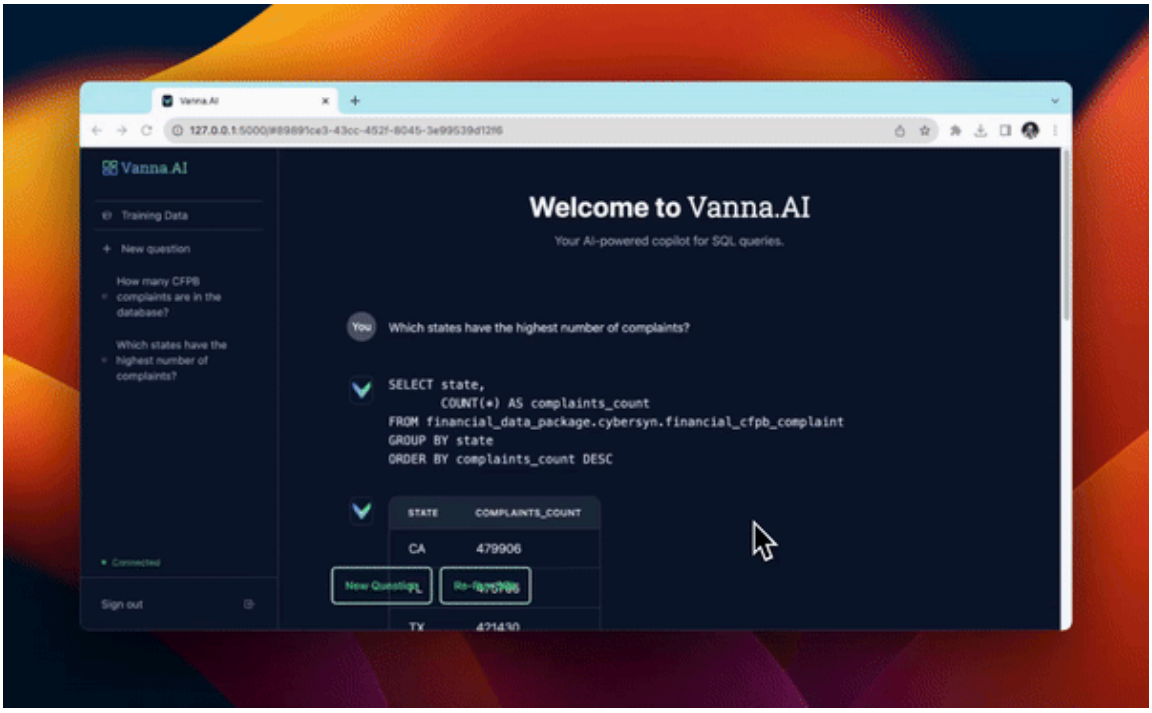
In [5]:
```
print(f"[{datetime.now()}] test on '{hostname}' with '{model_name}' LLM took
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[5], line 1
----> 1 print(f"[{datetime.now()}] test on '{hostname}' with '{model_name}'
LLM took : {elapsed_time:.2f} sec")

NameError: name 'model_name' is not defined
```

test running on 'ducklover1' with 'gemma2' LLM took : 2267.60 sec

Launch the User Interface



from vanna.flask import VannaFlaskApp app = VannaFlaskApp(vn) app.run()

Next Steps

Using Vanna via Jupyter notebooks is great for getting started but check out additional customizable interfaces like the

- Streamlit app
- Flask app
- Slackbot