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='^Number of requested results')
# warnings.filterwarnings('ignore', category=DeprecationWarning, message=re.escape(r'^Some regex pattern')

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 = '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", "documentation"]):
    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 DDL script.

    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 brackets
    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 null")
```

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 terminology or definitions.
    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" (SupportRepId)
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" (InvoiceId)
Adding ddl: CREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)
Adding ddl: CREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)
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';
```

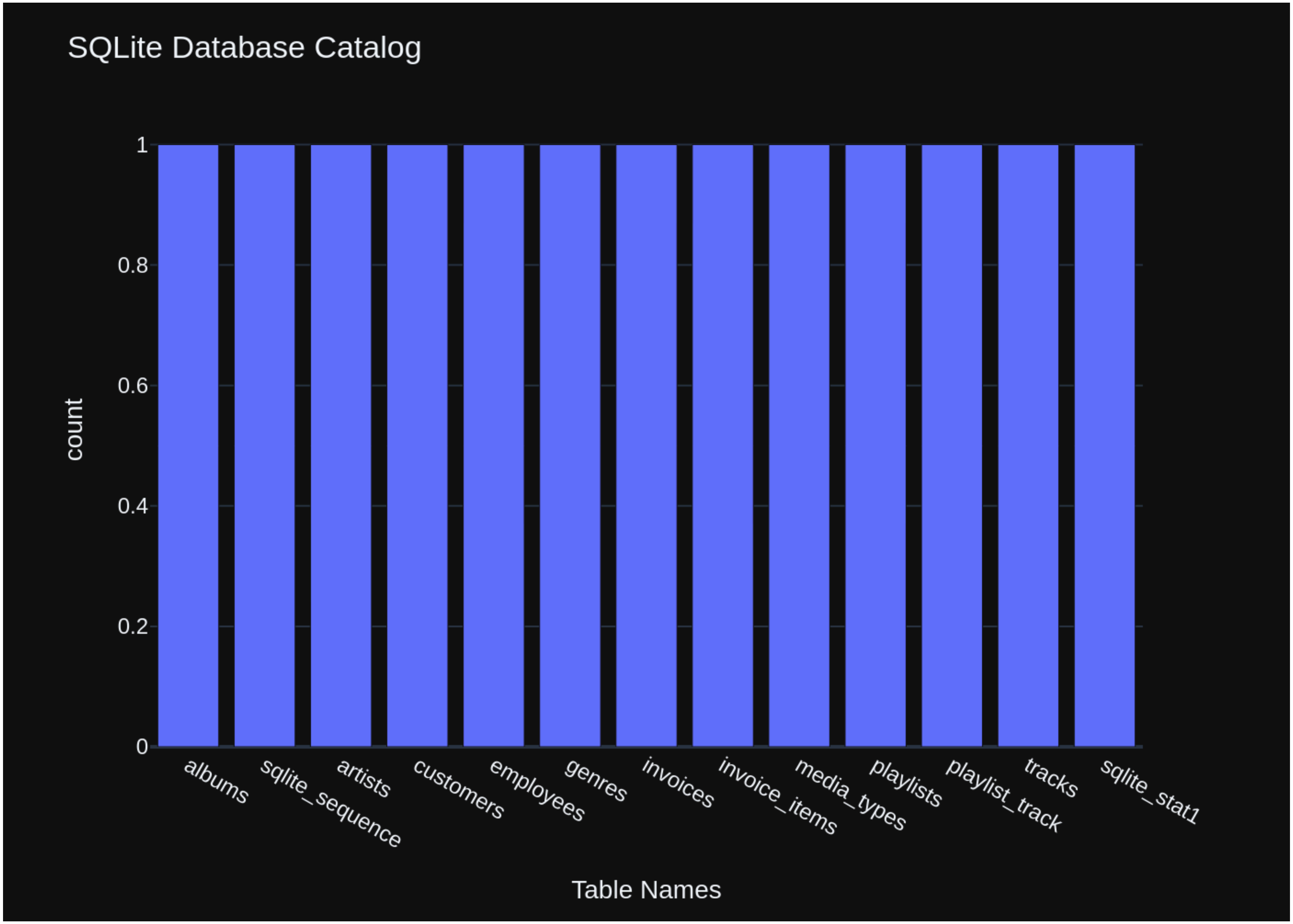
In [16]:
```
vn.ask(question="Can you list all tables in the SQLite database catalog?")
```

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

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE TABLE sqlite_sequence(na
me,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 "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOI
NCREMENT 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 "med
ia_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "media_type
s"\r\n(\r\n    MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCR
EATE TABLE "artists"\r\n(\r\n    ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR
(120)\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" (InvoiceI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (Track
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\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 (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELET
E NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELE
TE NO ACTION ON UPDATE NO ACTION\r\n)\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 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 p
rovided 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 particul
ar column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend t
he 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). \n5. If the question has been ask
ed and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'co
ntent': 'Can you list all tables in the SQLite database catalog?'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE TABLE sqlite_sequence(na
me,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 AUTOINCREM
ENT 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) REFERENC
ES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFER
ENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeI
d) REFERENCES \"media_types\" (MediaTypeId) \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 NVARC
HAR(120)\r\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n    ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMA
RY 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) REFERE
NCES \"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 \"play
list_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    Arti
stId INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE N
O ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means orde
r\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query w
ithout any explanations for the question. \n2. If the provided context is almost sufficient but requires kn
owledge 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 provi
ded context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant ta
ble(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?"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:50:49.794133924Z', 'message': {'role': 'assistant',
'content': "Here is a SQL query to get the list of tables in the SQLite database:\n\n```sql\nSELECT name FR
OM sqlite_master WHERE type='table';\n```\n\nThis will return a list of all the tables in the database."},
'done_reason': 'stop', 'done': True, 'total_duration': 39553622959, 'load_duration': 2738362209, 'prompt_ev
al_count': 752, 'prompt_eval_duration': 28994934000, 'eval_count': 45, 'eval_duration': 7777443000}
Here is a SQL query to get the list of tables in the SQLite database:

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

This will return a list of all the tables in the database.
~~~

~~~
Output from LLM: Here is a SQL query to get the list of tables in the SQLite database:

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

This will return a list of all the tables in the database.
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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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: 'Can you list all tables in the SQLite database catalog?'\n\nThe DataFrame was produced using this query: SELECT name FROM sqlite_master WHERE type='table'\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n name    object\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."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:51:02.860492597Z', 'message': {'role': 'assistant', 'content': '```\nimport plotly.express as px\n\nfig = px.bar(x=\'name\', y=None, data_frame=df)\n\nfig.update_layout(title_text="SQLite Database Catalog", xaxis_title_text="Table Names")\n\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 13033343060, 'load_duration': 42110129, 'prompt_eval_count': 146, 'prompt_eval_duration': 4516172000, 'eval_count': 48, 'eval_duration': 8430732000}
~~~


SQLite Database Catalog
count
1
0.8
0.6
0.4
0.2
0
albums
sqlite_sequence
artists
customers
employees
genres
invoices
invoice_items
media_types
playlists
playlist_track
tracks
sqlite_stat1
Table Names

```
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': [{'alignmentgroup': 'True',
                         'hovertemplate': 'name=%{x}<br>count=%{y}<extra></extra>',
                         'legendgroup': '',
                         'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
                         'name': '',
                         'offsetgroup': '',
                         'orientation': 'v',
                         'showlegend': False,
                         'textposition': 'auto',
                         'type': 'bar',
                         'x': array(['albums', 'sqlite_sequence', 'artists', 'customers', 'employees',
                                     'genres', 'invoices', 'invoice_items', 'media_types', 'playlists',
                                     'playlist_track', 'tracks', 'sqlite_stat1'], dtype=object),
                         'xaxis': 'x',
                         'y': array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]),
                         'yaxis': 'y'}],
               'layout': {'barmode': 'relative',
                          'legend': {'tracegroupgap': 0},
                          'margin': {'t': 60},
                          'template': '...',
                          'title': {'text': 'SQLite Database Catalog'},
                          'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Table Names'}},
                          'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'count'}}}
           }))
```

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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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\t
ON 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 P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  N
OT NULL,\r\n    Company NVARCHAR(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 (Sup
portRepId) 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 AUTOINCREMENT NOT NULL,\r\n    LastName NVA
RCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo I
NTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARC
HAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Pho
ne NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFEREN
CES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_se
quence(name,seq)\n\nCREATE TABLE "playlists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\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    A
rtistId 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 NUL
L,\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 UPDA
TE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE 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 mea
ns 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 requ
ires knowledge of a specific string in a particular column, please generate an intermediate SQL query to fi
nd the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If th
e provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most rel
evant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly a
s it was given before. \n'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite database ca
talog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 
'user', 'content': "which table stores customer's orders"}]
```

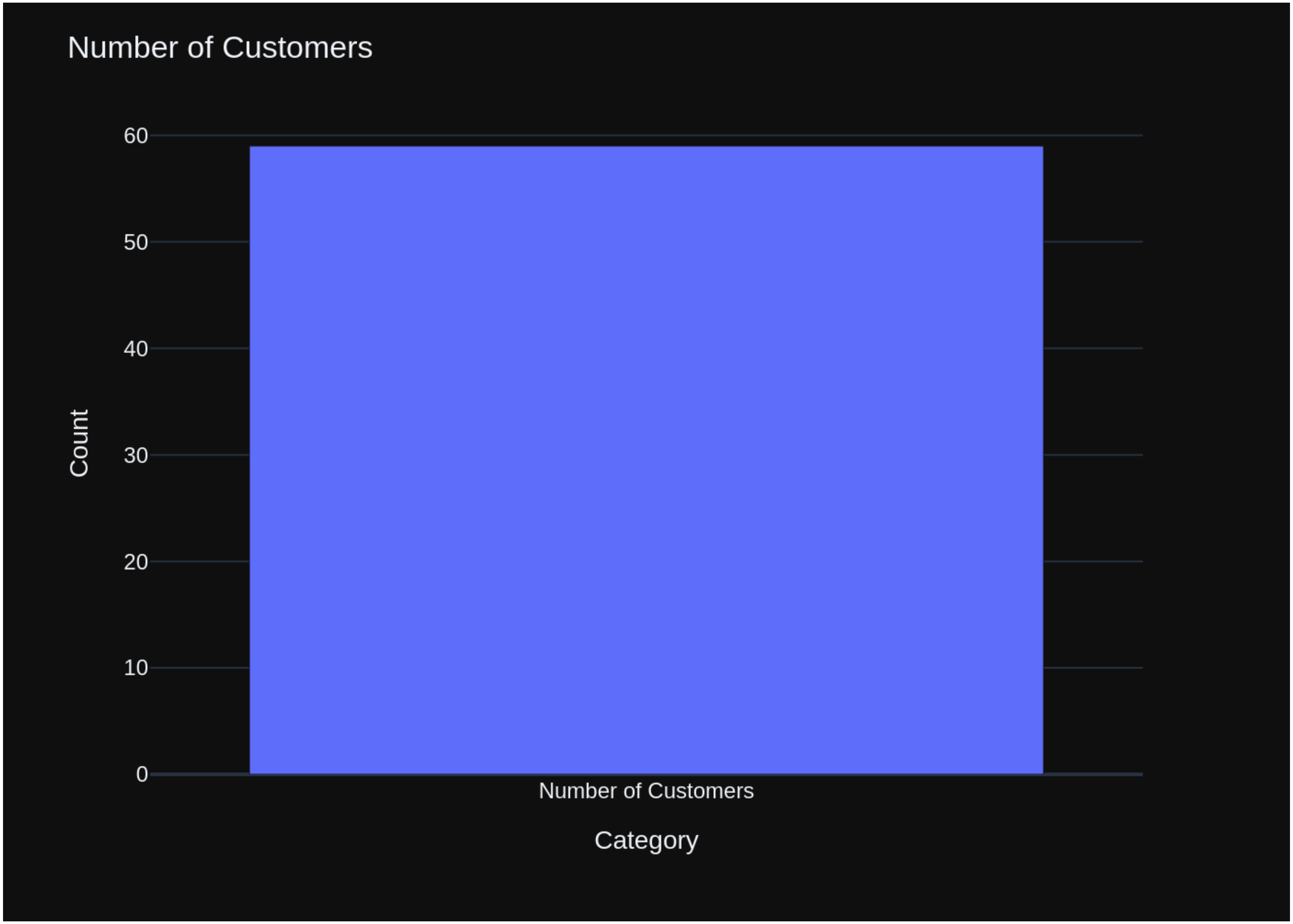
```
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the
question. Your response should ONLY be based on the given context and follow the response guidelines and fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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 NUL
L,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NO
T NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceI
d) \r\n\t\tON DELETE NO ACTION 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 \"customers\"\r\n(\r\n    Custom
erId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName N
VARCHAR(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    PostalCode NVARCHAR(10),\r\n    Phone NV
ARCHAR(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 A
CTION\r\n)\n\nCREATE TABLE \"employees\"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\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 DATETIME,\r\n    Address NVARCHAR
(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode N
VARCHAR(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 PRI
MARY 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 NV
ARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES \"artist
s\" (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\" (TrackI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n    MediaTy
peId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n\n===Additional Context 
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is s
ufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provide
d context is almost sufficient but requires knowledge of a specific string in a particular column, please g
enerate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a com
ment saying intermediate_sql \n3. 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 has been asked and answered befo
re, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": "Can you lis
t all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_m
aster WHERE type='table'"}, {"role": "user", "content": "which table stores customer's orders"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:52:07.373736157Z', 'message': {'role': 'assistant',
'content': 'The "invoices" table stores customer\'s orders.'}, 'done_reason': 'stop', 'done': True, 'total_
duration': 64097673269, 'load_duration': 543697, 'prompt_eval_count': 965, 'prompt_eval_duration': 62160009
000, 'eval_count': 11, 'eval_duration': 1803915000}
The "invoices" table stores customer's orders.
The "invoices" table stores customer's orders.
Couldn't run sql:  Execution failed on sql 'The "invoices" table stores customer's orders.': near "The": sy
ntax 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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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 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 NVAR
CHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Ph
one NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGE
R,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "invoic
e_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NO
T 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 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 INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TAB
LE "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_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(3
0),\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 N
VARCHAR(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 "playlists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name N
VARCHAR(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 explan
ations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a spec
ific 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 in
sufficient, 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': 'ass
istant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': 'How
many customers are there'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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 TABLE \"customers
\"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NU
LL,\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(1
0),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    Support
RepId 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_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREAT
E TABLE \"invoice_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Invo
iceId 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 D
ELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX 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    FOREIGN KEY (ArtistId) REFERENCES
\"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLine
TrackId 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 NUL
L,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIM
E,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHA
R(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVA
RCHAR(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 \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCRE
MENT NOT NULL,\r\n    Name NVARCHAR(120)\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 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 t
o 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 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": "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"}]
Ollama Response:
```

```
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:53:01.794844713Z', 'message': {'role': 'assistant',
'content': 'SELECT COUNT(*) FROM "customers";'}, 'done_reason': 'stop', 'done': True, 'total_duration': 543
73145119, 'load_duration': 815123, 'prompt_eval_count': 739, 'prompt_eval_duration': 52937829000, 'eval_cou
nt': 8, 'eval_duration': 1297937000}
SELECT COUNT(*) FROM "customers";
Output from LLM: SELECT COUNT(*) FROM "customers";
Extracted SQL: SELECT COUNT(*) FROM "customers"
SELECT COUNT(*) FROM "customers"
   COUNT(*)
0        59
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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: 'How many customers are there'\n\nThe DataFrame was produced usin
g this query: SELECT COUNT(*) FROM \"customers\"\n\nThe following is information about the resulting pandas
DataFrame 'df': \nRunning df.dtypes gives:\n COUNT(*)    int64\ndtype: object"}, {"role": "user", "conten
t": "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 o
nly Python code. Do not answer with any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:53:21.695077082Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nimport numpy as np\n\nfig = px.bar(x=['Number of Customers'],
y=df['COUNT(*)'].values)\nfig.update_layout(title='Number of Customers', xaxis_title='Category', yaxis_titl
e='Count')\nfig.show()\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 19875231509, 'load_du
ration': 41421195, 'prompt_eval_count': 140, 'prompt_eval_duration': 9696742000, 'eval_count': 59, 'eval_du
ration': 10091025000}
```


Number of Customers
Count
60
50
40
30
20
10
0
Number of Customers
Category

Out[18]:

```
('SELECT COUNT(*) FROM "customers"',
    COUNT(*)
 0        59,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'x=%{x}<br>y=%{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Number of Customers'], dtype=object),
               'xaxis': 'x',
               'y': array([59]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'title': {'text': 'Number of Customers'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Category'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'Count'}}}
 }))
```

In [ ]:

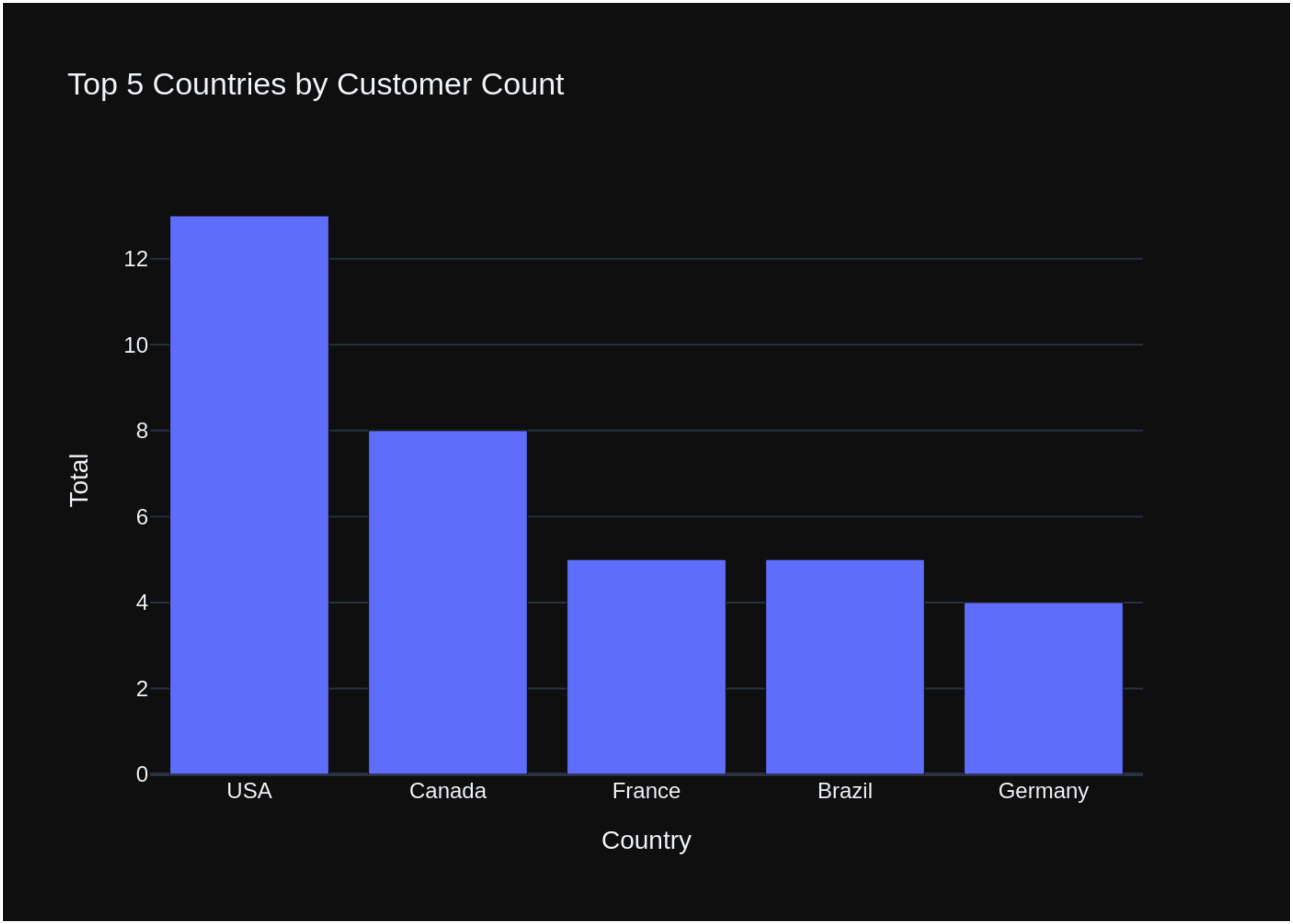
In [19]:

```
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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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    Firs
tName 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 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    InvoiceLineI
d INTEGER PRIMARY KEY AUTOINCREMENT 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 (I
nvoiceId) 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 NVARCHA
R(120)\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employe
es"\r\n(\r\n    EmployeeId INTEGER 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    Bi
rthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    S
tate NVARCHAR(40),\r\n    Country 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 "employee
s" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n    A
lbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId IN
TEGER  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    FO
REIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIM
ARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTy
peId 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) REFE
RENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REF
ERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeI
d) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Addi
tional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provid
ed 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 co
lumn, 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 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"'}, {'ro
le': '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': 'what are the
top 5 countries that customers come from?'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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    InvoiceL
ineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGE
R  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KE
Y (InvoiceId) REFERENCES \"invoices\" (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 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    N
ame NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE T
ABLE \"employees\"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName NVARC
HAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INT
EGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHA
R(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\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n    PlaylistId INTEG
ER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistI
d, TrackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACT
ION 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    Album
Id 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 Guide
lines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanation
s 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 th
at column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insuffi
cient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the qu
estion 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": "Can you list all tables in the SQLite database catal
og?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "use
r", "content": "what are the top 5 countries that customers come from?"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:54:38.733921696Z', 'message': {'role': 'assistant',
'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIM
IT 5'}, 'done_reason': 'stop', 'done': True, 'total_duration': 76937355145, 'load_duration': 623551, 'promp
t_eval_count': 1133, 'prompt_eval_duration': 72351087000, 'eval_count': 25, 'eval_duration': 4408571000}
SELECT Country, COUNT(*) as Total
FROM "customers"
GROUP BY Country
ORDER BY Total DESC
LIMIT 5
SELECT Country, COUNT(*) as Total
FROM "customers"
GROUP BY Country
ORDER BY Total DESC
LIMIT 5
   Country  Total
0      USA     13
1   Canada      8
2   France      5
3   Brazil      5
4  Germany      4
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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: 'what are the top 5 countries that customers come from?'\n\nThe D
ataFrame was produced using this query: SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Cou
ntry\nORDER BY Total DESC\nLIMIT 5\n\nThe following is information about the resulting pandas DataFrame 'd
f': \nRunning df.dtypes gives:\n Country    object\nTotal       int64\ndtype: object"}, {"role": "user", "c
ontent": "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 w
ith only Python code. Do not answer with any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:54:57.24984521Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nfig = px.bar(df, x='Country', y='Total', title='Top 5 Countri
es by Customer Count')\nfig.show()\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 184951877
22, 'load_duration': 42637805, 'prompt_eval_count': 167, 'prompt_eval_duration': 11718749000, 'eval_count':
39, 'eval_duration': 6643878000}
```


Top 5 Countries by Customer Count
Total
12
10
8
6
4
2
0
USA
Canada
France
Brazil
Germany
Country

```
Out[19]: ('SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5',
             Country  Total
          0      USA     13
          1   Canada      8
          2   France      5
          3   Brazil      5
          4  Germany      4,
          Figure({
              'data': [{'alignmentgroup': 'True',
                        'hovertemplate': 'Country=%{x}<br>Total=%{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 by Customer Count'},
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Country'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'Total'}}}
          }))
```

More SQL questions

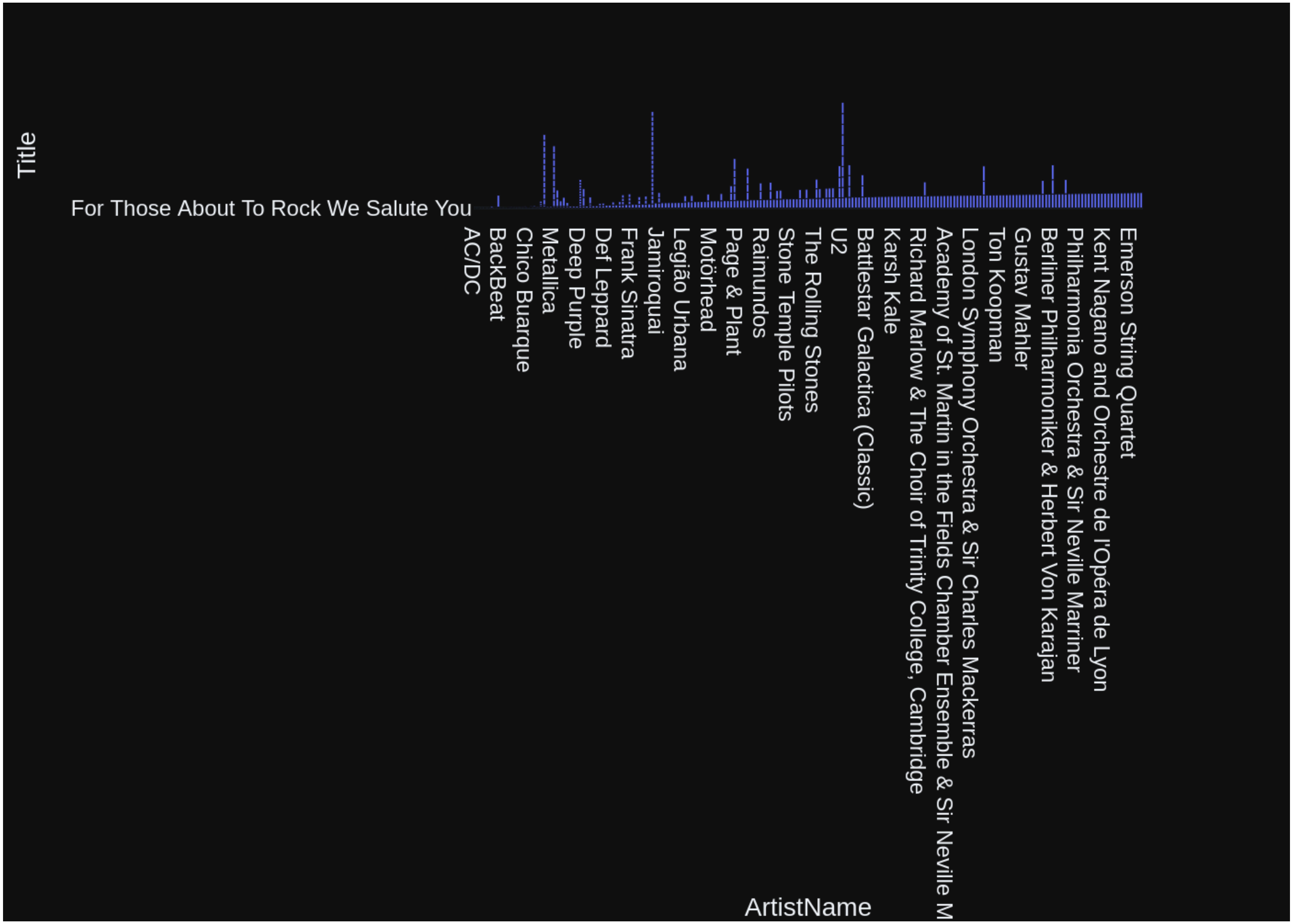
see `sample-sql-queries-sqlite-chinook.ipynb`

```
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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE TABLE "alb
ums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NUL
L,\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 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) REFERENC
ES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFEREN
CES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) RE
FERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX I
FK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "artists"\r\n(\r\n    ArtistId INTEGER PRIMARY KEY AU
TOINCREMENT 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 "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\n
CREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\n\n===Additional Context \n\nIn the chinook d
atabase invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please ge
nerate 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 intermedi
ate SQL query to find the distinct 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. Plea
se use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat t
he 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='ta
ble'"}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'as
sistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total
DESC\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistant', 'conten
t': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': '  \n    List all albums and their cor
responding artist names  \n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\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 \"tracks\"\r\n(\r\n    TrackId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    Medi
aTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEG
ER  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 (GenreI
d) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Med
iaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n
CREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n(\r\n    ArtistId INTE
GER 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 TAB
LE \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(12
0)\r\n)\n\nCREATE TABLE \"genres\"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    N
ame NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\n\n===Additional
Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided cont
ext 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, p
lease generate an intermediate SQL query to find the distinct strings in that column. Prepend the query wit
h a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it ca
n't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answe
red 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 s
qlite_master WHERE type='table'"}, {"role": "user", "content": "what are the top 5 countries that customers
come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROU
P BY Country\nORDER BY Total DESC\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"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:55:28.701562088Z', 'message': {'role': 'assistant',
'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.Arti
stId = ar.ArtistId'}, 'done_reason': 'stop', 'done': True, 'total_duration': 31352733115, 'load_duration':
1606219, 'prompt_eval_count': 682, 'prompt_eval_duration': 24628439000, 'eval_count': 37, 'eval_duration':
6490718000}
SELECT a.Title, a.ArtistId, ar.Name AS ArtistName
FROM "albums" a
JOIN "artists" ar ON a.ArtistId = ar.ArtistId
SELECT a.Title, a.ArtistId, ar.Name AS ArtistName
FROM "albums" a
JOIN "artists" ar ON a.ArtistId = ar.ArtistId
                                        Title  ArtistId  \
0                For Those About To Rock We Salute You         1
1                                    Balls to the Wall         2
2                                    Restless and Wild         2
3                                    Let There Be Rock         1
```

```
4                                             Big Ones         3
..                                                 ...       ...
342                             Respighi:Pines of Rome       226
343  Schubert: The Late String Quartets & String Qu...       272
344                                Monteverdi: L'Orfeo       273
345                              Mozart: Chamber Music       274
346  Koyaanisqatsi (Soundtrack from the Motion Pict...       275

                                            ArtistName
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 3 columns]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 albums and their corresponding artist names
\n'\n\nThe DataFrame was produced using this query: SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM
\"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId\n\nThe following is information about the res
ulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Title         object\nArtistId       int64\nArti
stName    object\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 explana
tions -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:55:46.994335493Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nfig = px.bar(df, x='ArtistName', y='Title')\nfig.show()\n``
`"}, 'done_reason': 'stop', 'done': True, 'total_duration': 18267578859, 'load_duration': 42478531, 'prompt
_eval_count': 185, 'prompt_eval_duration': 13024724000, 'eval_count': 30, 'eval_duration': 5153994000}
```


Title
For Those About To Rock We Salute You
AC/DC
BackBeat
Chico Buarque
Metallica
Deep Purple
Def Leppard
Frank Sinatra
Jamiroquai
Legião Urbana
Motörhead
Page & Plant
Raimundos
Stone Temple Pilots
The Rolling Stones
U2
Battlestar Galactica (Classic)
Karsh Kale
Richard Marlow & The Choir of Trinity College, Cambridge
Academy of St. Martin in the Fields Chamber Ensemble & Sir Neville M
London Symphony Orchestra & Sir Charles Mackerras
Ton Koopman
Gustav Mahler
Berliner Philharmoniker & Herbert Von Karajan
Philharmonia Orchestra & Sir Neville Marriner
Kent Nagano and Orchestre de l'Opéra de Lyon
Emerson String Quartet
ArtistName

```
Out[20]: ('SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.ArtistId = a
         r.ArtistId',
                                                           Title  ArtistId  \
          0                For Those About To Rock We Salute You         1
          1                                    Balls to the Wall         2
          2                                    Restless and Wild         2
          3                                    Let There Be Rock         1
          4                                             Big Ones         3
          ..                                                 ...       ...
          342                             Respighi:Pines of Rome       226
          343  Schubert: The Late String Quartets & String Qu...       272
          344                                Monteverdi: L'Orfeo       273
          345                              Mozart: Chamber Music       274
          346  Koyaanisqatsi (Soundtrack from the Motion Pict...       275

                                                      ArtistName
          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 3 columns],
          Figure({
              'data': [{'alignmentgroup': 'True',
                        'hovertemplate': 'ArtistName=%{x}<br>Title=%{y}<extra></extra>',
                        'legendgroup': '',
                        'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
                        'name': '',
                        'offsetgroup': '',
                        'orientation': 'v',
                        'showlegend': False,
                        'textposition': 'auto',
                        'type': 'bar',
                        'x': array(['AC/DC', 'Accept', 'Accept', ...,
                                    'C. Monteverdi, Nigel Rogers - Chiaroscuro; London Baroque; London Cornett & Sa
         ckbu',
                                    'Nash Ensemble', 'Philip Glass Ensemble'], dtype=object),
                        'xaxis': 'x',
                        'y': array(['For Those About To Rock We Salute You', 'Balls to the Wall',
                                    'Restless and Wild', ..., "Monteverdi: L'Orfeo",
                                    'Mozart: Chamber Music',
                                    'Koyaanisqatsi (Soundtrack from the Motion Picture)'], dtype=object),
                        'yaxis': 'y'}],
              'layout': {'barmode': 'relative',
                         'legend': {'tracegroupgap': 0},
                         'margin': {'t': 60},
                         'template': '...',
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'ArtistName'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'Title'}}}
          }))
```

In [21]:

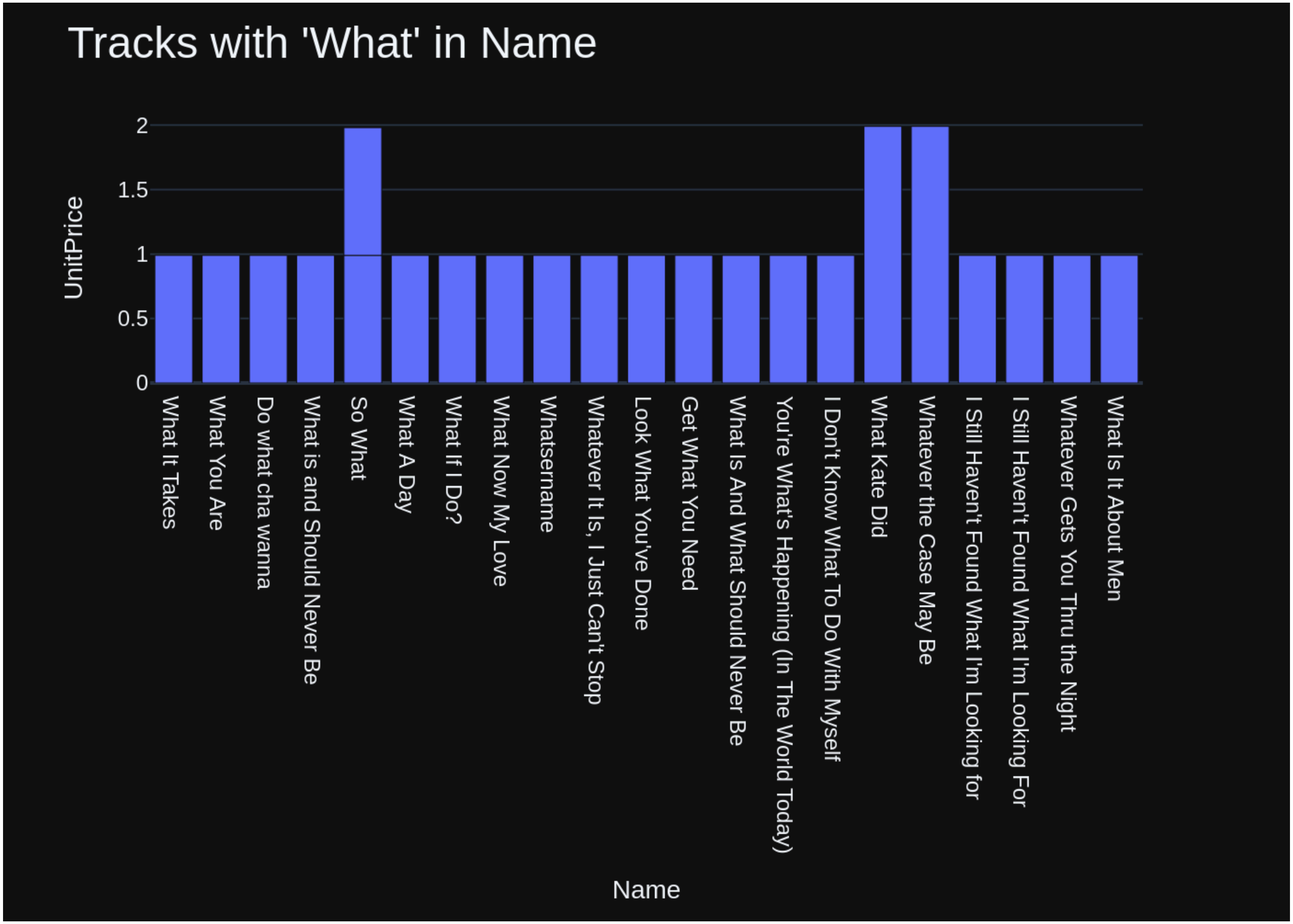
```
question = """
    Find all tracks with a name containing "What" (case-insensitive)
"""

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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX IFK_Pl
aylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTyp
eId 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) REFE
RENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REF
ERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeI
d) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IN
DEX 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  NOT NUL
L,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) R
EFERENCES "playlists" (PlaylistId) \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\nCREATE INDEX IF
K_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)
\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 NUL
L,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means orde
r\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query w
ithout any explanations for the question. \n2. If the provided context is almost sufficient but requires kn
owledge 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 provi
ded context is insufficient, please 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 repeat the answer exactly as it wa
s given before. \n'}, {'role': 'user', 'content': '  \n    List all albums and their corresponding artist n
ames  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albu
ms" a\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': 'Can you list all table
s 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 customers come from?'}, {'ro
le': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER
BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistan
t', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': '  \n    Find all tracks wi
th a name containing "What" (case-insensitive)\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE INDEX IFK_
PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\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    Med
iaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTE
GER  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 (GenreI
d) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Med
iaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n
CREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (M
ediaTypeId)\n\nCREATE TABLE \"playlist_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\" (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_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCREATE INDEX IFK_AlbumArtistId ON
\"albums\" (ArtistId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"genres\"\r\n(\r\n    GenreId INTEGER PRIMARY K
EY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook dat
abase invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please gene
rate 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 intermediat
e SQL query to find the distinct 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 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 albums and their
corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.ArtistId, ar.Name AS 
ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "conten
t": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT na
me FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": "what are the top 5 countries that 
customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"custome
rs\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "How many customers are
there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user", "conten
t": "  \n    Find all tracks with a name containing \"What\" (case-insensitive)\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:56:44.283505264Z', 'message': {'role': 'assistant',
'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, 'done_reason': 'stop', 'done': True, 'to
tal_duration': 57138812374, 'load_duration': 667428, 'prompt_eval_count': 768, 'prompt_eval_duration': 5460
0317000, 'eval_count': 13, 'eval_duration': 2228555000}
SELECT *
FROM "tracks"
WHERE Name LIKE '%what%'
SELECT *
FROM "tracks"
WHERE 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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 all tracks with a name containing \"What\" (case-in
sensitive)\n'\n\nThe DataFrame was produced using this query: SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%
what%'\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes give
s:\n TrackId           int64\nName             object\nAlbumId           int64\nMediaTypeId       int64\nGe
nreId           int64\nComposer         object\nMilliseconds      int64\nBytes             int64\nUnitPrice
float64\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 -- j
ust the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:57:24.638186928Z', 'message': {'role': 'assistant',
'content': '```\nimport plotly.express as px\nimport numpy as np\n\nfig = px.bar(df, x=\'Name\', y=\'UnitPr
ice\')\nfig.update_layout(title_text="Tracks with \'What\' in Name", title_font_size=24)\nfig.show()\n\nif
df.shape[0] == 1:\n    fig = go.Figure(data=[go.Indicator(\n        mode="number+delta",\n        value=df
[\'UnitPrice\'].values[0],\n        delta={\'reference\': df[\'UnitPrice\'].mean(), \'value\': 0, \'label
\': \'Average\'}\n    )])\n    fig.update_layout(title_text="Track with \'What\' in Name", title_font_size=
24)\n    fig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 40327792688, 'load_durat
ion': 1087726, 'prompt_eval_count': 200, 'prompt_eval_duration': 13972676000, 'eval_count': 147, 'eval_dura
tion': 26218790000}
```


Tracks with 'What' in Name
UnitPrice
2
1.5
1
0.5
0
What It Takes
What You Are
Do what cha wanna
What is and Should Never Be
So What
What A Day
What If I Do?
What Now My Love
Whatsername
Whatever It Is, I Just Can't Stop
Look What You've Done
Get What You Need
What Is And What Should Never Be
You're What's Happening (In The World Today)
I Don't Know What To Do With Myself
What Kate Did
Whatever the Case May Be
I Still Haven't Found What I'm Looking for
I Still Haven't Found What I'm Looking For
Whatever Gets You Thru the Night
What Is It About Men
Name

```
Out[21]:  ('SELECT *\nFROM "tracks"\nWHERE 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  ,
Figure({
    'data': [{'alignmentgroup': 'True',
              'hovertemplate': 'Name=%{x}<br>UnitPrice=%{y}<extra></extra>',
              'legendgroup': '',
              'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
              'name': '',
              'offsetgroup': '',
              'orientation': 'v',
              'showlegend': False,
              'textposition': 'auto',
              'type': 'bar',
              'x': array(['What It Takes', 'What You Are', 'Do what cha wanna',
                          'What is and Should Never Be', 'So What', 'What A Day', 'What If I Do?',
                          'What Now My Love', 'Whatsername', "Whatever It Is, I Just Can't Stop",
                          "Look What You've Done", 'Get What You Need',
                          'What Is And What Should Never Be',
                          "You're What's Happening (In The World Today)", 'So What',
                          "I Don't Know What To Do With Myself", 'What Kate Did',
                          'Whatever the Case May Be',
                          "I Still Haven't Found What I'm Looking for",
                          "I Still Haven't Found What I'm Looking For",
                          'Whatever Gets You Thru the Night', 'What Is It About Men'],
                         dtype=object),
              'xaxis': 'x',
              'y': array([0.99, 0.99, 0.99, 0.99, 0.99, 0.99, 0.99, 0.99, 0.99, 0.99, 0.99, 0.99,
                          0.99, 0.99, 0.99, 0.99, 1.99, 1.99, 0.99, 0.99, 0.99, 0.99]),
              'yaxis': 'y'}],
    'layout': {'barmode': 'relative',
               'legend': {'tracegroupgap': 0},
               'margin': {'t': 60},
               'template': '...',
               'title': {'font': {'size': 24}, 'text': "Tracks with 'What' in Name"},
               'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Name'}},
               'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'UnitPrice'}}}
}))
```

In [22]:

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

vn.ask(question=question)
```

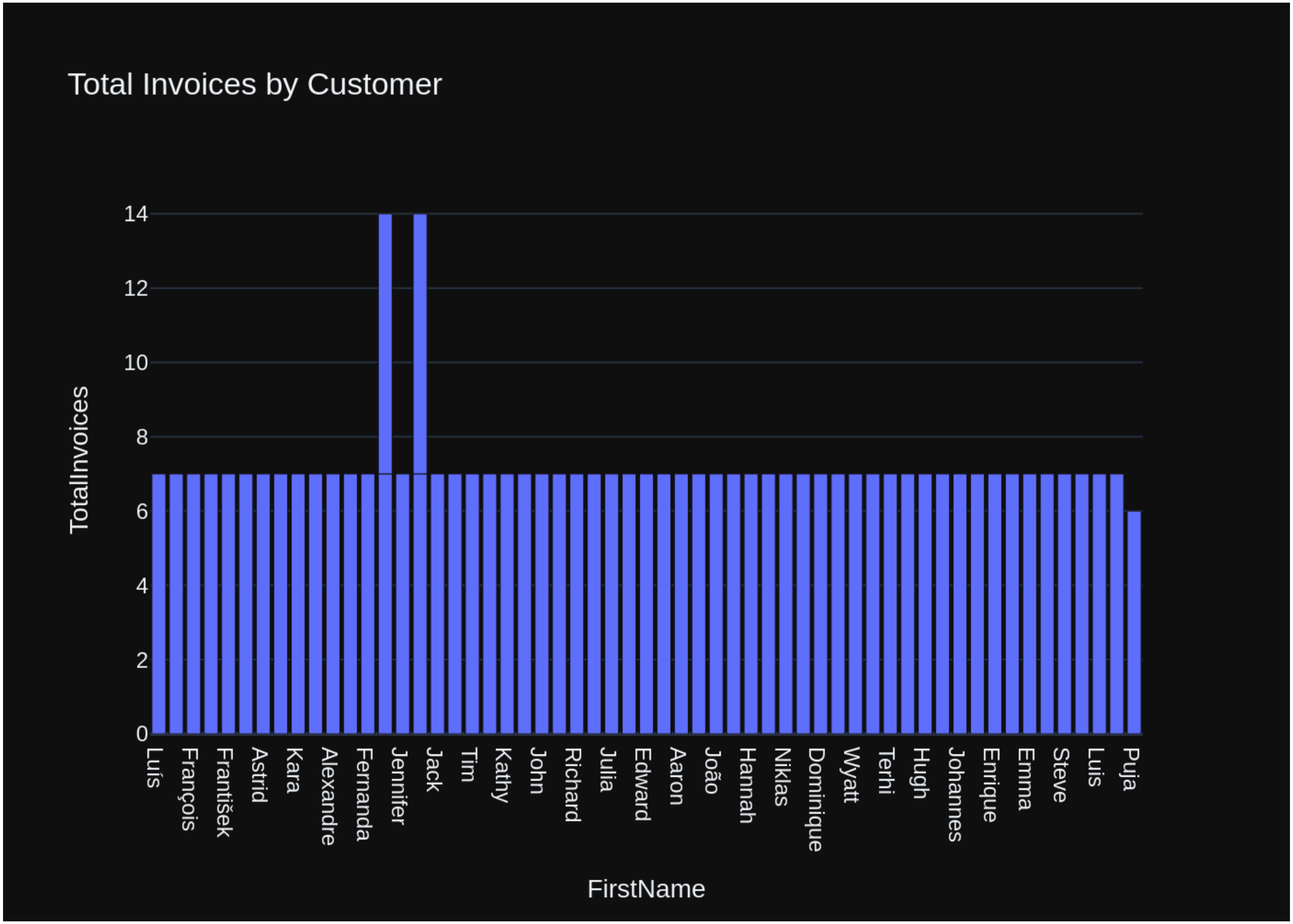
```
Number of requested results 10 is greater than number of elements in index 5, updating n_results = 5
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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 KE
Y AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    Unit
Price 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) REFERE
NCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLin
eTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NUL
L,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHA
R(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax N
VARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportR
epId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IN
DEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employees"\r\n(\r\n    Employee
Id INTEGER PRIMARY KEY AUTOINCREMENT 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 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 DEL
ETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\n
CREATE 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 NUMER
IC(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 DEL
ETE 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 que
ry without any explanations for the question. \n2. If the provided context is almost sufficient but require
s 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 p
rovided context is insufficient, 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 i
t was given before. \n'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistan
t', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': 'what are the top 5 countri
es that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM
```

```
"customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    List al
l albums and their corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.Ar
tistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" 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 *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, {'role': 'user', 'content':
'Can you list all tables in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT name F
ROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': '  \n    Get the total number of invoic
es for each customer\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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_InvoiceLineInv
oiceId 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 (Tr
ackId) 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 NVARCH
AR(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(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREI
GN 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 NUL
L,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    Bir
thDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    St
ate 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 NUL
L,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Byt
es INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES \"albums\" (A
lbumId) \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 \"me
dia_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 s
ufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provide
d context is almost sufficient but requires knowledge of a specific string in a particular column, please g
enerate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a com
ment saying intermediate_sql \n3. 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 has been asked and answered befo
re, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": "How many cu
stomers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "use
r", "content": "what are the top 5 countries that customers come from?"}, {"role": "assistant", "content":
"SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"},
{"role": "user", "content": "  \n    List all albums and their corresponding artist names  \n"}, {"role":
"assistant", "content": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artis
ts\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "  \n    Find all tracks with a name con
taining \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE
Name LIKE '%what%'"}, {"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": "use
r", "content": "  \n    Get the total number of invoices for each customer\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:58:55.286715135Z', 'message': {'role': 'assistant',
'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "c
ustomers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, 'done_reaso
n': 'stop', 'done': True, 'total_duration': 90516022933, 'load_duration': 743310, 'prompt_eval_count': 115
1, 'prompt_eval_duration': 80941845000, 'eval_count': 50, 'eval_duration': 9170186000}
SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId, c.FirstName, c.LastName
SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId, c.FirstName, c.LastName
    FirstName      LastName  TotalInvoices
0        Luís     Gonçalves              7
1      Leonie        Köhler              7
2    François      Tremblay              7
3       Bjørn        Hansen              7
4   František   Wichterlová              7
5      Helena          Holý              7
```

```
6      Astrid        Gruber              7
7        Daan       Peeters              7
8        Kara       Nielsen              7
9     Eduardo       Martins              7
10  Alexandre         Rocha              7
11    Roberto       Almeida              7
12   Fernanda         Ramos              7
13       Mark       Philips              7
14   Jennifer      Peterson              7
15      Frank        Harris              7
16       Jack         Smith              7
17   Michelle        Brooks              7
18        Tim         Goyer              7
19        Dan        Miller              7
20      Kathy         Chase              7
21    Heather       Leacock              7
22       John        Gordon              7
23      Frank       Ralston              7
24     Victor       Stevens              7
25    Richard    Cunningham              7
26    Patrick          Gray              7
27      Julia       Barnett              7
28     Robert         Brown              7
29     Edward       Francis              7
30     Martha          Silk              7
31      Aaron      Mitchell              7
32      Ellie      Sullivan              7
33       João     Fernandes              7
34   Madalena       Sampaio              7
35     Hannah     Schneider              7
36       Fynn    Zimmermann              7
37     Niklas      Schröder              7
38    Camille       Bernard              7
39  Dominique      Lefebvre              7
40       Marc        Dubois              7
41      Wyatt        Girard              7
42   Isabelle       Mercier              7
43      Terhi    Hämäläinen              7
44   Ladislav        Kovács              7
45       Hugh      O'Reilly              7
46      Lucas       Mancini              7
47   Johannes  Van der Berg              7
```

```
48  Stanisław        Wójcik              7
49    Enrique         Muñoz              7
50     Joakim     Johansson              7
51       Emma         Jones              7
52       Phil        Hughes              7
53      Steve        Murray              7
54       Mark        Taylor              7
55      Diego     Gutiérrez              7
56       Luis         Rojas              7
57      Manoj        Pareek              7
58       Puja    Srivastava              6
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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    Get the total number of invoices for each customer\n'\n
\nThe DataFrame was produced using this query: SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalI
nvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId,
c.FirstName, c.LastName\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunnin
g df.dtypes gives:\n FirstName        object\nLastName         object\nTotalInvoices     int64\ndtype: obje
ct"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the results of the data
frame? Assume the data is in a pandas dataframe called 'df'. If there is only one value in the dataframe, u
se an Indicator. Respond with only Python code. Do not answer with any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T21:59:15.667930921Z', 'message': {'role': 'assistant',
'content': "import plotly.express as px\nfig = px.bar(df, x='FirstName', y='TotalInvoices', title='Total In
voices by Customer')\nfig.show()"}, 'done_reason': 'stop', 'done': True, 'total_duration': 20355944297, 'lo
ad_duration': 672336, 'prompt_eval_count': 199, 'prompt_eval_duration': 14027639000, 'eval_count': 36, 'eva
l_duration': 6197701000}
```


Total Invoices by Customer
TotalInvoices
0
2
4
6
8
10
12
14
FirstName
Luís
François
František
Astrid
Kara
Alexandre
Fernanda
Jennifer
Jack
Tim
Kathy
John
Richard
Julia
Edward
Aaron
João
Hannah
Niklas
Dominique
Wyatt
Terhi
Hugh
Johannes
Enrique
Emma
Steve
Luis
Puja

```
Out[22]: ('SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers"
         c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName',
              FirstName      LastName  TotalInvoices
          0        Luís     Gonçalves              7
          1      Leonie        Köhler              7
          2    François      Tremblay              7
          3       Bjørn        Hansen              7
          4    František   Wichterlová              7
          5      Helena          Holý              7
          6      Astrid        Gruber              7
          7        Daan       Peeters              7
          8        Kara       Nielsen              7
          9     Eduardo       Martins              7
          10  Alexandre         Rocha              7
          11    Roberto       Almeida              7
          12   Fernanda         Ramos              7
          13       Mark       Philips              7
          14   Jennifer      Peterson              7
          15      Frank        Harris              7
          16       Jack         Smith              7
          17   Michelle        Brooks              7
          18        Tim         Goyer              7
          19        Dan        Miller              7
          20      Kathy         Chase              7
          21    Heather       Leacock              7
          22       John        Gordon              7
          23      Frank       Ralston              7
          24     Victor       Stevens              7
          25    Richard    Cunningham              7
          26    Patrick          Gray              7
          27      Julia       Barnett              7
          28     Robert         Brown              7
          29     Edward       Francis              7
          30     Martha          Silk              7
          31      Aaron      Mitchell              7
          32      Ellie      Sullivan              7
          33       João     Fernandes              7
          34   Madalena       Sampaio              7
          35     Hannah     Schneider              7
          36       Fynn    Zimmermann              7
          37     Niklas      Schröder              7
          38    Camille       Bernard              7
```

```
39  Dominique      Lefebvre              7
40       Marc        Dubois              7
41      Wyatt        Girard              7
42   Isabelle       Mercier              7
43      Terhi    Hämäläinen              7
44   Ladislav        Kovács              7
45       Hugh      O'Reilly              7
46      Lucas       Mancini              7
47   Johannes  Van der Berg              7
48  Stanisław        Wójcik              7
49    Enrique         Muñoz              7
50     Joakim     Johansson              7
51       Emma         Jones              7
52       Phil        Hughes              7
53      Steve        Murray              7
54       Mark        Taylor              7
55      Diego     Gutiérrez              7
56       Luis         Rojas              7
57      Manoj        Pareek              7
58       Puja    Srivastava              6,
Figure({
    'data': [{'alignmentgroup': 'True',
              'hovertemplate': 'FirstName=%{x}<br>TotalInvoices=%{y}<extra></extra>',
              'legendgroup': '',
              'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
              'name': '',
              'offsetgroup': '',
              'orientation': 'v',
              'showlegend': False,
              'textposition': 'auto',
              'type': 'bar',
              'x': array(['Luís', 'Leonie', 'François', 'Bjørn', 'František', 'Helena', 'Astrid',
                          'Daan', 'Kara', 'Eduardo', 'Alexandre', 'Roberto', 'Fernanda', 'Mark',
                          'Jennifer', 'Frank', 'Jack', 'Michelle', 'Tim', 'Dan', 'Kathy',
                          'Heather', 'John', 'Frank', 'Victor', 'Richard', 'Patrick', 'Julia',
                          'Robert', 'Edward', 'Martha', 'Aaron', 'Ellie', 'João', 'Madalena',
                          'Hannah', 'Fynn', 'Niklas', 'Camille', 'Dominique', 'Marc', 'Wyatt',
                          'Isabelle', 'Terhi', 'Ladislav', 'Hugh', 'Lucas', 'Johannes',
                          'Stanisław', 'Enrique', 'Joakim', 'Emma', 'Phil', 'Steve', 'Mark',
                          'Diego', 'Luis', 'Manoj', 'Puja'], dtype=object),
              '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},
               'template': '...',
               'title': {'text': 'Total Invoices by Customer'},
               'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'FirstName'}},
               'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalInvoices'}}}
}))
```

In [23]:

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

vn.ask(question=question)
```

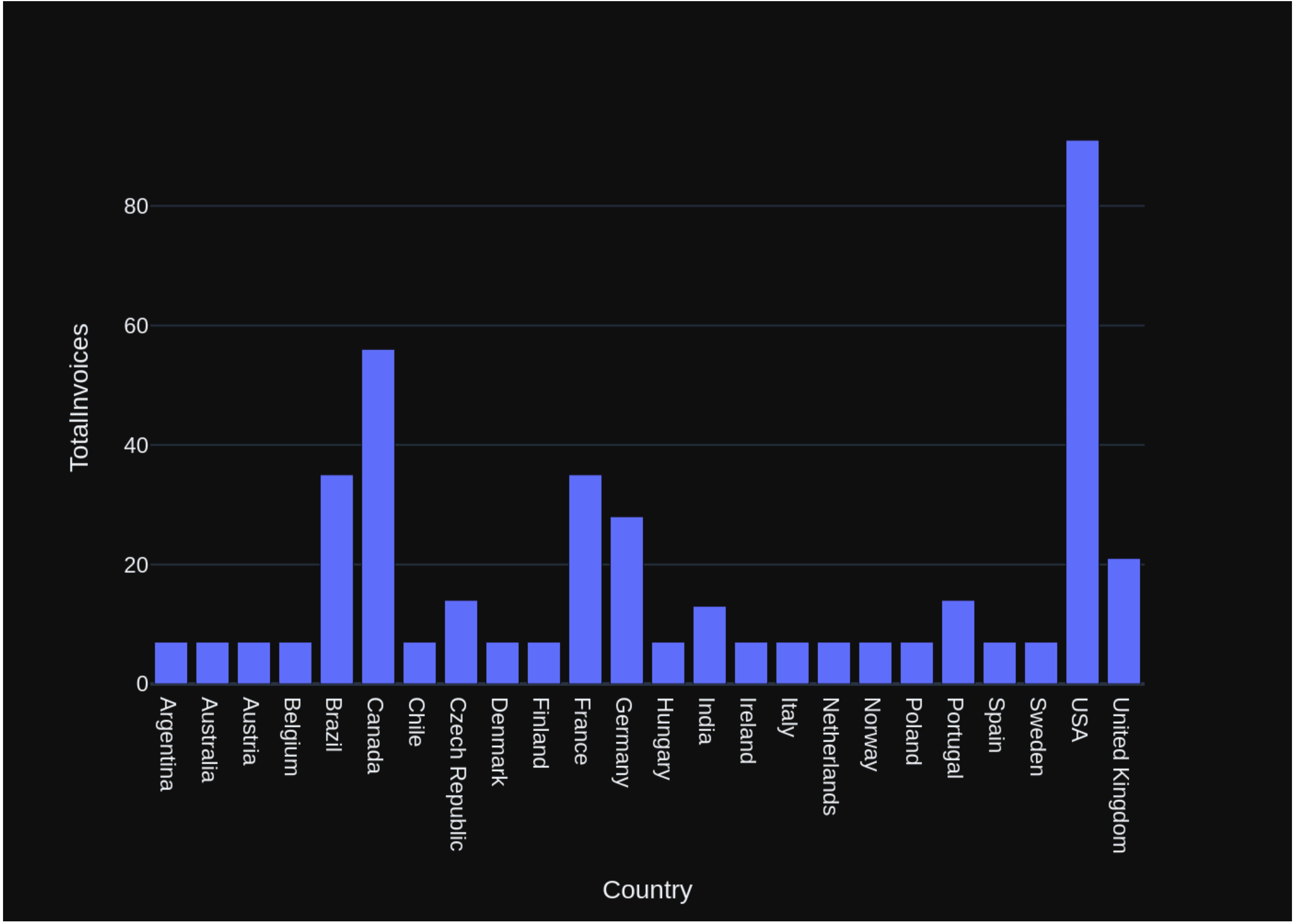
```
Number of requested results 10 is greater than number of elements in index 6, updating n_results = 6
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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\t
ON 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_InvoiceCustomerId ON "invoices" (Custome
rId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceLin
eTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "employees"\r\n(\r\n    EmployeeId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NUL
L,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIM
E,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHA
R(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVA
RCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMEN
T 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    Coun
try 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 "em
ployees" (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    Arti
stId 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 AUTOINCREM
ENT 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 "med
ia_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeRe
portsTo ON "employees" (ReportsTo)\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, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is almost sufficient but requires k
nowledge 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 provi
ded context is insufficient, please 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 repeat the answer exactly as it wa
s given before. \n'}, {'role': 'user', 'content': '  \n    Get the total number of invoices for each custom
```

```
er\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoic
es\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstNam
e, c.LastName'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'r
ole': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER
BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistan
t', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': '  \n    List all albums an
d their corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.
Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'con
tent': '  \n    Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistant',
'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, {'role': 'user', 'content': 'Can you lis
t all tables in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_m
aster WHERE type='table'"}, {'role': 'user', 'content': '  \n    Find the total number of invoices per coun
try:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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 NUL
L,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NO
T NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceI
d) \r\n\t\tON DELETE NO ACTION 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 INDEX IFK_InvoiceCustomerId ON \"invoi
ces\" (CustomerId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE INDE
X IFK_InvoiceLineTrackId ON \"invoice_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 NVAR
CHAR(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 D
ELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n    CustomerId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  NOT N
ULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARC
HAR(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 (Support
RepId) 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\" (ArtistI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTE
GER 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 I
NTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (Album
Id) REFERENCES \"albums\" (AlbumId) \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\" (MediaTypeId) \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 c
hinook database invoice means order\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 in
termediate SQL query to find the 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. \n
4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please r
epeat 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": "SELECT c.FirstName, c.LastName,
COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.Customer
Id\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "user", "content": "what are the top 5 count
ries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM
\"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "How many cust
omers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user",
"content": "  \n    List all albums and their corresponding artist names  \n"}, {"role": "assistant", "cont
ent": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.Artis
tId = ar.ArtistId"}, {"role": "user", "content": "  \n    Find all tracks with a name containing \"What\"
(case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%wha
t%'"}, {"role": "user", "content": "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 country:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:00:54.93352211Z', 'message': {'role': 'assistant',
'content': 'SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON
i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, 'done_reason': 'stop', 'done': True, 'total_duration': 9
9143357340, 'load_duration': 787360, 'prompt_eval_count': 1267, 'prompt_eval_duration': 91592011000, 'eval_
count': 40, 'eval_duration': 7151241000}
SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.Country
SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices
```

```
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 total number of invoices per country:\n'\n\nThe
DataFrame was produced using this query: SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invo
ices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country\n\nThe following is infor
mation about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Country          object\nTot
alInvoices     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 on
ly one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with any expl
anations -- just the code."}]
```

```
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:01:13.012233305Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\n\nfig = px.bar(df, x='Country', y='TotalInvoices')\nfig.show
()"}, 'done_reason': 'stop', 'done': True, 'total_duration': 18051153220, 'load_duration': 44135812, 'promp
t_eval_count': 184, 'prompt_eval_duration': 12792921000, 'eval_count': 30, 'eval_duration': 5168773000}
```



```
Out[23]: ('SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.Custom
         erId = c.CustomerId\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', 'Netherlands', '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},
                         'margin': {'t': 60},
                         'template': '...',
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Country'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalInvoices'}}}
          }))
```

In [24]:

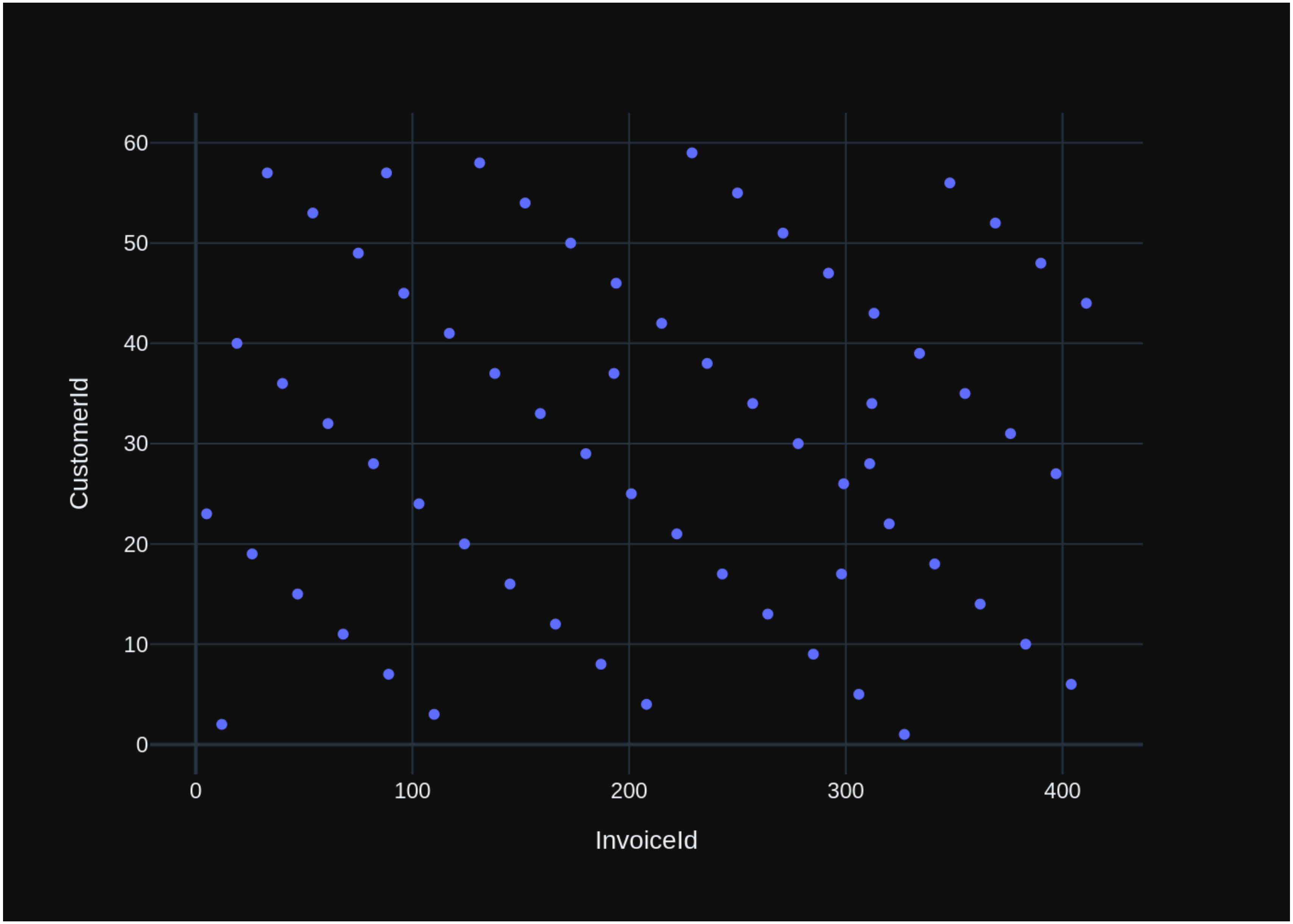
```
question = """
    List all invoices with a total exceeding $10:
"""

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
```

```
[{'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 fo
rmat 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    UnitPr
ice 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) REFERE
NCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLin
eInvoiceId 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    Bill
ingCountry 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 ACTIO
N\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREATE INDEX IFK_InvoiceCusto
merId ON "invoices" (CustomerId)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCRE
MENT 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 "med
ia_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeRe
portsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY A
UTOINCREMENT 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) REFERENC
ES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "employee
s"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NUL
L,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    Bir
thDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    St
ate 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" (E
mployeeId) \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 conte
xt 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 b
efore. \n'}, {'role': 'user', 'content': '  \n    Get the total number of invoices for each customer\n'},
{'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM
"invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.Last
```

```
Name'}, {'role': 'user', 'content': '  \n    Find the total number of invoices per country:\n'}, {'role':
'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "cu
stomers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': 'what are the
top 5 countries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as
Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total DESC\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'}, {'role': 'assistan
t', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.
ArtistId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    Find all tracks with a name containing "Wha
t" (case-insensitive)\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%wh
at%\''}, {'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'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoice_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY K
EY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    Uni
tPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCE
S \"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 ACTION\r\n)\n\nCREATE INDEX IFK_Invo
iceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGE
R 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(4
0),\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 IN
DEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEG
ER 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 I
NTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (Album
Id) REFERENCES \"albums\" (AlbumId) \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\" (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    Last
Name 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 UPDAT
E 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 NVARCHA
R(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCH
AR(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 sufficie
nt, please generate a valid SQL query without any explanations for the question. \n2. If the provided conte
xt 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 say
ing intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generate
d. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, ple
ase repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": "  \n    Get the to
tal number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT c.FirstName, c.LastN
ame, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.Cus
tomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "user", "content": "  \n    Find the to
tal number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.Invoic
eId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY
c.Country"}, {"role": "user", "content": "what are the top 5 countries that customers come from?"}, {"rol
e": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER
BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistan
t", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user", "content": "  \n    List all albums
and their corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.ArtistId, a
r.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "use
r", "content": "  \n    Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "a
ssistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%what%'"}, {"role": "user", "content":
"Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name F
ROM sqlite_master WHERE type='table'"}, {"role": "user", "content": "  \n    List all invoices with a total
exceeding $10:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:02:48.413536218Z', 'message': {'role': 'assistant',
'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, 'done_reason': 'stop', 'done': True, 'total_d
uration': 95291660155, 'load_duration': 657630, 'prompt_eval_count': 1282, 'prompt_eval_duration': 92448598
000, 'eval_count': 14, 'eval_duration': 2402653000}
SELECT *
FROM "invoices"
WHERE Total > 10.00
SELECT *
FROM "invoices"
```

```
WHERE Total > 10.00
    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]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 invoices with a total exceeding $10:\n'\n\nThe
DataFrame was produced using this query: SELECT *\nFROM \"invoices\"\nWHERE Total > 10.00\n\nThe following
is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n InvoiceId
int64\nCustomerId             int64\nInvoiceDate           object\nBillingAddress        object\nBillingCit
y           object\nBillingState          object\nBillingCountry        object\nBillingPostalCode     objec
t\nTotal                float64\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 wi
th any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:03:16.492714003Z', 'message': {'role': 'assistant',
'content': '```\nimport plotly.express as px\nfig = px.histogram(df, x=\'Total\', title=\'Invoices Total Va
lue\')\nfig.update_layout(yaxis={\'categoryorder\':\'total descending\'})\nif df.shape[0] == 1:\n    fig =
px.bar(x=[\'Total\'], y=[df[\'Total\'].values[0]], textposition="auto")\nelse:\n    fig.show()\n```'}, 'don
e_reason': 'stop', 'done': True, 'total_duration': 28053334717, 'load_duration': 41893087, 'prompt_eval_cou
nt': 195, 'prompt_eval_duration': 13700094000, 'eval_count': 81, 'eval_duration': 14222647000}
```


CustomerId
0
10
20
30
40
50
60
InvoiceId
0
100
200
300
400

```
Out[24]:  ('SELECT *\nFROM "invoices"\nWHERE Total > 10.00',
               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],
           Figure({
               'data': [{'hovertemplate': 'InvoiceId=%{x}<br>CustomerId=%{y}<extra></extra>',
                         'legendgroup': '',
                         'marker': {'color': '#636efa', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': '',
                         'orientation': 'v',
                         'showlegend': False,
                         'type': 'scatter',
                         'x': array([  5,  12,  19,  26,  33,  40,  47,  54,  61,  68,  75,  82,  88,  89,
                                      96, 103, 110, 117, 124, 131, 138, 145, 152, 159, 166, 173, 180, 187,
                                     193, 194, 201, 208, 215, 222, 229, 236, 243, 250, 257, 264, 271, 278,
                                     285, 292, 298, 299, 306, 311, 312, 313, 320, 327, 334, 341, 348, 355,
                                     362, 369, 376, 383, 390, 397, 404, 411]),
                         'xaxis': 'x',
                         'y': array([23,  2, 40, 19, 57, 36, 15, 53, 32, 11, 49, 28, 57,  7, 45, 24,  3, 41,
                                     20, 58, 37, 16, 54, 33, 12, 50, 29,  8, 37, 46, 25,  4, 42, 21, 59, 38,
                                     17, 55, 34, 13, 51, 30,  9, 47, 17, 26,  5, 28, 34, 43, 22,  1, 39, 18,
                                     56, 35, 14, 52, 31, 10, 48, 27,  6, 44]),
                         'yaxis': 'y'}],
               'layout': {'legend': {'tracegroupgap': 0},
                          'margin': {'t': 60},
                          'template': '...',
                          'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'InvoiceId'}},
                          'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'CustomerId'}}}
           }))
```

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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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\t
ON 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_InvoiceLin
eTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "employees"\r\n(\r\n    EmployeeId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NUL
L,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIM
E,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHA
R(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVA
RCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMEN
T 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    Coun
try 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 "em
ployees" (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    Album
Id 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 A
CTION\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_trac
k"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_Playlis
tTrack 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 (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 dat
abase invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please gene
rate 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 intermediat
e SQL query to find the distinct 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 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': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.0
0'}, {'role': 'user', 'content': '  \n    Find the total number of invoices per country:\n'}, {'role': 'ass
istant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "custom
ers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '  \n    Get the t
otal number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.Last
Name, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.Custom
erId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user', 'content': 'what are the top 5 cou
ntries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFR
OM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': 'How many cus
tomers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user',
'content': '  \n    Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'assista
nt', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, {'role': 'user', 'content': '  \n
List all albums and their corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Titl
e, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'r
ole': '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 invoices since 2010 and the total amount invoiced:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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 NUL
L,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NO
T NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceI
d) \r\n\t\tON DELETE NO ACTION 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 INDEX IFK_InvoiceLineInvoiceId ON \"in
voice_items\" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDE
X IFK_InvoiceLineTrackId ON \"invoice_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 NVAR
CHAR(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 D
ELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n    CustomerId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  NOT N
ULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARC
HAR(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 (Support
RepId) 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    Comp
oser NVARCHAR(220),\r\n    Milliseconds 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 (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 D
ELETE 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    FOR
EIGN 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 N
ULL,\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===Addi
tional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provid
ed 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 co
lumn, 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 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 invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *\nFRO
M \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "  \n    Find the total number of invoic
es per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoice
s\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country"}, {"role":
"user", "content": "  \n    Get the total number of invoices for each customer\n"}, {"role": "assistant",
"content": "SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN
\"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role":
"user", "content": "what are the top 5 countries that customers come from?"}, {"role": "assistant", "conten
t": "SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT
5"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistant", "content": "SELECT
COUNT(*) FROM \"customers\""}, {"role": "user", "content": "  \n    Find all tracks with a name containing
\"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LI
KE '%what%'"}, {"role": "user", "content": "  \n    List all albums and their corresponding artist names
\n"}, {"role": "assistant", "content": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\"
```

```
a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "Can you list all tables
in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE t
ype='table'"}, {"role": "user", "content": "  \n    Find all invoices since 2010 and the total amount invoi
ced:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:05:04.183174769Z', 'message': {'role': 'assistant',
'content': 'SELECT *\nFROM "invoices"\nWHERE InvoiceDate >= \'2010-01-01\'\nORDER BY InvoiceDate\nSELECT SU
M(Total) AS TotalAmount\nFROM "invoices"\nWHERE InvoiceDate >= \'2010-01-01\''}, 'done_reason': 'stop', 'do
ne': True, 'total_duration': 107587032474, 'load_duration': 686792, 'prompt_eval_count': 1444, 'prompt_eval
_duration': 98166411000, 'eval_count': 49, 'eval_duration': 8930446000}
SELECT *
FROM "invoices"
WHERE InvoiceDate >= '2010-01-01'
ORDER BY InvoiceDate
SELECT SUM(Total) AS TotalAmount
FROM "invoices"
WHERE InvoiceDate >= '2010-01-01'
SELECT *
FROM "invoices"
WHERE InvoiceDate >= '2010-01-01'
ORDER BY InvoiceDate
SELECT SUM(Total) AS TotalAmount
FROM "invoices"
WHERE InvoiceDate >= '2010-01-01'
Couldn't run sql:  Execution failed on sql 'SELECT *
FROM "invoices"
WHERE InvoiceDate >= '2010-01-01'
ORDER BY InvoiceDate
SELECT SUM(Total) AS TotalAmount
FROM "invoices"
WHERE InvoiceDate >= '2010-01-01'': near "SELECT": syntax error
```

In [26]:

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

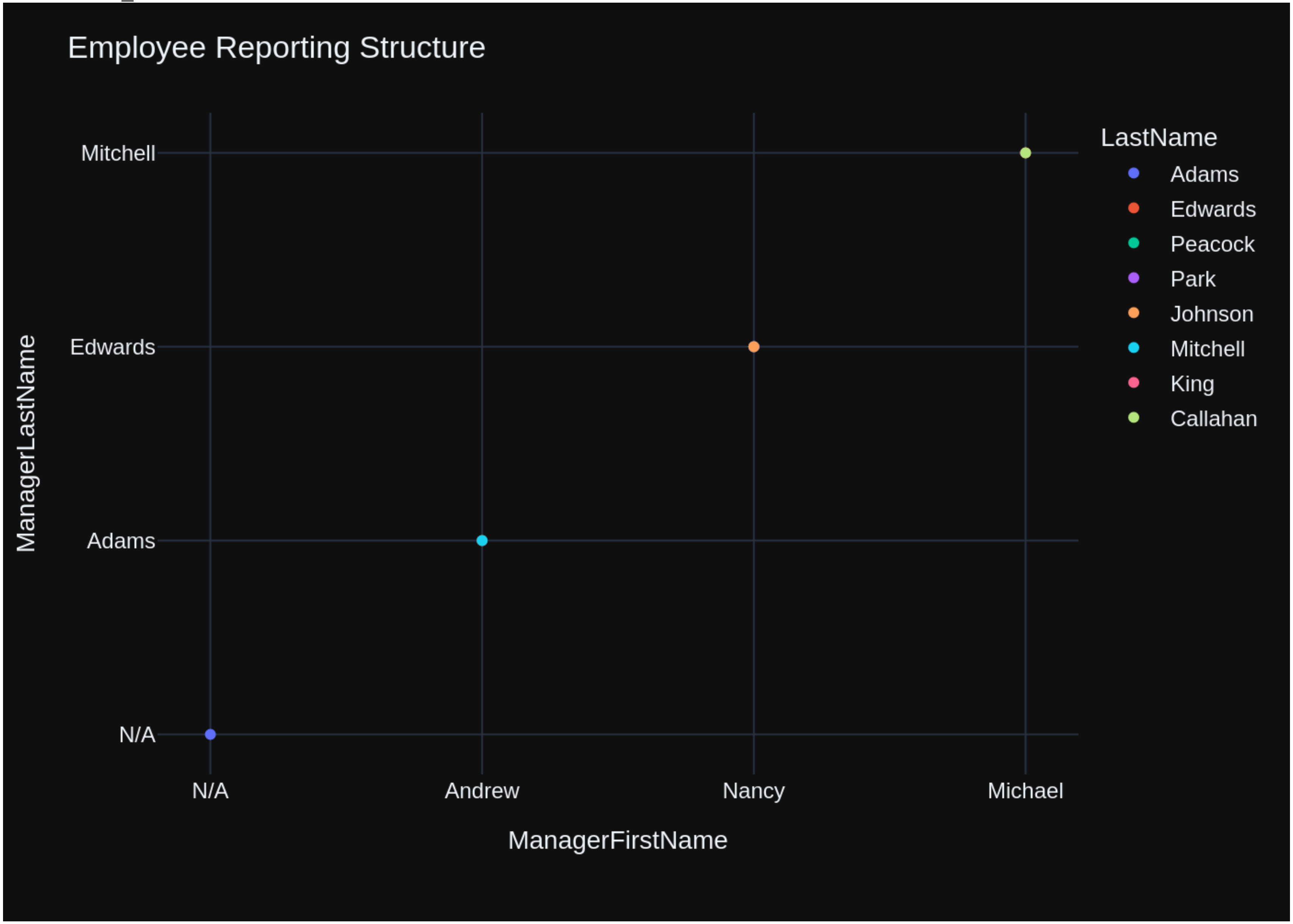
```
[{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the
question. Your response should ONLY be based on the given context and follow the response guidelines and fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TA
BLE "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 INTEGE
R,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(4
0),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NV
ARCHAR(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(1
0),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    Support
RepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCR
EATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId I
NTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    Billing
City NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCountry NVARCHAR(40),\r\n    BillingPost
alCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cust
omers" (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 NUL
L,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE N
O 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 AUTOINCREMEN
T NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NU
LL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    By
tes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (Al
bumId) \r\n\t\tON DELETE NO ACTION 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 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    Arti
stId 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\nI
n the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is suffic
ient, please generate a valid SQL query without any explanations for the question. \n2. If the provided con
text is almost sufficient but requires knowledge of a specific string in a particular column, please genera
te 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 gener
ated. \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 c.FirstName, c.Las
tName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.Custo
merId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user', 'content': 'what are the top 5 co
untries that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nF
ROM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Fin
d the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT
(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP
BY c.Country'}, {'role': 'user', 'content': '  \n    List all albums and their corresponding artist names
\n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a
\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    List all invoices w
ith a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total >
10.00'}, {'role': 'user', 'content': 'How many customers are there'}, {'role': 'assistant', 'content': 'SEL
ECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite databas
e catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'rol
e': 'user', 'content': '  \n    Find all tracks with a name containing "What" (case-insensitive)\n'}, {'rol
e': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, {'role': 'user', 'conte
nt': "  \n    List all employees and their reporting manager's name (if any):\n"}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat 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 NVAR
CHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo IN
TEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCH
AR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phon
e NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENC
ES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"custom
ers\"\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(1
0),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    Support
RepId 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    Custom
erId 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    BillingCountry NVARCHAR(40),\r\n    Billi
ngPostalCode 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_i
```

```
tems\"\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 INT
EGER  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    FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENC
ES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) R
EFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABL
E \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NO
T 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===Add
itional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provi
ded context is sufficient, please 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 insufficient, please explain w
hy it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked a
nd answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "conten
t": "  \n    Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELE
CT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c
ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "user", "conten
t": "what are the top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Cou
ntry, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "us
er", "content": "  \n    Find the total number of invoices per country:\n"}, {"role": "assistant", "conten
t": "SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.
CustomerId = c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "content": "  \n    List all albums and t
heir corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.ArtistId, ar.Nam
e AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "co
ntent": "  \n    List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELEC
T *\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "How many customers are there"},
{"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user", "content": "Can yo
u list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sql
ite_master WHERE type='table'"}, {"role": "user", "content": "  \n    Find all tracks with a name containin
g \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name
LIKE '%what%'"}, {"role": "user", "content": "  \n    List all employees and their reporting manager's name
(if any):\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:06:53.003624205Z', 'message': {'role': 'assistant',
```

````
'content': 'SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS ManagerFirstName,\n
COALESCE(e2.LastName, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "employees" e2 ON e.Report
sTo = e2.EmployeeId'}, 'done_reason': 'stop', 'done': True, 'total_duration': 108777082648, 'load_duratio
n': 1252006, 'prompt_eval_count': 1326, 'prompt_eval_duration': 96611930000, 'eval_count': 64, 'eval_durati
on': 11670005000}
SELECT e.FirstName, e.LastName, 
       COALESCE(e2.FirstName, 'N/A') AS ManagerFirstName,
       COALESCE(e2.LastName, 'N/A') AS ManagerLastName
FROM "employees" e
LEFT JOIN "employees" e2 ON e.ReportsTo = e2.EmployeeId
SELECT e.FirstName, e.LastName, 
       COALESCE(e2.FirstName, 'N/A') AS ManagerFirstName,
       COALESCE(e2.LastName, 'N/A') AS ManagerLastName
FROM "employees" e
LEFT JOIN "employees" e2 ON e.ReportsTo = e2.EmployeeId
  FirstName  LastName ManagerFirstName ManagerLastName
0    Andrew     Adams              N/A             N/A
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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 (i
f any):\n'\n\nThe DataFrame was produced using this query: SELECT e.FirstName, e.LastName, \n       COALESC
E(e2.FirstName, 'N/A') AS ManagerFirstName,\n       COALESCE(e2.LastName, 'N/A') AS ManagerLastName\nFROM 
\"employees\" e\nLEFT JOIN \"employees\" e2 ON e.ReportsTo = e2.EmployeeId\n\nThe following is information 
about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n FirstName           object\nLastNam
e            object\nManagerFirstName    object\nManagerLastName     object\ndtype: object"}, {"role": "use
r", "content": "Can you generate the Python plotly code to chart the results of the dataframe? Assume the d
ata is in a pandas dataframe called 'df'. If there is only one value in the dataframe, use an Indicator. Re
spond with only Python code. Do not answer with any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:07:24.287831683Z', 'message': {'role': 'assistant', 
'content': "```\nimport plotly.express as px\nimport numpy as np\n\nfig = px.scatter(df, x='ManagerFirstNam
````

```
e', y='ManagerLastName', hover_name='FirstName', color='LastName')\n\nfig.update_layout(title='Employee Rep
orting Structure')\n\nif df.shape[0] == 1:\n    fig = px.bar(x=[''], y=[df['FirstName'].values], labels=
{'x': '', 'y': ''})\n\nfig.show()\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 3126211743
2, 'load_duration': 43573330, 'prompt_eval_count': 219, 'prompt_eval_duration': 15467972000, 'eval_count':
90, 'eval_duration': 15706150000}
```



```
Out[26]:  ('SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS ManagerFirstName,\n       CO
          ALESCE(e2.LastName, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "employees" e2 ON e.Reports
          To = e2.EmployeeId',
             FirstName  LastName ManagerFirstName ManagerLastName
           0    Andrew     Adams              N/A             N/A
           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': [{'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Andrew'], dtype=object),
                         'legendgroup': 'Adams',
                         'marker': {'color': '#636efa', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Adams',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['N/A'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['N/A'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Nancy'], dtype=object),
                         'legendgroup': 'Edwards',
                         'marker': {'color': '#EF553B', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Edwards',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Andrew'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Adams'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Jane'], dtype=object),
                         'legendgroup': 'Peacock',
                         'marker': {'color': '#00cc96', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Peacock',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Nancy'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Edwards'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Margaret'], dtype=object),
                         'legendgroup': 'Park',
                         'marker': {'color': '#ab63fa', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Park',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Nancy'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Edwards'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Steve'], dtype=object),
                         'legendgroup': 'Johnson',
                         'marker': {'color': '#FFA15A', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Johnson',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Nancy'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Edwards'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Michael'], dtype=object),
                         'legendgroup': 'Mitchell',
                         'marker': {'color': '#19d3f3', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Mitchell',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Andrew'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Adams'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Robert'], dtype=object),
                         'legendgroup': 'King',
                         'marker': {'color': '#FF6692', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'King',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Michael'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Mitchell'], dtype=object),
                         'yaxis': 'y'},
                        {'hovertemplate': ('<b>%{hovertext}</b><br><br>Las' ... 'erLastName=%{y}<extra></extra>'),
                         'hovertext': array(['Laura'], dtype=object),
                         'legendgroup': 'Callahan',
                         'marker': {'color': '#B6E880', 'symbol': 'circle'},
                         'mode': 'markers',
                         'name': 'Callahan',
                         'orientation': 'v',
                         'showlegend': True,
                         'type': 'scatter',
                         'x': array(['Michael'], dtype=object),
                         'xaxis': 'x',
                         'y': array(['Mitchell'], dtype=object),
                         'yaxis': 'y'}],
               'layout': {'legend': {'title': {'text': 'LastName'}, 'tracegroupgap': 0},
                          'margin': {'t': 60},
                          'template': '...',
                          'title': {'text': 'Employee Reporting Structure'},
                          'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'ManagerFirstName'}},
                          'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'ManagerLastName'}}}
           }))
```

In [27]:

```
question = """
    Get the average invoice total for each customer:
"""

vn.ask(question=question)
```

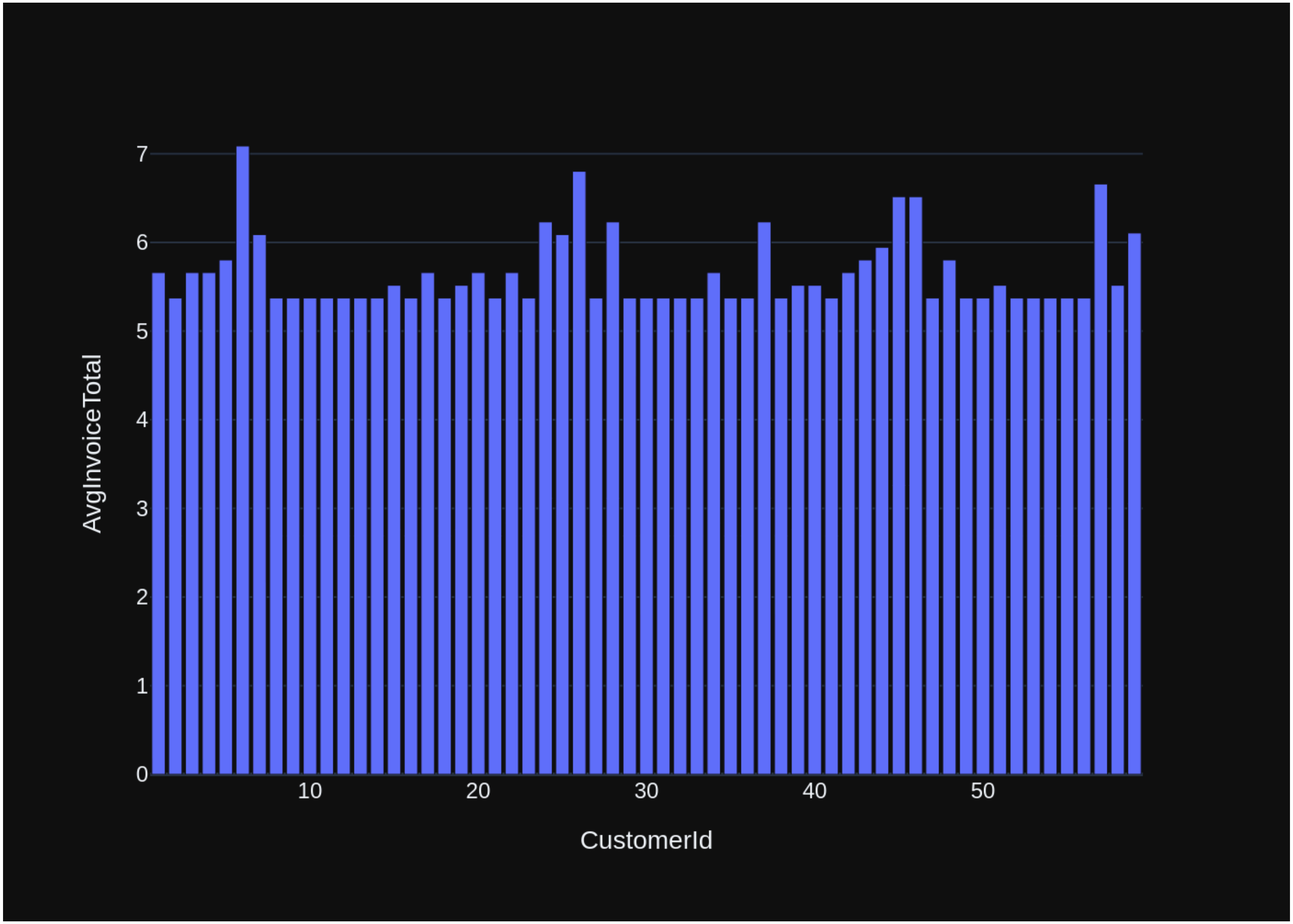
```
Number of requested results 10 is greater than number of elements in index 9, updating n_results = 9
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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 KE
Y AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    Unit
Price 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) REFERE
NCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLin
eTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE INDEX IFK_Custom
erSupportRepId ON "customers" (SupportRepId)\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    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVAR
CHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fa
x NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREIGN KEY (Suppo
rtRepId) 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 I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHA
R(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    Hir
eDate 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(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 wi
thout 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 intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provid
ed context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant ta
ble(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 c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices
\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName,
c.LastName'}, {'role': 'user', 'content': '  \n    Find the total number of invoices per country:\n'}, {'ro
le': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOI
N "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '  \n
List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoic
es"\nWHERE Total > 10.00'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come f
rom?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Cou
```

```
ntry\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'}, {'role':
'assistant', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': "  \n    List all
employees and their reporting manager's name (if any):\n"}, {'role': 'assistant', 'content': 'SELECT e.Firs
tName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS ManagerFirstName,\n       COALESCE(e2.LastNa
me, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "employees" e2 ON e.ReportsTo = e2.EmployeeI
d'}, {'role': 'user', 'content': '  \n    List all albums and their corresponding artist names  \n'}, {'rol
e': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "arti
sts" ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    Find all tracks with a name con
taining "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Nam
e LIKE \'%what%\''}, {'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': 'use
r', 'content': '  \n    Get the average invoice total for each customer:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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_InvoiceLineInv
oiceId 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 (Tr
ackId) 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    Cu
stomerId INTEGER PRIMARY KEY AUTOINCREMENT 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 NVARCHAR(70),\r\n    City NVARCHAR(4
0),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NV
ARCHAR(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 A
CTION\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 NUL
L,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    Bir
thDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    St
ate 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 chi
nook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, ple
ase 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 int
ermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying in
termediate_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 repe
at the answer exactly as it was given before. \n"}, {"role": "user", "content": "  \n    Get the total numb
er of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT c.FirstName, c.LastName, COU
NT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId
\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"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\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Countr
y"}, {"role": "user", "content": "  \n    List all invoices with a total exceeding $10:\n"}, {"role": "assi
stant", "content": "SELECT *\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "what a
re the top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT
(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "conte
nt": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers
\""}, {"role": "user", "content": "  \n    List all employees and their reporting manager's name (if an
y):\n"}, {"role": "assistant", "content": "SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName,
'N/A') AS ManagerFirstName,\n       COALESCE(e2.LastName, 'N/A') AS ManagerLastName\nFROM \"employees\" e\n
LEFT JOIN \"employees\" e2 ON e.ReportsTo = e2.EmployeeId"}, {"role": "user", "content": "  \n    List all
albums and their corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.Arti
stId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role":
"user", "content": "  \n    Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"rol
e": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%what%'"}, {"role": "user", "conte
nt": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT n
ame FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": "  \n    Get the average invoice t
otal for each customer:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:09:03.309363656Z', 'message': {'role': 'assistant',
'content': 'SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON
i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId'}, 'done_reason': 'stop', 'done': True, 'total_duratio
n': 98888661022, 'load_duration': 1301737, 'prompt_eval_count': 1255, 'prompt_eval_duration': 91116399000,
'eval_count': 40, 'eval_duration': 7230889000}
SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal
FROM "invoices" i
```

```
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
    CustomerId  AvgInvoiceTotal
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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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    Get the average invoice total for each customer:\n'\n\nT
he DataFrame was produced using this query: SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"in
voices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\n\nThe following is
information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId           in
t64\nAvgInvoiceTotal    float64\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 wi
th any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:09:21.788686384Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nfig = px.bar(df, x='CustomerId', y='AvgInvoiceTotal')\nfig.sh
ow()\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 18451579989, 'load_duration': 693682,
'prompt_eval_count': 186, 'prompt_eval_duration': 12927630000, 'eval_count': 31, 'eval_duration': 539038700
0}
```


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

```
Out[27]: ('SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.Custome
         rId = c.CustomerId\nGROUP BY c.CustomerId',
              CustomerId  AvgInvoiceTotal
          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>AvgInvoiceTotal=%{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': {'text': 'CustomerId'}},
               'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'AvgInvoiceTotal'}}}
}))
```

In [28]:

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

vn.ask(question=question)
```

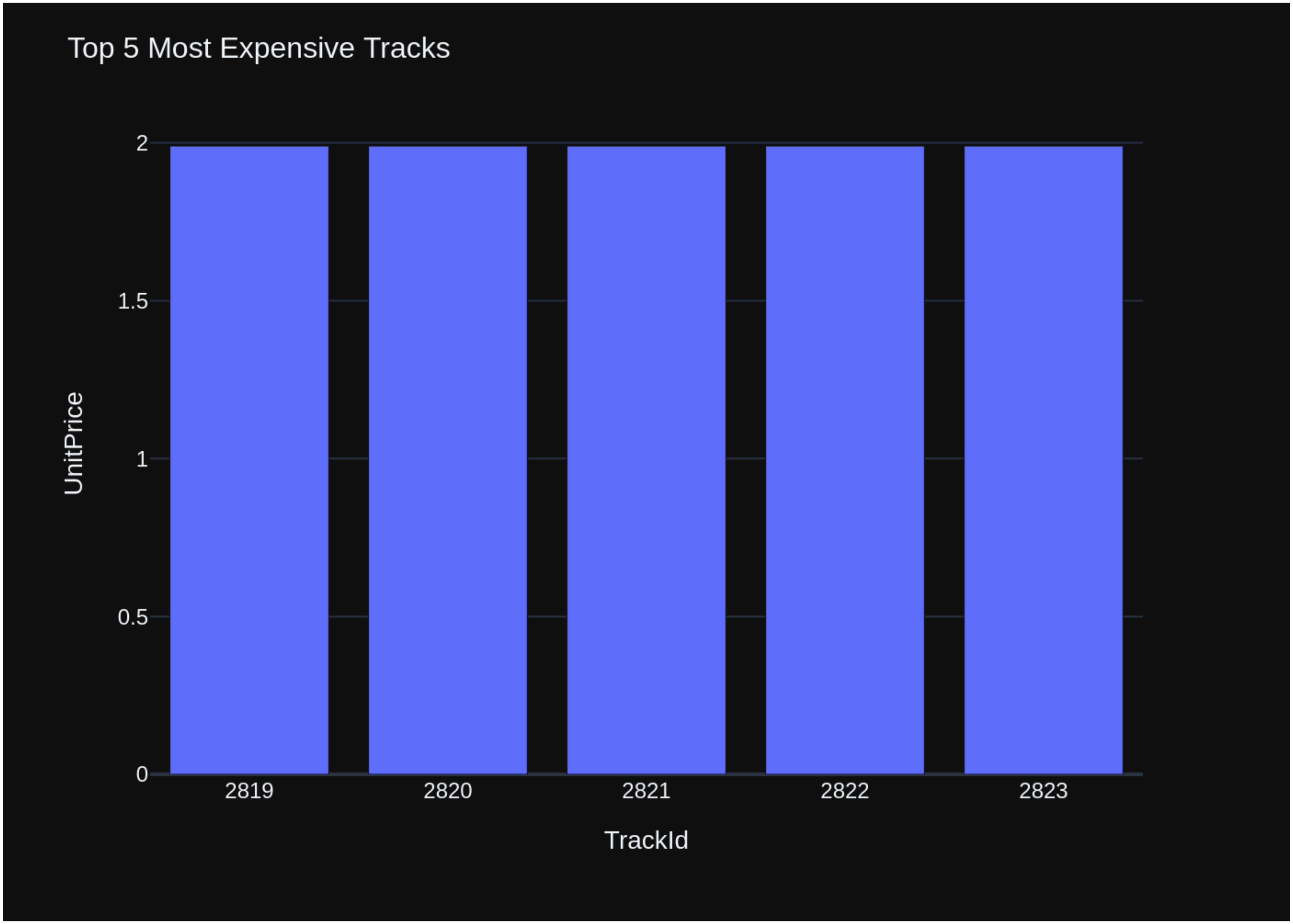
```
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE 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 NU
LL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    By
tes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (Al
bumId) \r\n\t\tON DELETE NO ACTION 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 ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackAlbumId
ON "tracks" (AlbumId)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX IFK_PlaylistTr
ackTrackId ON "playlist_track" (TrackId)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackI
d)\n\nCREATE INDEX IFK_TrackMediaTypeId 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    TrackI
d INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FO
REIGN 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    TrackId INTEGER  NOT
NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistI
d) 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 INDE
X IFK_AlbumArtistId ON "albums" (ArtistId)\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 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 ques
tion. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a pa
rticular column, please generate an intermediate SQL query to find the distinct strings in that column. Pre
pend 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 be
en asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'use
r', 'content': '  \n    Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'ass
istant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, {'role': 'user', 'content': '
\n    List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM
"invoices"\nWHERE Total > 10.00'}, {'role': 'user', 'content': '  \n    List all albums and their correspon
ding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistNam
e\nFROM "albums" a\nJOIN "artists" 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 Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '
\n    Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.Cust
omerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.Customer
Id\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': '  \n    Find the total number of invoices per cou
ntry:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "i
```

```
nvoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'cont
ent': '  \n    Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SE
LECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON
i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user', 'content':
'Can you list all tables in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT name F
ROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': 'How many customers are there'}, {'rol
e': 'assistant', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': "  \n    List
all employees and their reporting manager's name (if any):\n"}, {'role': 'assistant', 'content': 'SELECT e.
FirstName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS ManagerFirstName,\n       COALESCE(e2.La
stName, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "employees" e2 ON e.ReportsTo = e2.Emplo
yeeId'}, {'role': 'user', 'content': '  \n    Find the top 5 most expensive tracks (based on unit pric
e):\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREM
ENT 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_Trac
kAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE INDEX I
FK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice
_items\" (TrackId)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE TABLE \"invoi
ce_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 INTEG
ER  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\" (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 NUL
L,\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 UP
DATE 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\" (ArtistId)\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\tO
N 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 S
```

```
QL query without any explanations for the question. \n2. If the provided context is almost sufficient but r
equires 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 re
levant 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 all tracks with a name containing
\"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LI
KE '%what%'"}, {"role": "user", "content": "  \n    List all invoices with a total exceeding $10:\n"}, {"ro
le": "assistant", "content": "SELECT *\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "conten
t": "  \n    List all albums and their corresponding artist names  \n"}, {"role": "assistant", "content":
"SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId =
ar.ArtistId"}, {"role": "user", "content": "what are the top 5 countries that customers come from?"}, {"rol
e": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER
BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "  \n    Get the average invoice total for each custo
mer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"in
voices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user",
"content": "  \n    Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "S
ELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.Custom
erId = c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "content": "  \n    Get the total number of inv
oices for each customer\n"}, {"role": "assistant", "content": "SELECT c.FirstName, c.LastName, COUNT(i.Invo
iceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY
c.CustomerId, c.FirstName, c.LastName"}, {"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    List all employees and their reporting manag
er's name (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstName, e.LastName, \n       COALESC
E(e2.FirstName, 'N/A') AS ManagerFirstName,\n       COALESCE(e2.LastName, 'N/A') AS ManagerLastName\nFROM
\"employees\" e\nLEFT JOIN \"employees\" e2 ON e.ReportsTo = e2.EmployeeId"}, {"role": "user", "content": "
\n    Find the top 5 most expensive tracks (based on unit price):\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:10:39.533111719Z', 'message': {'role': 'assistant',
'content': 'SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, 'done_reason': 'stop', 'done': Tru
e, 'total_duration': 77651271804, 'load_duration': 792863, 'prompt_eval_count': 1241, 'prompt_eval_duratio
n': 74192090000, 'eval_count': 16, 'eval_duration': 2783201000}
SELECT *
FROM "tracks"
ORDER BY UnitPrice DESC
LIMIT 5
SELECT *
FROM "tracks"
ORDER BY UnitPrice DESC
LIMIT 5
```

```
   TrackId                                    Name  AlbumId  MediaTypeId  \
0     2819  Battlestar Galactica: The Story So Far      226            3   
1     2820                  Occupation / Precipice      227            3   
2     2821                           Exodus, Pt. 1      227            3   
3     2822                           Exodus, Pt. 2      227            3   
4     2823                           Collaborators      227            3   

   GenreId Composer  Milliseconds       Bytes  UnitPrice  
0       18     None       2622250   490750393       1.99  
1       19     None       5286953  1054423946       1.99  
2       19     None       2621708   475079441       1.99  
3       19     None       2618000   466820021       1.99  
4       19     None       2626626   483484911       1.99  
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 top 5 most expensive tracks (based on unit pric
e):\n'\n\nThe DataFrame was produced using this query: SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\n
LIMIT 5\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes give
s:\n TrackId           int64\nName             object\nAlbumId           int64\nMediaTypeId       int64\nGe
nreId           int64\nComposer         object\nMilliseconds      int64\nBytes             int64\nUnitPrice
float64\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 -- j
ust the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:11:02.658229127Z', 'message': {'role': 'assistant',
'content': '```\nimport plotly.express as px\nfig = px.bar(df, x="TrackId", y="UnitPrice")\nfig.update_layo
ut(title=\'Top 5 Most Expensive Tracks\', titlefont_size=16)\nfig.show()\n```'}, 'done_reason': 'stop', 'do
ne': True, 'total_duration': 23096466168, 'load_duration': 680743, 'prompt_eval_count': 202, 'prompt_eval_d
uration': 14170649000, 'eval_count': 50, 'eval_duration': 8785832000}
```


Top 5 Most Expensive Tracks
UnitPrice
2
1.5
1
0.5
0
2819
2820
2821
2822
2823
TrackId

Out[28]:

```
('SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5',
    TrackId                                    Name  AlbumId  MediaTypeId  \
 0     2819  Battlestar Galactica: The Story So Far      226            3   
 1     2820                    Occupation / Precipice      227            3   
 2     2821                           Exodus, Pt. 1      227            3   
 3     2822                           Exodus, Pt. 2      227            3   
 4     2823                           Collaborators      227            3   

    GenreId Composer  Milliseconds       Bytes  UnitPrice  
 0       18     None       2622250   490750393       1.99  
 1       19     None       5286953  1054423946       1.99  
 2       19     None       2621708   475079441       1.99  
 3       19     None       2618000   466820021       1.99  
 4       19     None       2626626   483484911       1.99  ,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'TrackId=%{x}<br>UnitPrice=%{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([2819, 2820, 2821, 2822, 2823]),
               'xaxis': 'x',
               'y': array([1.99, 1.99, 1.99, 1.99, 1.99]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'title': {'font': {'size': 16}, 'text': 'Top 5 Most Expensive Tracks'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'TrackId'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'UnitPrice'}}}
 }))
```

In [29]:

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

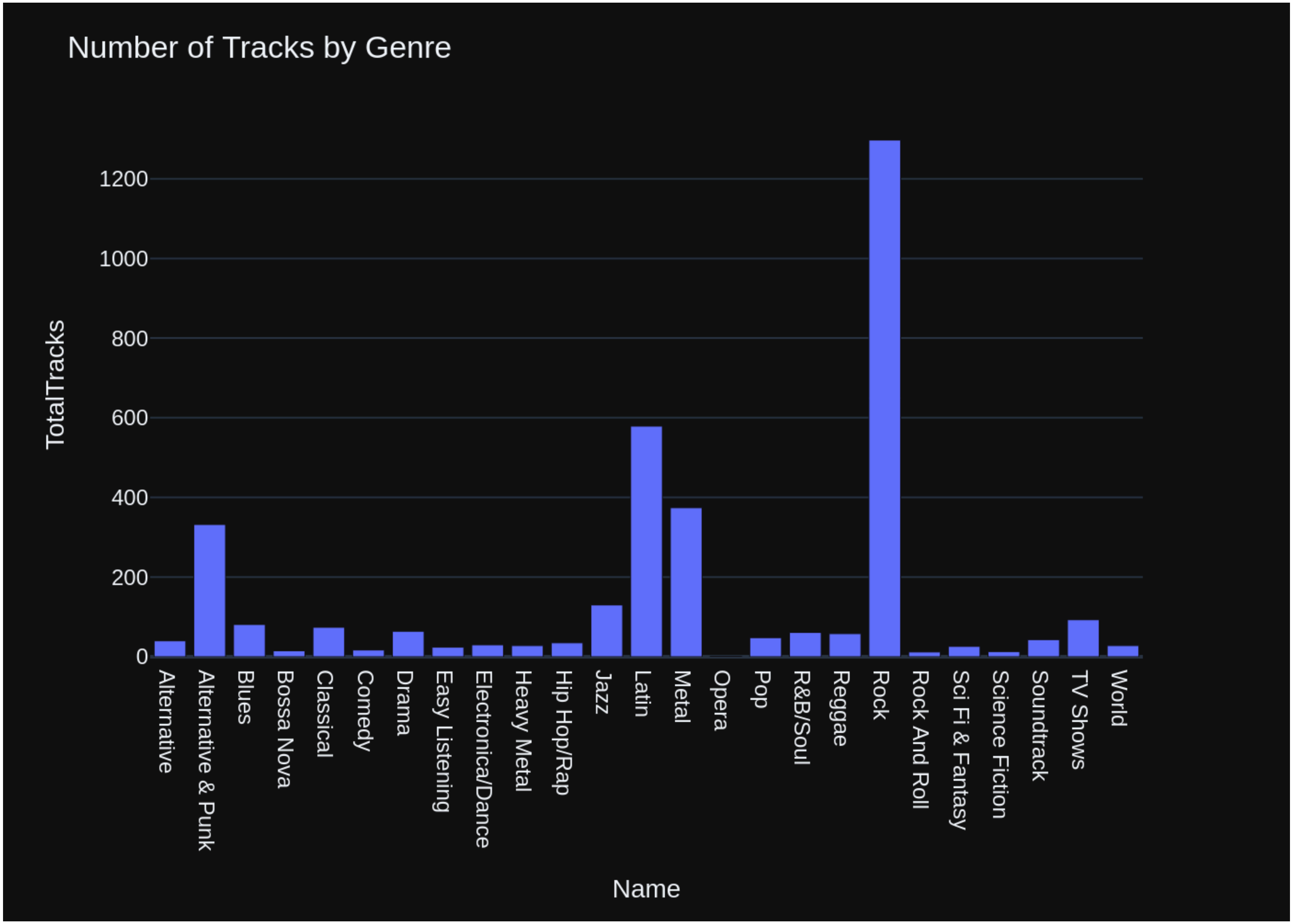
```
vn.ask(question=question)
```

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

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE 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 NU
LL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    By
tes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (Al
bumId) \r\n\t\tON DELETE NO ACTION 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 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 NU
LL,\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 IN
TEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackMediaType
Id 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    FO
REIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Tit
le NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES "ar
tists" (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===Respon
se Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any exp
lanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a s
pecific 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. I
f 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 *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'us
er', 'content': '  \n    List all albums and their corresponding artist names  \n'}, {'role': 'assistant',
'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.Arti
stId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    Find all tracks with a name containing "What" (c
ase-insensitive)\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%
\''}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assi
stant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total D
ESC\nLIMIT 5'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite database catalog?'}, {'r
ole': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'conte
nt': '  \n    Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT
c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.C
ustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '  \n    Get the total number of invoices for
each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as
TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId,
c.FirstName, c.LastName'}, {'role': 'user', 'content': '  \n    List all invoices with a total exceeding $1
0:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'role': 'use
r', 'content': '  \n    Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'conten
t': 'SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.Custo
merId = c.CustomerId\nGROUP BY c.CustomerId'}, {'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'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREM
ENT 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_Trac
kGenreId ON \"tracks\" (GenreId)\n\nCREATE TABLE \"genres\"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINC
REMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_trac
k\" (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  (Playl
istId, 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 AUTOI
NCREMENT 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\nCR
EATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\n\n===Additional Context \n\nIn the chinook databas
e 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 suffici
ent 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 e
xactly as it was given before. \n"}, {"role": "user", "content": "  \n    Find the top 5 most expensive tra
cks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitP
rice DESC\nLIMIT 5"}, {"role": "user", "content": "  \n    List all albums and their corresponding artist n
```

```
ames  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"alb
ums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "  \n    Find all t
racks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nF
ROM \"tracks\"\nWHERE Name LIKE '%what%'"}, {"role": "user", "content": "what are the top 5 countries that
customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"custome
rs\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "Can you list all table
s 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\nFROM \"invoices\"
i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "content": "
\n    Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT c.Fi
rstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.Cu
stomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "user", "content": "
\n    List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *\nFROM
\"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "  \n    Get the average invoice total for
each customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal
\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"rol
e": "user", "content": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) F
ROM \"customers\""}, {"role": "user", "content": "  \n    List all genres and the number of tracks in each
genre:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:12:30.836107598Z', 'message': {'role': 'assistant',
'content': 'SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "genres" g ON t.GenreId =
g.GenreId\nGROUP BY g.Name'}, 'done_reason': 'stop', 'done': True, 'total_duration': 88074006089, 'load_dur
ation': 1138168, 'prompt_eval_count': 1116, 'prompt_eval_duration': 80150057000, 'eval_count': 40, 'eval_du
ration': 7240237000}
SELECT g.Name, COUNT(t.TrackId) as TotalTracks
FROM "tracks" t
JOIN "genres" g ON t.GenreId = g.GenreId
GROUP BY g.Name
SELECT g.Name, COUNT(t.TrackId) as TotalTracks
FROM "tracks" t
JOIN "genres" g ON t.GenreId = g.GenreId
GROUP BY g.Name
                  Name  TotalTracks
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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 genres and the number of tracks in each genr
e:\n'\n\nThe DataFrame was produced using this query: SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM
\"tracks\" t\nJOIN \"genres\" g ON t.GenreId = g.GenreId\nGROUP BY g.Name\n\nThe following is information a
bout the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Name           object\nTotalTracks
int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the r
esults of the dataframe? Assume the data is in a pandas dataframe called 'df'. If there is only one value i
n the dataframe, use an Indicator. Respond with only Python code. Do not answer with any explanations -- ju
st the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:12:49.196728816Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\n\nfig = px.bar(df, x='Name', y='TotalTracks')\nfig.update_lay
out(title='Number of Tracks by Genre')\nfig.show()\n```"}, 'done_reason': 'stop', 'done': True, 'total_dura
tion': 18334012390, 'load_duration': 751110, 'prompt_eval_count': 186, 'prompt_eval_duration': 11186918000,
'eval_count': 41, 'eval_duration': 7012180000}
```


Number of Tracks by Genre
TotalTracks
1200
1000
800
600
400
200
0
Alternative
Alternative & Punk
Blues
Bossa Nova
Classical
Comedy
Drama
Easy Listening
Electronica/Dance
Heavy Metal
Hip Hop/Rap
Jazz
Latin
Metal
Opera
Pop
R&B/Soul
Reggae
Rock
Rock And Roll
Sci Fi & Fantasy
Science Fiction
Soundtrack
TV Shows
World
Name

```
Out[29]: ('SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "genres" g ON t.GenreId = g.GenreI
         d\nGROUP BY g.Name',
                           Name  TotalTracks
          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,
          Figure({
              'data': [{'alignmentgroup': 'True',
                        'hovertemplate': 'Name=%{x}<br>TotalTracks=%{y}<extra></extra>',
                        'legendgroup': '',
                        'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
                        'name': '',
                        'offsetgroup': '',
                        'orientation': 'v',
                        'showlegend': False,
                        'textposition': 'auto',
                        'type': 'bar',
                        'x': array(['Alternative', 'Alternative & Punk', 'Blues', 'Bossa Nova', 'Classical',
                                    'Comedy', 'Drama', 'Easy Listening', 'Electronica/Dance', 'Heavy Metal',
                                    'Hip Hop/Rap', 'Jazz', 'Latin', 'Metal', 'Opera', 'Pop', 'R&B/Soul',
                                    'Reggae', 'Rock', 'Rock And Roll', 'Sci Fi & Fantasy',
                                    'Science Fiction', 'Soundtrack', 'TV Shows', 'World'], dtype=object),
                        'xaxis': 'x',
                        'y': array([  40,  332,   81,   15,   74,   17,   64,   24,   30,   28,   35,  130,
                                     579,  374,    1,   48,   61,   58, 1297,   12,   26,   13,   43,   93,
                                      28]),
                        'yaxis': 'y'}],
              'layout': {'barmode': 'relative',
                         'legend': {'tracegroupgap': 0},
                         'margin': {'t': 60},
                         'template': '...',
                         'title': {'text': 'Number of Tracks by Genre'},
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Name'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalTracks'}}}
          }))
```

In [30]:

```python
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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE TABLE "track
s"\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(22
0),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NUL
L,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION,\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_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE INDE
X IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\n
CREATE TABLE "genres"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR
(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    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_trac
k"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_Playlis
tTrack 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 (TrackId) REFERENCES "tracks" (Trac
kId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (Ar
tistId)\n\nCREATE TABLE "playlists"\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 database invoice means order\n\n===Re
sponse 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 st
rings 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). \n
5. If the question has been asked and answered before, please repeat the answer exactly as it was given bef
ore. \n'}, {'role': 'user', 'content': '  \n    List all genres and the number of tracks in each genr
e:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\n
JOIN "genres" g ON t.GenreId = g.GenreId\nGROUP BY g.Name'}, {'role': 'user', 'content': '  \n    Find the
top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "t
racks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Find all tracks with a nam
e containing "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHER
E Name LIKE \'%what%\''}, {'role': 'user', 'content': '  \n    List all albums and their corresponding arti
st names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM
"albums" a\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'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': 'what are the top 5 countries that customers come fro
m?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Count
ry\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    List all invoices with a total exc
eeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'rol
e': 'user', 'content': "  \n    List all employees and their reporting manager's name (if any):\n"}, {'rol
```

```
e': 'assistant', 'content': 'SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS Ma
nagerFirstName,\n       COALESCE(e2.LastName, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "e
mployees" e2 ON e.ReportsTo = e2.EmployeeId'}, {'role': 'user', 'content': 'How many customers are there'},
{'role': 'assistant', 'content': 'SELECT COUNT(*) FROM "customers"'}, {'role': 'user', 'content': '  \n
Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerI
d, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nG
ROUP BY c.CustomerId'}, {'role': 'user', 'content': '  \n    Get all genres that do not have any tracks ass
ociated with them:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE TABLE \"tr
acks\"\r\n(\r\n    TrackId INTEGER 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    GenreId INTEGER,\r\n    Composer NV
ARCHAR(220),\r\n    Milliseconds 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\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDA
TE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE N
O 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 \"tra
cks\" (AlbumId)\n\nCREATE TABLE \"genres\"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(120)\r\n)\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 INTEGER  NOT NULL,\r\n    FOREIGN K
EY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCRE
ATE 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) REFER
ENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Track
Id) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IF
K_AlbumArtistId ON \"albums\" (ArtistId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMA
RY 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 g
enerate 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 intermedi
ate SQL query to find the distinct 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. Pleas
e use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat th
e 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.TrackId) as T
```

```
otalTracks\nFROM \"tracks\" t\nJOIN \"genres\" g ON t.GenreId = g.GenreId\nGROUP BY g.Name"}, {"role": "use
r", "content": "  \n    Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistan
t", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content":
"  \n    Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "con
tent": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%what%'"}, {"role": "user", "content": "  \n    List al
l albums and their corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.Ar
tistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"rol
e": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "c
ontent": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": "what are the to
p 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as To
tal\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "
\n    List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *\nFROM
\"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "  \n    List all employees and their repo
rting manager's name (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstName, e.LastName, \n
COALESCE(e2.FirstName, 'N/A') AS ManagerFirstName,\n       COALESCE(e2.LastName, 'N/A') AS ManagerLastName
\nFROM \"employees\" e\nLEFT JOIN \"employees\" e2 ON e.ReportsTo = e2.EmployeeId"}, {"role": "user", "cont
ent": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers
\""}, {"role": "user", "content": "  \n    Get the average invoice total for each customer:\n"}, {"role":
"assistant", "content": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"invoices\" i\nJOIN
\"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "  \n
Get all genres that do not have any tracks associated with them:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:14:15.261276153Z', 'message': {'role': 'assistant',
'content': 'SELECT *\nFROM "genres"\nWHERE GenreId NOT IN (SELECT GenreId FROM "tracks")'}, 'done_reason':
'stop', 'done': True, 'total_duration': 85956769401, 'load_duration': 798451, 'prompt_eval_count': 1137, 'p
rompt_eval_duration': 81787132000, 'eval_count': 20, 'eval_duration': 3500534000}
SELECT *
FROM "genres"
WHERE GenreId NOT IN (SELECT GenreId FROM "tracks")
SELECT *
FROM "genres"
WHERE GenreId NOT IN (SELECT GenreId FROM "tracks")
Empty DataFrame
Columns: [GenreId, Name]
Index: []
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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    Get all genres that do not have any tracks associated wi
```

```
th them:\n'\n\nThe DataFrame was produced using this query: SELECT *\nFROM \"genres\"\nWHERE GenreId NOT IN
(SELECT GenreId FROM \"tracks\")\n\nThe following is information about the resulting pandas DataFrame 'df':
\nRunning df.dtypes gives:\n GenreId    object\nName       object\ndtype: object"}, {"role": "user", "conte
nt": "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."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:14:34.862269904Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nfig = px.bar(df, x='Name', y='GenreId')\nfig.update_layout(ti
tle_text='Genres without Tracks', title_font_size=16)\nfig.show()"}, 'done_reason': 'stop', 'done': True,
'total_duration': 19598886898, 'load_duration': 845167, 'prompt_eval_count': 165, 'prompt_eval_duration': 1
1501759000, 'eval_count': 45, 'eval_duration': 7965641000}
```

```
Out[30]: ('SELECT *\nFROM "genres"\nWHERE GenreId NOT IN (SELECT GenreId FROM "tracks")',
          Empty DataFrame
          Columns: [GenreId, Name]
          Index: [],
          Figure({
              'data': [{'domain': {'x': [0.0, 1.0], 'y': [0.0, 1.0]},
                        'hovertemplate': 'GenreId=%{label}<extra></extra>',
                        'labels': array([], dtype=object),
                        'legendgroup': '',
                        'name': '',
                        'showlegend': True,
                        'type': 'pie'}],
              'layout': {'legend': {'tracegroupgap': 0}, 'margin': {'t': 60}, 'template': '...'}
          }))
```

```
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
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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    Firs
tName 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 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    InvoiceLineI
d INTEGER PRIMARY KEY AUTOINCREMENT 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 (I
nvoiceId) 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 NVARCH
AR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTE
GER,\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_tra
ck"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_Playli
stTrack PRIMARY KEY  (PlaylistId, 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 (TrackId) REFERENCES "tracks" (Tra
ckId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INT
EGER 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 N
O ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "playl
ists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n
CREATE 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 NUMER
IC(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 DEL
ETE 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 t
he provided context is sufficient, please generate a valid SQL query without any explanations for the quest
ion. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a par
ticular column, please generate an intermediate SQL query to find the distinct strings in that column. Prep
```

```
end the query with a comment saying 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 bee
n asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'use
r', 'content': '  \n    Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'cont
ent': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "custom
ers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user',
'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assistant', 'content': 'SEL
ECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5'}, {'rol
e': 'user', 'content': 'How many customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) F
ROM "customers"'}, {'role': 'user', 'content': '  \n    Get the average invoice total for each custome
r:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoi
ces" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'conte
nt': '  \n    Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT
c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.C
ustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '  \n    List all invoices with a total exceed
ing $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'role':
'user', 'content': "  \n    List all employees and their reporting manager's name (if any):\n"}, {'role':
'assistant', 'content': 'SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS Manage
rFirstName,\n       COALESCE(e2.LastName, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "emplo
yees" e2 ON e.ReportsTo = e2.EmployeeId'}, {'role': 'user', 'content': '  \n    List all albums and their c
orresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS A
rtistName\nFROM "albums" a\nJOIN "artists" 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.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "genres" g ON t.GenreId = g.GenreId\nGROUP B
Y g.Name'}, {'role': 'user', 'content': '  \n    Find the top 5 most expensive tracks (based on unit pric
e):\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'r
ole': 'user', 'content': '  \n    List all customers who have not placed any orders:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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    InvoiceL
ineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGE
R  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KE
Y (InvoiceId) REFERENCES \"invoices\" (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 ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE \"employees\"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Last
Name NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    Rep
ortsTo INTEGER,\r\n    BirthDate 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(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsT
o) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TAB
LE \"playlist_track\"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CO
NSTRAINT 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) RE
FERENCES \"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\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (Supp
ortRepId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOIN
CREMENT 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_Invo
iceCustomerId ON \"invoices\" (CustomerId)\n\n\n===Additional Context \n\nIn the chinook database invoice m
eans order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQ
L query without any explanations for the question. \n2. If the provided context is almost sufficient but re
quires 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 re
levant 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 eac
h customer\n"}, {"role": "assistant", "content": "SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as Tot
alInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerI
d, c.FirstName, c.LastName"}, {"role": "user", "content": "what are the top 5 countries that customers come
from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY
Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "How many customers are there"}, {"rol
e": "assistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user", "content": "  \n    Get
```

```
the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, AV
G(i.Total) as AvgInvoiceTotal\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGR
OUP BY c.CustomerId"}, {"role": "user", "content": "  \n    Find the total number of invoices per countr
y:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"inv
oices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "con
tent": "  \n    List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT
*\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "  \n    List all employees and th
eir reporting manager's name (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstName, e.LastNam
e, \n       COALESCE(e2.FirstName, 'N/A') AS ManagerFirstName,\n       COALESCE(e2.LastName, 'N/A') AS Mana
gerLastName\nFROM \"employees\" e\nLEFT JOIN \"employees\" e2 ON e.ReportsTo = e2.EmployeeId"}, {"role": "u
ser", "content": "  \n    List all albums and their corresponding artist names  \n"}, {"role": "assistant",
"content": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" 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.TrackId) as TotalTracks\nFROM \"t
racks\" t\nJOIN \"genres\" g ON t.GenreId = g.GenreId\nGROUP BY g.Name"}, {"role": "user", "content": "  \n
Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT *
\nFROM \"tracks\"\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": "  \n    List all custom
ers who have not placed any orders:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:16:39.059728199Z', 'message': {'role': 'assistant',
'content': 'SELECT *\nFROM "customers" c\nWHERE c.CustomerId NOT IN (SELECT CustomerId FROM "invoices")'},
'done_reason': 'stop', 'done': True, 'total_duration': 124113900157, 'load_duration': 977903, 'prompt_eval_
count': 1629, 'prompt_eval_duration': 119325945000, 'eval_count': 23, 'eval_duration': 4115914000}
SELECT *
FROM "customers" c
WHERE c.CustomerId NOT IN (SELECT CustomerId FROM "invoices")
SELECT *
FROM "customers" c
WHERE c.CustomerId NOT IN (SELECT CustomerId FROM "invoices")
Empty DataFrame
Columns: [CustomerId, FirstName, LastName, Company, Address, City, State, Country, PostalCode, Phone, Fax,
Email, SupportRepId]
Index: []
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 who have not placed any orders:\n'\n
\nThe DataFrame was produced using this query: SELECT *\nFROM \"customers\" c\nWHERE c.CustomerId NOT IN (S
ELECT CustomerId FROM \"invoices\")\n\nThe following is information about the resulting pandas DataFrame 'd
```

```
f': \nRunning df.dtypes gives:\n CustomerId       object\nFirstName        object\nLastName         object\nCo
mpany         object\nAddress         object\nCity            object\nState           object\nCountry      
object\nPostalCode      object\nPhone           object\nFax             object\nEmail           object\nSup
portRepId    object\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 on
ly one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with any expl
anations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:17:02.194986525Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nimport pandas as pd\n\nfig = px.bar(df, x='CustomerId', y='',
title='Customers without Orders')\nfig.update_layout(yaxis_title='Count')\n\nfig.show()\n```"}, 'done_reaso
n': 'stop', 'done': True, 'total_duration': 23132701583, 'load_duration': 793243, 'prompt_eval_count': 212,
'prompt_eval_duration': 14738733000, 'eval_count': 48, 'eval_duration': 8258548000}
```

Out[31]:

```
('SELECT *\nFROM "customers" c\nWHERE c.CustomerId NOT IN (SELECT CustomerId FROM "invoices")',
 Empty DataFrame
 Columns: [CustomerId, FirstName, LastName, Company, Address, City, State, Country, PostalCode, Phone, Fa
x, Email, SupportRepId]
 Index: [],
 Figure({
     'data': [{'domain': {'x': [0.0, 1.0], 'y': [0.0, 1.0]},
               'hovertemplate': 'CustomerId=%{label}<extra></extra>',
               'labels': array([], dtype=object),
               'legendgroup': '',
               'name': '',
               'showlegend': True,
               'type': 'pie'}],
     'layout': {'legend': {'tracegroupgap': 0}, 'margin': {'t': 60}, 'template': '...'}
 }))
```

In [32]:

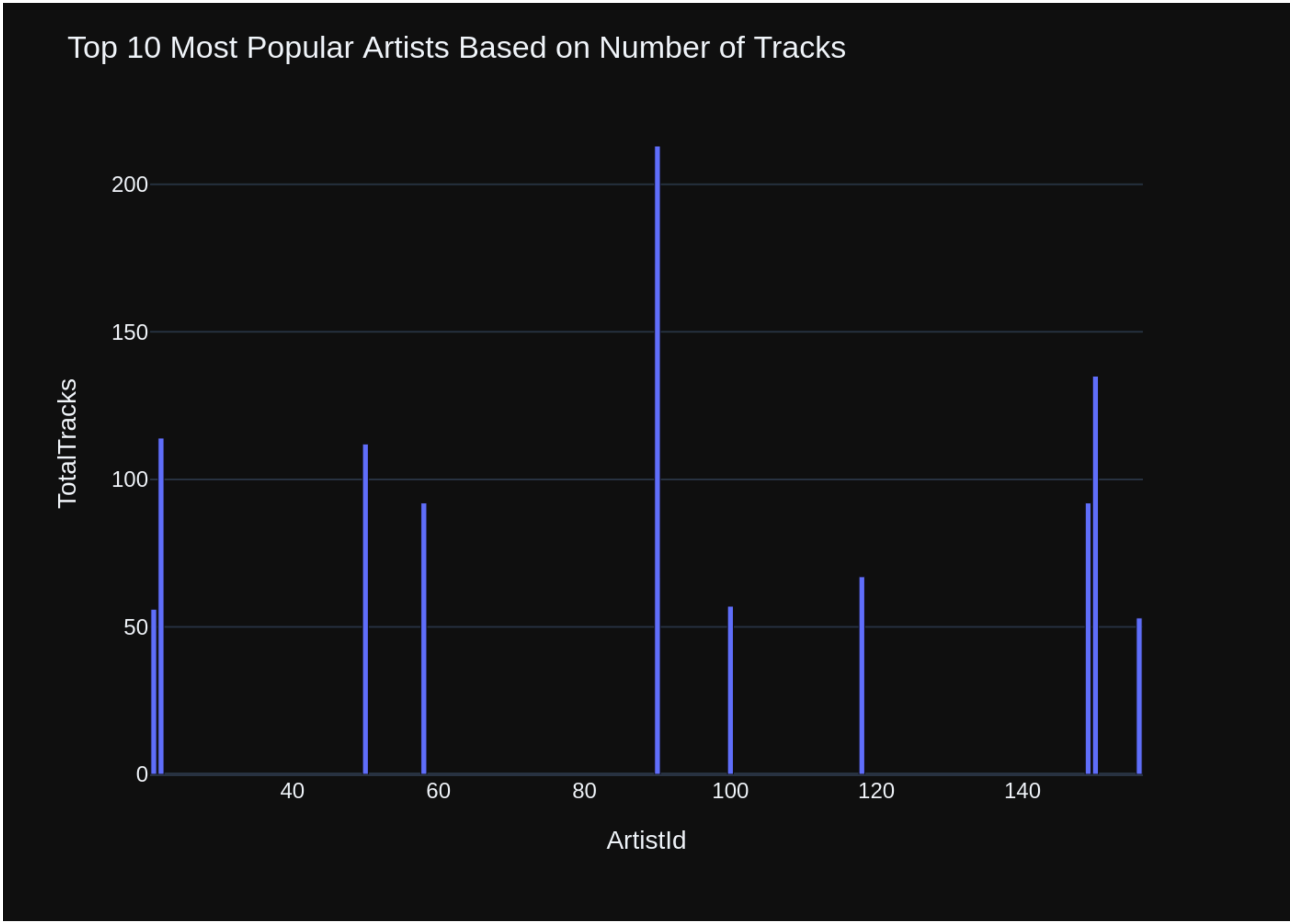
```python
question = """
    There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums
    Can you find the top 10 most popular artists based on the number of tracks
"""

vn.ask(question=question)
```

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

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE 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 NU
LL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    By
tes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (Al
bumId) \r\n\t\tON DELETE NO ACTION 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 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    Arti
stId 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 AUTOINCR
EMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\n
CREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "playlists"\r\n(\r\n    PlaylistId INTE
GER 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 "playlis
t_track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_P
laylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlists" (Pl
aylistId) \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 s
ufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provide
d context is almost sufficient but requires knowledge of a specific string in a particular column, please g
enerate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a com
ment 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': '  \n    Li
st all albums and their corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, 
a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'rol
e': 'user', 'content': '  \n    List all genres and the number of tracks in each genre:\n'}, {'role': 'assi
stant', 'content': 'SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "genres" g ON t.G
enreId = g.GenreId\nGROUP BY g.Name'}, {'role': 'user', 'content': '  \n    Find the top 5 most expensive t
racks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nORDER BY UnitP
rice DESC\nLIMIT 5'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come fro
m?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Count
ry\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Find all tracks with a name contai
ning "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name L
IKE \'%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', 'co
ntent': '  \n    Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELE
CT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = 
c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '  \n    Get the total number of invoices f
or each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId)
as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Customer
Id, c.FirstName, c.LastName'}, {'role': 'user', 'content': '  \n    Get the average invoice total for each
customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM
"invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user',
'content': '  \n    List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SE
LECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'role': 'user', 'content': ' \n    There are 3 tables: art
ists, 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 number of tracks\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREM
ENT 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\tO
N DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n    ArtistId INTEGER PRIMAR
Y 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 NUL
L,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) R
EFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (T
rackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDE
X 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 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 context is almost sufficient but requires knowledge of a specific st
ring 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 insufficie
nt, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the quest
```

```
ion has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"ro
le": "user", "content": "  \n    List all albums and their corresponding artist names  \n"}, {"role": "assi
stant", "content": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\"
ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "  \n    List all genres and the number of tr
acks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name, COUNT(t.TrackId) as TotalTracks\n
FROM \"tracks\" t\nJOIN \"genres\" g ON t.GenreId = g.GenreId\nGROUP BY g.Name"}, {"role": "user", "conten
t": "  \n    Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "conten
t": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": "what are t
he top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*)
as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content":
"  \n    Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "con
tent": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%what%'"}, {"role": "user", "content": "Can you list al
l tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_maste
r WHERE type='table'"}, {"role": "user", "content": "  \n    Find the total number of invoices per countr
y:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"inv
oices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "con
tent": "  \n    Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "S
ELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\"
c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "user", "conte
nt": "  \n    Get the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELEC
T c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId
= c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "  \n    List all invoices with a tota
l exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"invoices\"\nWHERE Total > 10.00"},
{"role": "user", "content": " \n    There are 3 tables: artists, albums and tracks, where albums and artist
s are linked by ArtistId, albums and tracks are linked by AlbumId,\n    Can you find the top 10 most popula
r artists based on the number of tracks\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:17:57.212298304Z', 'message': {'role': 'assistant',
'content': 'SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "albums" a ON t.Album
Id = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10'}, 'done_reason': 'stop', 'done':
True, 'total_duration': 54935166949, 'load_duration': 849460, 'prompt_eval_count': 1203, 'prompt_eval_durat
ion': 44540445000, 'eval_count': 54, 'eval_duration': 9722163000}
SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks
FROM "tracks" t
JOIN "albums" a ON t.AlbumId = a.AlbumId
GROUP BY a.ArtistId
ORDER BY TotalTracks DESC
LIMIT 10
SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks
FROM "tracks" t
JOIN "albums" a ON t.AlbumId = a.AlbumId
GROUP BY a.ArtistId
```

```
ORDER BY TotalTracks DESC
LIMIT 10
   ArtistId  TotalTracks
0        90          213
1       150          135
2        22          114
3        50          112
4        58           92
5       149           92
6       118           67
7       100           57
8        21           56
9       156           53
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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    There are 3 tables: artists, albums and tracks, where alb
ums 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'\n\nThe DataFrame was produced using this query: SE
LECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM \"tracks\" t\nJOIN \"albums\" a ON t.AlbumId = a.Alb
umId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10\n\nThe following is information about the re
sulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n ArtistId       int64\nTotalTracks    int64\ndty
pe: 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 data
frame, use an Indicator. Respond with only Python code. Do not answer with any explanations -- just the cod
e."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:18:22.750156989Z', 'message': {'role': 'assistant',
'content': '```\nimport plotly.express as px\nfig = px.bar(df, x=\'ArtistId\', y=\'TotalTracks\')\nfig.upda
te_layout(title_text="Top 10 Most Popular Artists Based on Number of Tracks")\nfig.show()\n```'}, 'done_rea
son': 'stop', 'done': True, 'total_duration': 25517414045, 'load_duration': 733113, 'prompt_eval_count': 23
9, 'prompt_eval_duration': 16688088000, 'eval_count': 49, 'eval_duration': 8695452000}
```


Top 10 Most Popular Artists Based on Number of Tracks
TotalTracks
200
150
100
50
0
40
60
80
100
120
140
ArtistId

```
Out[32]: ('SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "albums" a ON t.AlbumId = a.Al
bumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10',
    ArtistId  TotalTracks
 0        90          213
 1       150          135
 2        22          114
 3        50          112
 4        58           92
 5       149           92
 6       118           67
 7       100           57
 8        21           56
 9       156           53,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'ArtistId=%{x}<br>TotalTracks=%{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([ 90, 150,  22,  50,  58, 149, 118, 100,  21, 156]),
               'xaxis': 'x',
               'y': array([213, 135, 114, 112,  92,  92,  67,  57,  56,  53]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'title': {'text': 'Top 10 Most Popular Artists Based on Number of Tracks'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'ArtistId'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalTracks'}}}
 }))
```

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

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

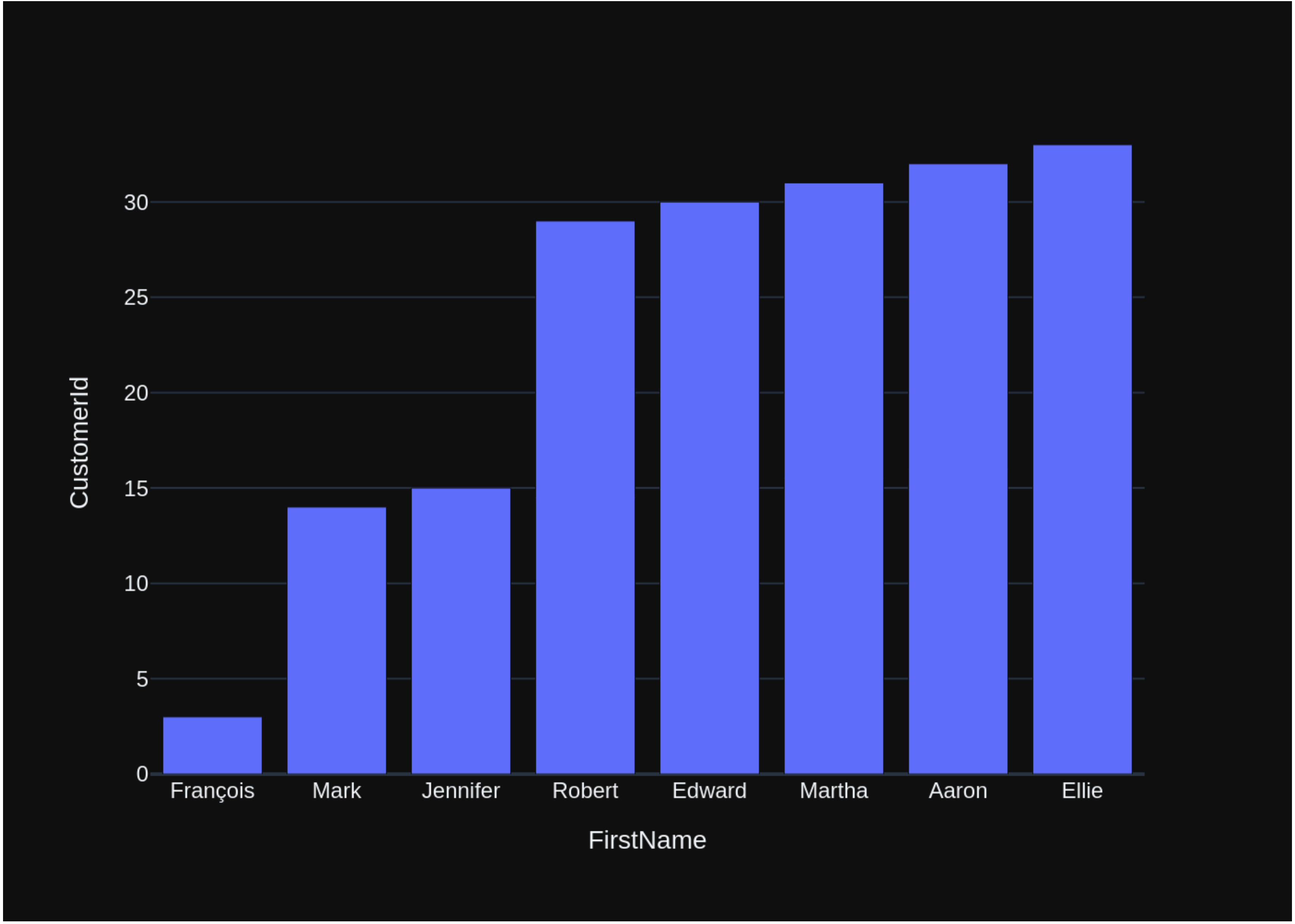
```
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCRE
ATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName N
VARCHAR(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    Post
alCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NUL
L,\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 P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NU
LL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(4
0),\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 O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "e
mployees"\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(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENCES "employee
s" (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 ACTIO
N,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "playlist_track"\r\n(\r\n    PlaylistId I
NTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (Playli
stId, TrackId),\r\n    FOREIGN KEY (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 A
CTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE T
ABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  N
OT 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 sufficien
t but requires knowledge of a specific string in a particular column, please generate an intermediate SQL q
uery 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 th
e 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 custom
ers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGR
OUP BY Country\nORDER BY Total DESC\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 Tot
```

```
alInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'r
ole': 'user', 'content': '  \n    Get the total number of invoices for each customer\n'}, {'role': 'assista
nt', 'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJO
IN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role':
'user', 'content': 'How many customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM
"customers"'}, {'role': 'user', 'content': '  \n    Get the average invoice total for each customer:\n'},
{'role': 'assistant', 'content': 'SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\n
JOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': "
\n    List all employees and their reporting manager's name (if any):\n"}, {'role': 'assistant', 'content':
'SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName, \'N/A\') AS ManagerFirstName,\n       COAL
ESCE(e2.LastName, \'N/A\') AS ManagerLastName\nFROM "employees" e\nLEFT JOIN "employees" e2 ON e.ReportsTo
= e2.EmployeeId'}, {'role': 'user', 'content': '  \n    List all invoices with a total exceeding $10:\n'},
{'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'role': 'user', 'conte
nt': '  \n    List all albums and their corresponding artist names  \n'}, {'role': 'assistant', 'content':
'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN "artists" ar ON a.ArtistId = ar.A
rtistId'}, {'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': 'SE
LECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n     List all c
ustomers from Canada and their email addresses:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nC
REATE TABLE \"customers\"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstNa
me NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Addr
ess 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 INT
EGER 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(2
0)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGE
R,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(4
0),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NV
ARCHAR(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_i
tems\"\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\" (TrackI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employee
s\" (ReportsTo)\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 NULL,\r\n    FOREIGN KEY (ArtistId) R
EFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Co
ntext \n\nIn the chinook database 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 p
rovided context is almost sufficient but requires knowledge of a specific string in a particular column, pl
ease 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 ar
e the top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT
(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "conte
nt": "  \n    Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT 
c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId =
c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "content": "  \n    Get the total number of invoices f
or each customer\n"}, {"role": "assistant", "content": "SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) 
as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Cust
omerId, c.FirstName, c.LastName"}, {"role": "user", "content": "How many customers are there"}, {"role": "a
ssistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user", "content": "  \n    Get the a
verage invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, AVG(i.T
otal) as AvgInvoiceTotal\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP B
Y c.CustomerId"}, {"role": "user", "content": "  \n    List all employees and their reporting manager's nam
e (if any):\n"}, {"role": "assistant", "content": "SELECT e.FirstName, e.LastName, \n       COALESCE(e2.Fir
stName, 'N/A') AS ManagerFirstName,\n       COALESCE(e2.LastName, 'N/A') AS ManagerLastName\nFROM \"employe
es\" e\nLEFT JOIN \"employees\" e2 ON e.ReportsTo = e2.EmployeeId"}, {"role": "user", "content": "  \n    L
ist all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"invoic
es\"\nWHERE Total > 10.00"}, {"role": "user", "content": "  \n    List all albums and their corresponding a
rtist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFRO
M \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "Can you lis
t all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_m
```

```
aster WHERE type='table'"}, {"role": "user", "content": "  \n    Find the top 5 most expensive tracks (base
d on unit price):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC
\nLIMIT 5"}, {"role": "user", "content": "  \n     List all customers from Canada and their email addresse
s:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:20:12.73748968Z', 'message': {'role': 'assistant',
'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, c.Email\nFROM "customers" c\nWHERE c.Country =
\'Canada\''}, 'done_reason': 'stop', 'done': True, 'total_duration': 109889690149, 'load_duration': 713017,
'prompt_eval_count': 1436, 'prompt_eval_duration': 104185699000, 'eval_count': 28, 'eval_duration': 5034105
000}
SELECT c.CustomerId, c.FirstName, c.LastName, c.Email
FROM "customers" c
WHERE c.Country = 'Canada'
SELECT c.CustomerId, c.FirstName, c.LastName, c.Email
FROM "customers" c
WHERE c.Country = 'Canada'
   CustomerId FirstName  LastName                   Email
0           3  François  Tremblay     ftremblay@gmail.com
1          14      Mark   Philips      mphilips12@shaw.ca
2          15  Jennifer  Peterson     jenniferp@rogers.ca
3          29    Robert     Brown        robbrown@shaw.ca
4          30    Edward   Francis     edfrancis@yachoo.ca
5          31    Martha      Silk    marthasilk@gmail.com
6          32     Aaron  Mitchell  aaronmitchell@yahoo.ca
7          33     Ellie  Sullivan  ellie.sullivan@shaw.ca
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 addresse
s:\n'\n\nThe DataFrame was produced using this query: SELECT c.CustomerId, c.FirstName, c.LastName, c.Email
\nFROM \"customers\" c\nWHERE c.Country = 'Canada'\n\nThe following is information about the resulting pand
as DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId     int64\nFirstName     object\nLastName     o
bject\nEmail         object\ndtype: 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 pandas dataframe called 'df'. If the
re is only one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with
any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:20:34.649627914Z', 'message': {'role': 'assistant',
'content': '```\nimport plotly.express as px\n\nfig = px.bar(df, x=\'CustomerName\', y=\'Email\')\nfig.upda
```

```
te_layout(title="Canada Customers and Email", xaxis_title="Customer Name", yaxis_title="Email")\n\nfig.show
()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 21886520385, 'load_duration': 685566, 'pr
ompt_eval_count': 179, 'prompt_eval_duration': 12488636000, 'eval_count': 53, 'eval_duration': 9263166000}
```



```
Out[33]: ('SELECT c.CustomerId, c.FirstName, c.LastName, c.Email\nFROM "customers" c\nWHERE c.Country = \'Canada
         \'',
             CustomerId FirstName  LastName                    Email
          0           3  François  Tremblay     ftremblay@gmail.com
          1          14      Mark   Philips      mphilips12@shaw.ca
          2          15  Jennifer  Peterson     jenniferp@rogers.ca
          3          29    Robert     Brown        robbrown@shaw.ca
          4          30    Edward   Francis     edfrancis@yachoo.ca
          5          31    Martha      Silk    marthasilk@gmail.com
          6          32     Aaron  Mitchell  aaronmitchell@yahoo.ca
          7          33     Ellie  Sullivan  ellie.sullivan@shaw.ca,
          Figure({
              'data': [{'alignmentgroup': 'True',
                        'hovertemplate': 'FirstName=%{x}<br>CustomerId=%{y}<extra></extra>',
                        'legendgroup': '',
                        'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
                        'name': '',
                        'offsetgroup': '',
                        'orientation': 'v',
                        'showlegend': False,
                        'textposition': 'auto',
                        'type': 'bar',
                        'x': array(['François', 'Mark', 'Jennifer', 'Robert', 'Edward', 'Martha', 'Aaron',
                                    'Ellie'], dtype=object),
                        'xaxis': 'x',
                        'y': array([ 3, 14, 15, 29, 30, 31, 32, 33]),
                        'yaxis': 'y'}],
              'layout': {'barmode': 'relative',
                         'legend': {'tracegroupgap': 0},
                         'margin': {'t': 60},
                         'template': '...',
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'FirstName'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'CustomerId'}}}
          }))
```

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

         vn.ask(question=question)
```

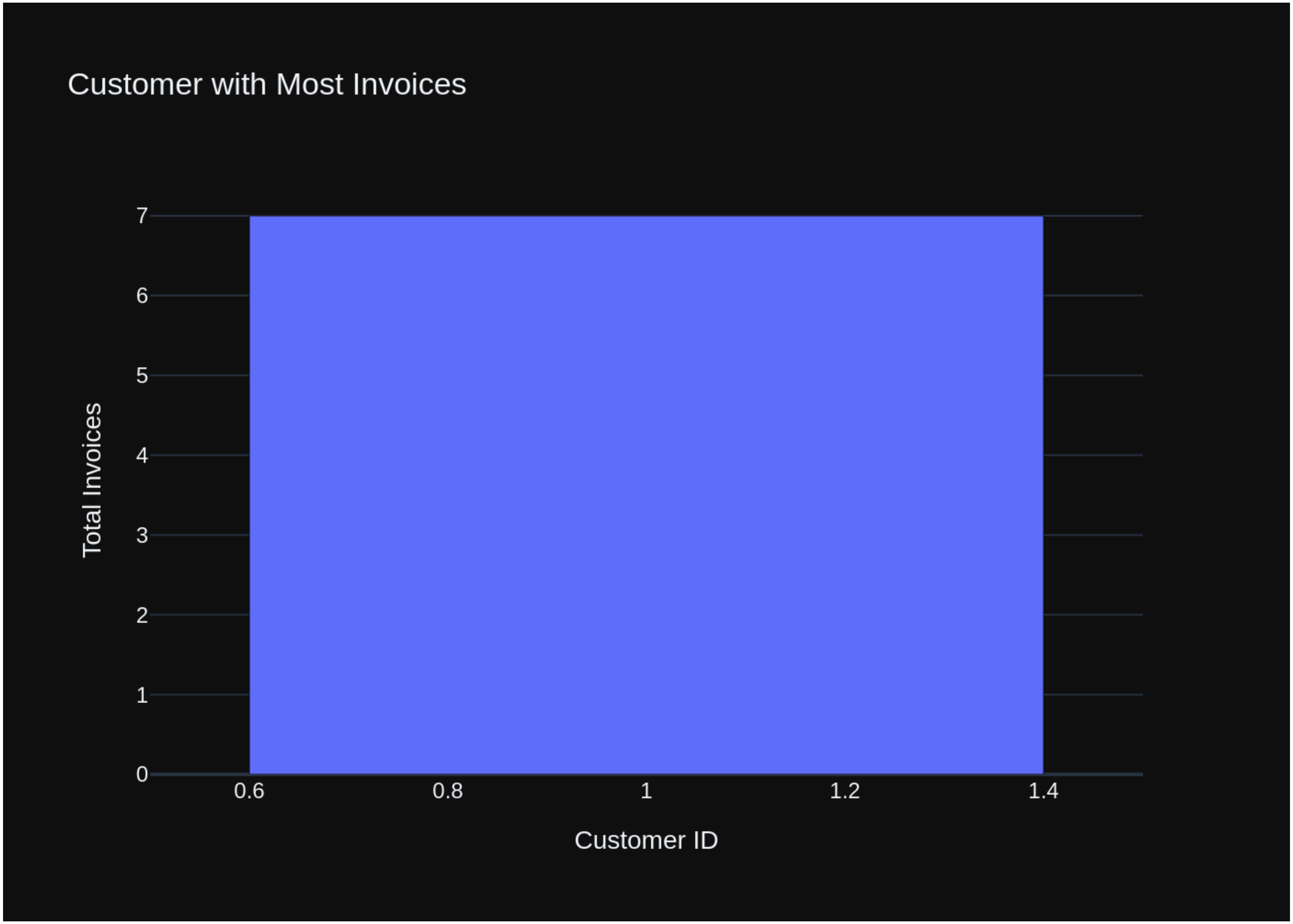
```
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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 KE
Y AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    Unit
Price 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) REFERE
NCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLin
eTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NUL
L,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHA
R(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax N
VARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportR
epId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IN
DEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employees"\r\n(\r\n    Employee
Id INTEGER PRIMARY KEY AUTOINCREMENT 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 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 DEL
ETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\n
CREATE 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 NUMER
IC(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 DEL
ETE 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 que
ry without any explanations for the question. \n2. If the provided context is almost sufficient but require
s 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 p
rovided context is insufficient, 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 i
t was given before. \n'}, {'role': 'user', 'content': '  \n    Get the total number of invoices for each cu
stomer\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalIn
voices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.Firs
```

```
tName, c.LastName'}, {'role': 'user', 'content': '  \n    Find the total number of invoices per countr
y:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invo
ices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'conten
t': '  \n    Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT
c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.Cu
stomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': '  \n    List all invoices with a total exce
eding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'rol
e': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'assistant', 'co
ntent': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT
5'}, {'role': 'user', 'content': '  \n    Find the top 5 most expensive tracks (based on unit price):\n'},
{'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'us
er', 'content': 'How many customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) FROM "cu
stomers"'}, {'role': 'user', 'content': '  \n     List all customers from Canada and their email addresse
s:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, c.Email\nFROM "cust
omers" c\nWHERE c.Country = \'Canada\''}, {'role': 'user', 'content': "  \n    List all employees and their
reporting manager's name (if any):\n"}, {'role': 'assistant', 'content': 'SELECT e.FirstName, e.LastName,
\n       COALESCE(e2.FirstName, \'N/A\') AS ManagerFirstName,\n       COALESCE(e2.LastName, \'N/A\') AS Man
agerLastName\nFROM "employees" e\nLEFT JOIN "employees" e2 ON e.ReportsTo = e2.EmployeeId'}, {'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 AlbumId,\n    Can you find the top 10 most popular artists bas
ed on the number of tracks\n'}, {'role': 'assistant', 'content': 'SELECT a.ArtistId, COUNT(t.TrackId) as To
talTracks\nFROM "tracks" t\nJOIN "albums" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTr
acks DESC\nLIMIT 10'}, {'role': 'user', 'content': '  \n     Find the customer with the most invoices \n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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_InvoiceLineInv
oiceId 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 (Tr
ackId) 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 NVARCH
AR(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(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREI
GN 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 NUL
L,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    Bir
thDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    St
ate 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 NUL
L,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Byt
es INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES \"albums\" (A
lbumId) \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 \"me
dia_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 s
ufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provide
d context is almost sufficient but requires knowledge of a specific string in a particular column, please g
enerate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a com
ment saying intermediate_sql \n3. 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 has been asked and answered befo
re, 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 c.FirstName, 
c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId 
= c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "user", "content": "  \n    Find 
the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.
InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROU
P BY c.Country"}, {"role": "user", "content": "  \n    Get the average invoice total for each custome
r:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"invo
ices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "c
ontent": "  \n    List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELE
CT *\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "what are the top 5 countries t
hat customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as Total\nFROM \"cus
tomers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "  \n    Find the t
op 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"t
racks\"\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": "How many customers are there"}, 
{"role": "assistant", "content": "SELECT COUNT(*) FROM \"customers\""}, {"role": "user", "content": "  \n    
List all customers from Canada and their email addresses:\n"}, {"role": "assistant", "content": "SELECT c.C
```

```
ustomerId, c.FirstName, c.LastName, c.Email\nFROM \"customers\" c\nWHERE c.Country = 'Canada'"}, {"role":
"user", "content": "  \n    List all employees and their reporting manager's name (if any):\n"}, {"role":
"assistant", "content": "SELECT e.FirstName, e.LastName, \n       COALESCE(e2.FirstName, 'N/A') AS ManagerF
irstName,\n       COALESCE(e2.LastName, 'N/A') AS ManagerLastName\nFROM \"employees\" e\nLEFT JOIN \"employ
ees\" e2 ON e.ReportsTo = e2.EmployeeId"}, {"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 AlbumI
d,\n    Can you find the top 10 most popular artists based on the number of tracks\n"}, {"role": "assistan
t", "content": "SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM \"tracks\" t\nJOIN \"albums\" a ON
t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10"}, {"role": "user", "conten
t": "  \n     Find the customer with the most invoices \n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:22:37.843597752Z', 'message': {'role': 'assistant',
'content': 'SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c
ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1'}, 'done_reaso
n': 'stop', 'done': True, 'total_duration': 123098663782, 'load_duration': 955789, 'prompt_eval_count': 154
0, 'prompt_eval_duration': 112511221000, 'eval_count': 53, 'eval_duration': 9918587000}
SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalInvoices DESC
LIMIT 1
SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalInvoices DESC
LIMIT 1
   CustomerId  TotalInvoices
0           1              7
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 with the most invoices \n'\n\nThe Dat
aFrame was produced using this query: SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invo
ices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoice
s DESC\nLIMIT 1\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dty
pes gives:\n CustomerId       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 pan
```

```
das 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."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:23:01.969478596Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\n\nfig = px.bar(df, x='CustomerId', y='TotalInvoices', title
='Customer with Most Invoices')\n\nfig.update_xaxes(title='Customer ID')\nfig.update_yaxes(title='Total Inv
oices')\n\nfig.show()\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 24100284850, 'load_dur
ation': 675716, 'prompt_eval_count': 198, 'prompt_eval_duration': 13913372000, 'eval_count': 58, 'eval_dura
tion': 10046384000}
```


Customer with Most Invoices
Total Invoices
0
1
2
3
4
5
6
7
0.6
0.8
1
1.2
1.4
Customer ID

Out[34]:

```
('SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.Cus
tomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1',
    CustomerId  TotalInvoices
 0           1              7,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>TotalInvoices=%{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array([1]),
               'xaxis': 'x',
               'y': array([7]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Customer with Most Invoices'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Customer ID'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'Total Invoices'}}}
 }))
```

In [ ]:

Advanced SQL questions

In [35]:

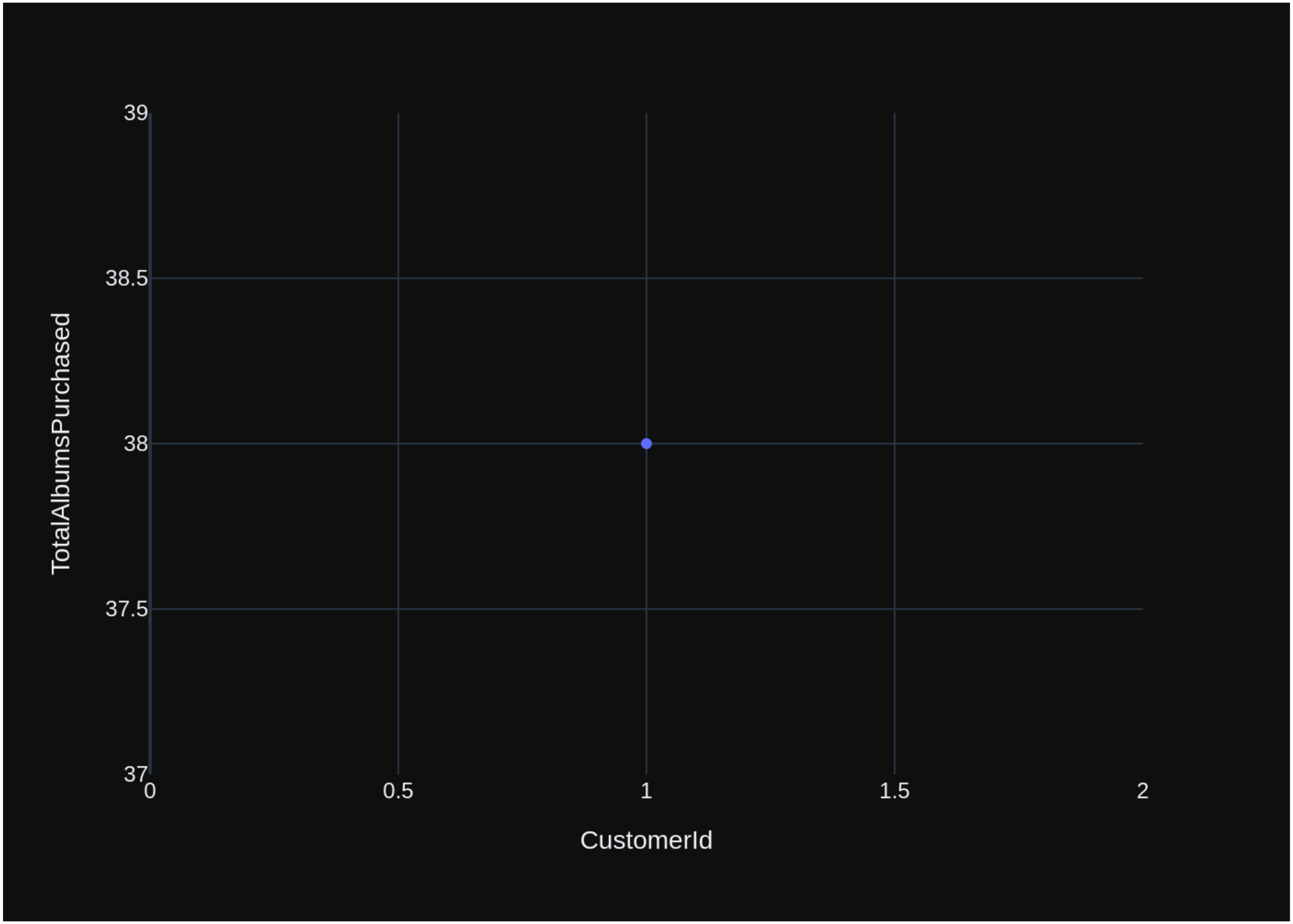
```
question = """
     Find the customer who bought the most albums in total quantity (across all invoices):
"""

vn.ask(question=question)
```

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

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE 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 NU
LL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    By
tes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (Al
bumId) \r\n\t\tON DELETE NO ACTION 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 ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "invoice_item
s"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NUL
L,\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 UPDAT
E NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDA
TE 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 NULL,\r\n    FOREIGN KEY (ArtistId) R
EFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_Alb
umArtistId ON "albums" (ArtistId)\n\nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTO
INCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Bill
ingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingC
ountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FO
REIGN 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_InvoiceLineIn
voiceId ON "invoice_items" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\n
CREATE 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 ch
inook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, pl
ease 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 in
termediate SQL query to find the 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. \n
4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please r
epeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': '  \n     Find the custo
mer with the most invoices \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, COUNT(i.InvoiceId) 
as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Customer
Id\nORDER BY TotalInvoices DESC\nLIMIT 1'}, {'role': 'user', 'content': ' \n    There are 3 tables: artist
s, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks are linked by Albu
mId,\n    Can you find the top 10 most popular artists based on the number of tracks\n'}, {'role': 'assista
nt', 'content': 'SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "albums" a ON t.
AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10'}, {'role': 'user', 'conten
t': '  \n    Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELE
CT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON 
i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user', 'content':
```

```
'  \n    Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content':
'SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Get the
average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, AVG(i.
Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY
c.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\nFROM "invoices" i
\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '
\n    List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM
"invoices"\nWHERE Total > 10.00'}, {'role': 'user', 'content': '  \n    List all albums and their correspon
ding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistNam
e\nFROM "albums" a\nJOIN "artists" ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    L
ist all genres and the number of tracks in each genre:\n'}, {'role': 'assistant', 'content': 'SELECT g.Nam
e, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "genres" g ON t.GenreId = g.GenreId\nGROUP BY g.N
ame'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'role': 'ass
istant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER BY Total
DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n     Find the customer who bought the most albums in tota
l quantity (across all invoices): \n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREM
ENT 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 \"invoic
e_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  N
OT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGE
R  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 ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (Ar
tistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IN
DEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\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 NU
LL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(4
0),\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 IN
DEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"in
voices\" (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 pr
ovided 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 particul
ar column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend t
he 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). \n5. If the question has been aske
d and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "con
tent": "  \n     Find the customer with the most invoices \n"}, {"role": "assistant", "content": "SELECT c.
CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId
= c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1"}, {"role": "user", "content":
" \n    There are 3 tables: artists, 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 popular artists based on the numbe
r of tracks\n"}, {"role": "assistant", "content": "SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM
\"tracks\" t\nJOIN \"albums\" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nL
IMIT 10"}, {"role": "user", "content": "  \n    Get the total number of invoices for each customer\n"}, {"r
ole": "assistant", "content": "SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM
\"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.
LastName"}, {"role": "user", "content": "  \n    Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\nLIMIT 5"},
{"role": "user", "content": "  \n    Get the average invoice total for each customer:\n"}, {"role": "assist
ant", "content": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"invoices\" i\nJOIN \"custome
rs\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "  \n    Find t
he total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.I
nvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP
BY c.Country"}, {"role": "user", "content": "  \n    List all invoices with a total exceeding $10:\n"}, {"r
ole": "assistant", "content": "SELECT *\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "conten
t": "  \n    List all albums and their corresponding artist names  \n"}, {"role": "assistant", "content":
"SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId =
ar.ArtistId"}, {"role": "user", "content": "  \n    List all genres and the number of tracks in each genr
e:\n"}, {"role": "assistant", "content": "SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM \"tracks\" t
\nJOIN \"genres\" g ON t.GenreId = g.GenreId\nGROUP BY g.Name"}, {"role": "user", "content": "what are the
top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUNT(*) as
Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "content": "
\n     Find the customer who bought the most albums in total quantity (across all invoices): \n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:24:56.467861124Z', 'message': {'role': 'assistant',
'content': 'SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "i
```

```
nvoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Cus
tomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 1'}, 'done_reason': 'stop', 'done': True, 'total_duratio
n': 114364687489, 'load_duration': 865731, 'prompt_eval_count': 1382, 'prompt_eval_duration': 100510962000,
'eval_count': 72, 'eval_duration': 13183080000}
SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased
FROM "invoice_items" ii
JOIN "invoices" i ON ii.InvoiceId = i.InvoiceId
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 1
SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased
FROM "invoice_items" ii
JOIN "invoices" i ON ii.InvoiceId = i.InvoiceId
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 1
   CustomerId  TotalAlbumsPurchased
0           1                    38
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 q
uantity (across all invoices): \n'\n\nThe DataFrame was produced using this query: SELECT c.CustomerId, SUM
(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_items\" ii\nJOIN \"invoices\" i ON ii.InvoiceId = i.I
nvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsP
urchased DESC\nLIMIT 1\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning
df.dtypes gives:\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 t
he data is in a pandas dataframe 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 -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:25:19.528315122Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\n\nfig = px.bar(df, x='CustomerId', y='TotalAlbumsPurchased',
title='Customer Who Bought Most Albums')\n\nfig.show()"}, 'done_reason': 'stop', 'done': True, 'total_durat
ion': 23035372562, 'load_duration': 717113, 'prompt_eval_count': 229, 'prompt_eval_duration': 16003884000,
'eval_count': 40, 'eval_duration': 6898859000}
```


39
38.5
38
37.5
37
0
0.5
1
1.5
2
TotalAlbumsPurchased
CustomerId

Out[35]:

```
('SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoices"
i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId
\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 1',
    CustomerId  TotalAlbumsPurchased
 0           1                    38,
 Figure({
     'data': [{'hovertemplate': 'CustomerId=%{x}<br>TotalAlbumsPurchased=%{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([38]),
               'yaxis': 'y'}],
     'layout': {'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'CustomerId'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalAlbumsPurchased'}}}
 }))
```

In [36]:

```
question = """
    Hint: album quantity is found in invoice_items,

    Find the top 5 customers who bought the most albums in total quantity (across all invoices):
"""

vn.ask(question=question)
```

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

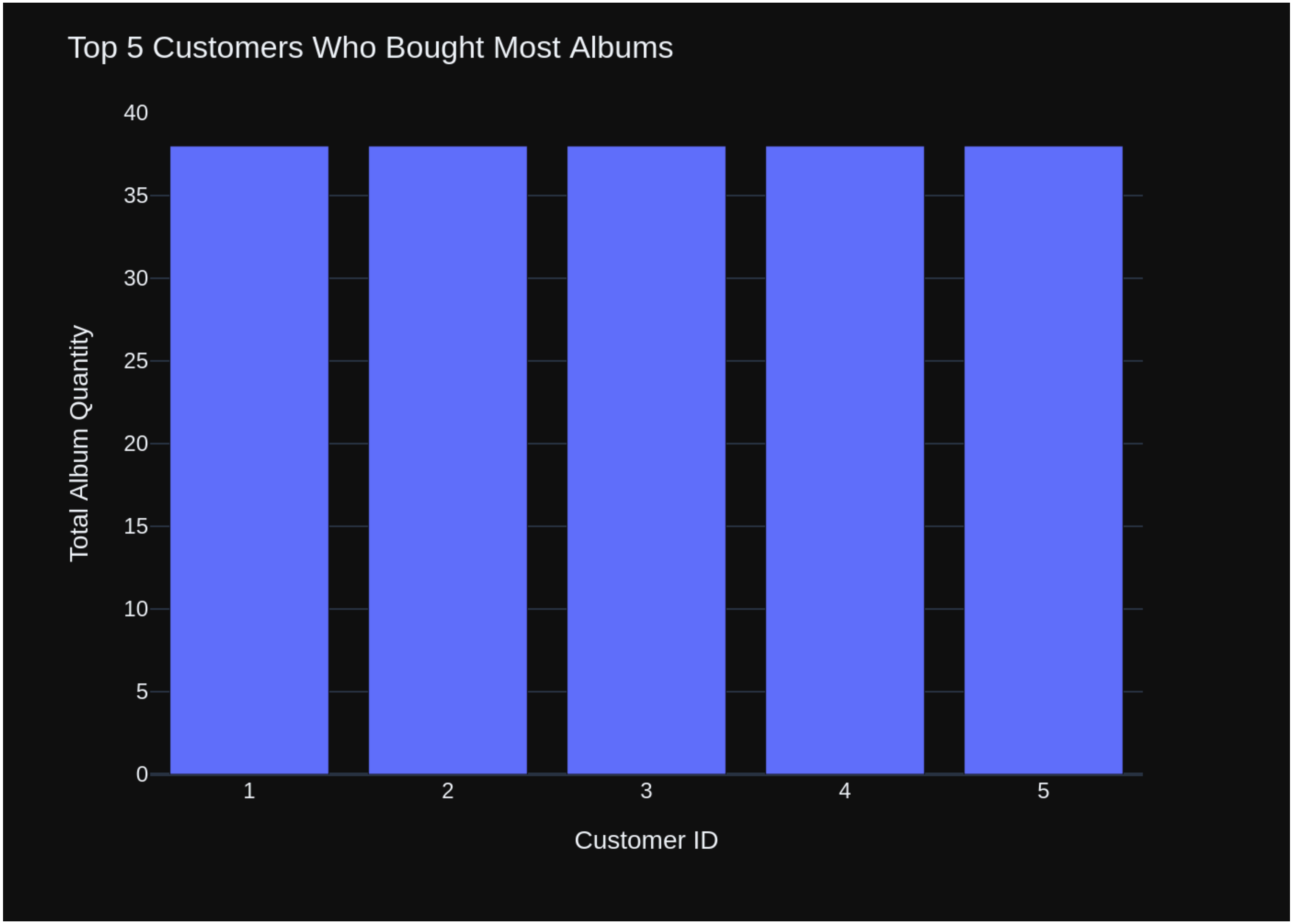
```
[{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the
question. Your response should ONLY be based on the given context and follow the response guidelines and fo
rmat instructions. \n===Tables \nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPr
ice 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) REFERE
NCES "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    Al
bumId INTEGER,\r\n    MediaTypeId 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    UnitPrice NUMERIC(10,2)  NOT NUL
L,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION,\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 NU
LL,\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_Al
bumArtistId 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    InvoiceDa
te DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingS
tate 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 DEL
ETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\n
CREATE 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 ch
inook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, pl
ease 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 in
termediate SQL query to find the 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. \n
4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please r
epeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': '  \n     Find the custo
mer who bought the most albums in total quantity (across all invoices): \n'}, {'role': 'assistant', 'conten
t': 'SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoice
s" i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerI
d\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 1'}, {'role': 'user', 'content': ' \n    There are 3 tables: a
rtists, 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': 'ass
istant', 'content': 'SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "albums" a O
N t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10'}, {'role': 'user', 'cont
ent': '  \n     Find the customer with the most invoices \n'}, {'role': 'assistant', 'content': 'SELECT c.C
ustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.C
```

```
ustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1'}, {'role': 'user', 'content': '  \n
Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT *
\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Get the total num
ber of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, CO
UNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGR
OUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user', 'content': '  \n    Get the average invoic
e total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, AVG(i.Total) as AvgI
nvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.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\nFROM "invoices" i\nJOIN "customers"
c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country'}, {'role': 'user', 'content': '  \n    List all invoi
ces with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "invoices"\nWHERE Tot
al > 10.00'}, {'role': 'user', 'content': 'what are the top 5 countries that customers come from?'}, {'rol
e': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "customers"\nGROUP BY Country\nORDER B
Y Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    List all albums and their corresponding artis
t names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "a
lbums" a\nJOIN "artists" 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 tota
l quantity (across all invoices):\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoice_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY K
EY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    Uni
tPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCE
S \"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 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(22
0),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NUL
L,\r\n    FOREIGN KEY (AlbumId) 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) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (Ar
tistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IN
DEX 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 \"invo
```

```
ices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NUL
L,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(4
0),\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\" (Custo
merId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invo
ices\" (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===Addi
tional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provid
ed 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 co
lumn, 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 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"}, {"rol
e": "assistant", "content": "SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_
items\" ii\nJOIN \"invoices\" i ON ii.InvoiceId = i.InvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.Cus
tomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 1"}, {"role": "user", "content":
" \n    There are 3 tables: artists, 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 popular artists based on the numbe
r of tracks\n"}, {"role": "assistant", "content": "SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM
\"tracks\" t\nJOIN \"albums\" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nL
IMIT 10"}, {"role": "user", "content": "  \n     Find the customer with the most invoices \n"}, {"role": "a
ssistant", "content": "SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN
\"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT
1"}, {"role": "user", "content": "  \n    Find the top 5 most expensive tracks (based on unit price):\n"},
{"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role":
"user", "content": "  \n    Get the total number of invoices for each customer\n"}, {"role": "assistant",
"content": "SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN
\"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role":
"user", "content": "  \n    Get the average invoice total for each customer:\n"}, {"role": "assistant", "co
ntent": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM \"invoices\" i\nJOIN \"customers\" c ON
i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "  \n    Find the total n
umber of invoices per country:\n"}, {"role": "assistant", "content": "SELECT c.Country, COUNT(i.InvoiceId)
as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Coun
try"}, {"role": "user", "content": "  \n    List all invoices with a total exceeding $10:\n"}, {"role": "as
sistant", "content": "SELECT *\nFROM \"invoices\"\nWHERE Total > 10.00"}, {"role": "user", "content": "what
are the top 5 countries that customers come from?"}, {"role": "assistant", "content": "SELECT Country, COUN
T(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, {"role": "user", "cont
ent": "  \n    List all albums and their corresponding artist names  \n"}, {"role": "assistant", "content":
"SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" 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"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:27:18.489003729Z', 'message': {'role': 'assistant',
'content': 'SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "i
nvoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Cus
tomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 5'}, 'done_reason': 'stop', 'done': True, 'total_duratio
n': 118873304903, 'load_duration': 702929, 'prompt_eval_count': 1434, 'prompt_eval_duration': 104955623000,
'eval_count': 72, 'eval_duration': 13247724000}
SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased
FROM "invoice_items" ii
JOIN "invoices" i ON ii.InvoiceId = i.InvoiceId
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 5
SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased
FROM "invoice_items" ii
JOIN "invoices" i ON ii.InvoiceId = i.InvoiceId
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsPurchased DESC
LIMIT 5
   CustomerId  TotalAlbumsPurchased
0           1                    38
1           2                    38
2           3                    38
3           4                    38
4           5                    38
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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    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'\n\nThe Data
Frame was produced using this query: SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM
\"invoice_items\" ii\nJOIN \"invoices\" i ON ii.InvoiceId = i.InvoiceId\nJOIN \"customers\" c ON i.Customer
Id = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 5\n\nThe following is i
nformation about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId
int64\nTotalAlbumsPurchased    int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Py
thon plotly code to chart the results of the dataframe? Assume the data is in a pandas dataframe called 'd
```

```
f'. If there is only one value in the dataframe, use an Indicator. Respond with only Python code. Do not an
swer with any explanations -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:27:47.790301078Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nimport pandas as pd\n\nfig = px.bar(df, x='CustomerId', y='To
talAlbumsPurchased')\n\nfig.update_layout(title='Top 5 Customers Who Bought Most Albums',\n
xaxis_title='Customer ID',\n                  yaxis_title='Total Album Quantity')\n\nfig.show()\n```"}, 'do
ne_reason': 'stop', 'done': True, 'total_duration': 29260057770, 'load_duration': 889862, 'prompt_eval_coun
t': 243, 'prompt_eval_duration': 17063418000, 'eval_count': 69, 'eval_duration': 12065030000}
```


Top 5 Customers Who Bought Most Albums
Total Album Quantity
0
5
10
15
20
25
30
35
40
1
2
3
4
5
Customer ID

```
Out[36]: ('SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoices"
         i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId
         \nORDER BY TotalAlbumsPurchased DESC\nLIMIT 5',
             CustomerId  TotalAlbumsPurchased
          0           1                    38
          1           2                    38
          2           3                    38
          3           4                    38
          4           5                    38,
          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([1, 2, 3, 4, 5]),
                        'xaxis': 'x',
                        'y': array([38, 38, 38, 38, 38]),
                        'yaxis': 'y'}],
              'layout': {'barmode': 'relative',
                         'legend': {'tracegroupgap': 0},
                         'margin': {'t': 60},
                         'template': '...',
                         'title': {'text': 'Top 5 Customers Who Bought Most Albums'},
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'Customer ID'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'Total Album Quantity'}}}
          }))
```

```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 invoice_items detail table is unne
"""

vn.ask(question=question)
```

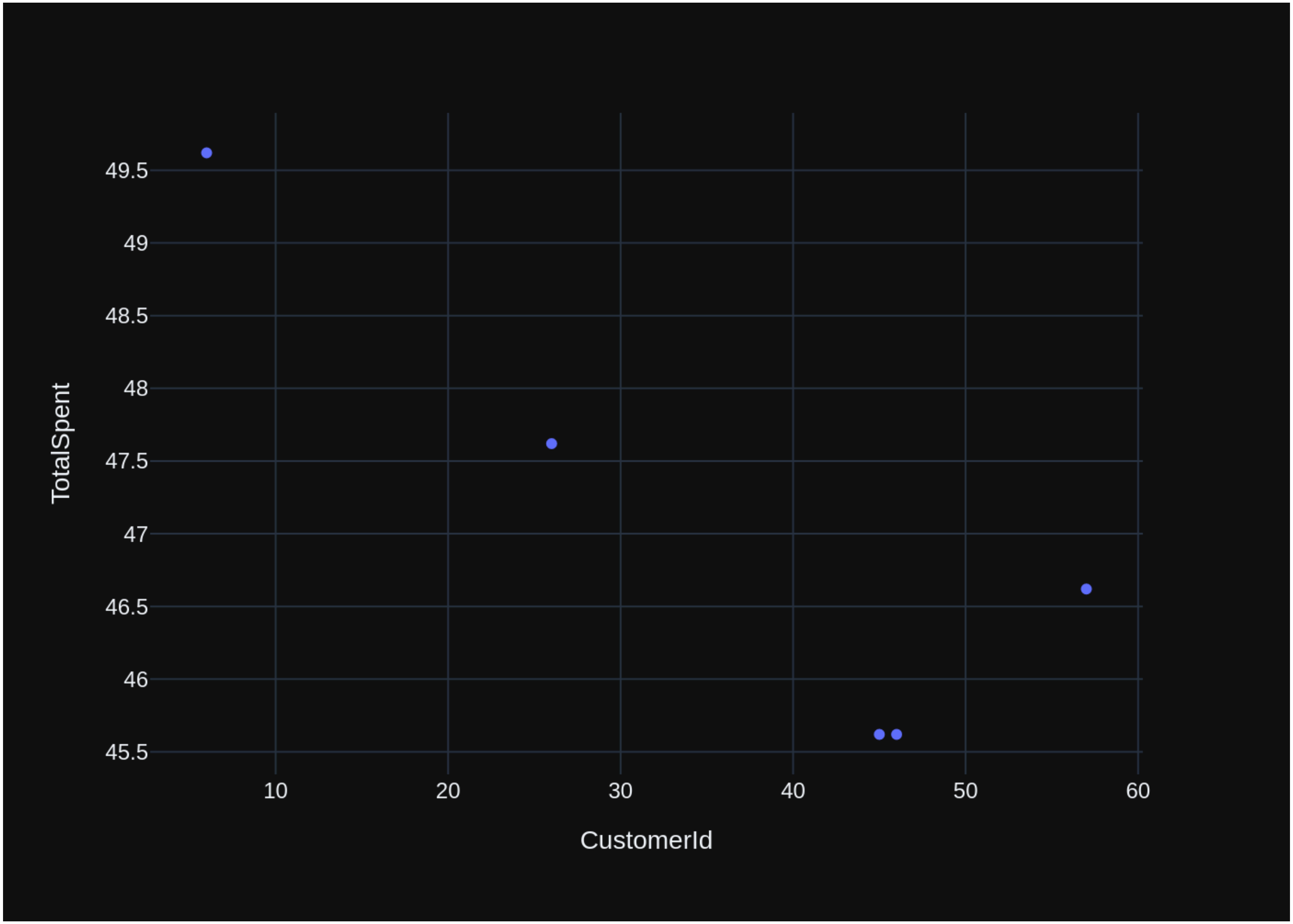
```
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingA
ddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCount
ry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIG
N 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\t
ON 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_InvoiceLin
eTrackId ON "invoice_items" (TrackId)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NUL
L,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHA
R(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax N
VARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportR
epId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TA
BLE "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 INTEGE
R,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(4
0),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NV
ARCHAR(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    Al
bumId INTEGER,\r\n    MediaTypeId 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    UnitPrice NUMERIC(10,2)  NOT NUL
L,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION,\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    Tra
ckId 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 "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 i
s 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    Hint: album qu
antity 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.Quantit
y) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN
"customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLI
MIT 5'}, {'role': 'user', 'content': '  \n     Find the customer who bought the most albums in total quanti
ty (across all invoices): \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(ii.Quantity) as
TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN "custo
mers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 
1'}, {'role': 'user', 'content': '  \n     Find the customer with the most invoices \n'}, {'role': 'assista
nt', 'content': 'SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "custome
rs" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1'}, {'rol
e': 'user', 'content': '  \n    Get the average invoice total for each customer:\n'}, {'role': 'assistant',
'content': 'SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM "invoices" i\nJOIN "customers" c ON
i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': '  \n    Get the total nu
mber of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.FirstName, c.LastName, C
OUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nG
ROUP BY c.CustomerId, c.FirstName, c.LastName'}, {'role': 'user', 'content': 'what are the top 5 countries
that customers come from?'}, {'role': 'assistant', 'content': 'SELECT Country, COUNT(*) as Total\nFROM "cus
tomers"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Find the to
tal number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT c.Country, COUNT(i.Invoic
eId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Cou
ntry'}, {'role': 'user', 'content': '  \n    Find the top 5 most expensive tracks (based on unit pric
e):\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'r
ole': 'user', 'content': '  \n    List all invoices with a total exceeding $10:\n'}, {'role': 'assistant',
'content': 'SELECT *\nFROM "invoices"\nWHERE Total > 10.00'}, {'role': 'user', 'content': ' \n    There are
3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks ar
e 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.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN
"albums" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10'}, {'role':
'user', 'content': '  \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'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    Billin
gAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FORE
IGN 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 NUL
L,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NO
T NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceI
d) \r\n\t\tON DELETE NO ACTION 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 INDEX IFK_InvoiceLineInvoiceId ON \"in
voice_items\" (InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDE
X IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCREATE TABLE \"customers\"\r\n(\r\n    CustomerI
d INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVAR
CHAR(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(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n    FOREI
GN 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(1
0),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (Repo
rtsTo) 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    Comp
oser NVARCHAR(220),\r\n    Milliseconds 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 (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 D
ELETE 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, Tr
ackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\n\n===Addit
ional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provide
d 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 co
lumn, 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 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    Hint: album quantity is found in invoice_items, \n    \n    Find the top 5 customers who bought th
e most albums in total quantity (across all invoices):\n"}, {"role": "assistant", "content": "SELECT c.Cust
omerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_items\" ii\nJOIN \"invoices\" i ON ii.Invo
```

```
iceId = i.InvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY T
otalAlbumsPurchased DESC\nLIMIT 5"}, {"role": "user", "content": "  \n     Find the customer who bought the
most albums in total quantity (across all invoices): \n"}, {"role": "assistant", "content": "SELECT c.Custo
merId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_items\" ii\nJOIN \"invoices\" i ON ii.Invoi
ceId = i.InvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY To
talAlbumsPurchased DESC\nLIMIT 1"}, {"role": "user", "content": "  \n     Find the customer with the most i
nvoices \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nF
ROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY To
talInvoices DESC\nLIMIT 1"}, {"role": "user", "content": "  \n    Get the average invoice total for each cu
stomer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, AVG(i.Total) as AvgInvoiceTotal\nFROM 
\"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId"}, {"role": "use
r", "content": "  \n    Get the total number of invoices for each customer\n"}, {"role": "assistant", "cont
ent": "SELECT c.FirstName, c.LastName, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"cus
tomers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId, c.FirstName, c.LastName"}, {"role": "use
r", "content": "what are the top 5 countries that customers come from?"}, {"role": "assistant", "content": 
"SELECT Country, COUNT(*) as Total\nFROM \"customers\"\nGROUP BY Country\nORDER BY Total DESC\nLIMIT 5"}, 
{"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\nFROM \"invoices\" i\nJOIN \"customer
s\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Country"}, {"role": "user", "content": "  \n    Find the t
op 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"t
racks\"\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": "  \n    List all invoices with a 
total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"invoices\"\nWHERE Total > 10.0
0"}, {"role": "user", "content": " \n    There are 3 tables: artists, albums and tracks, where albums and a
rtists are linked by ArtistId, albums and tracks are linked by AlbumId,\n    Can you find the top 10 most p
opular artists based on the number of tracks\n"}, {"role": "assistant", "content": "SELECT a.ArtistId, COUN
T(t.TrackId) as TotalTracks\nFROM \"tracks\" t\nJOIN \"albums\" a ON t.AlbumId = a.AlbumId\nGROUP BY a.Arti
stId\nORDER BY TotalTracks DESC\nLIMIT 10"}, {"role": "user", "content": "  \n     Find the top 5 customers
who spent the most money overall, \n     \n     Hint: order total can be found on invoices table, calculati
on using invoice_items detail table is unnecessary \n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:30:07.402043563Z', 'message': {'role': 'assistant',
'content': 'SELECT c.CustomerId, SUM(i.Total) as TotalSpent\nFROM "invoices" i\nJOIN "customers" c ON i.Cus
tomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5'}, 'done_reason': 'stop',
'done': True, 'total_duration': 139512996884, 'load_duration': 787476, 'prompt_eval_count': 1773, 'prompt_e
val_duration': 129423335000, 'eval_count': 51, 'eval_duration': 9416710000}
SELECT c.CustomerId, SUM(i.Total) as TotalSpent
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalSpent DESC
LIMIT 5
SELECT c.CustomerId, SUM(i.Total) as TotalSpent
```

```
FROM "invoices" i
JOIN "customers" c ON i.CustomerId = c.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
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 top 5 customers who spent the most money overa
ll, \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 this query: SELECT c.CustomerId, SUM(i.Total)
as TotalSpent\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.Custome
rId\nORDER BY TotalSpent DESC\nLIMIT 5\n\nThe following is information about the resulting pandas DataFrame
'df': \nRunning df.dtypes gives:\n CustomerId      int64\nTotalSpent    float64\ndtype: object"}, {"role":
"user", "content": "Can you generate the Python plotly code to chart the results of the dataframe? Assume t
he data is in a pandas dataframe 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 -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:30:30.384060489Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\n\nfig = px.bar(df, x='CustomerId', y='TotalSpent', title='Top
5 Customers by Total Spent')\n\nfig.show()"}, 'done_reason': 'stop', 'done': True, 'total_duration': 229568
91504, 'load_duration': 672590, 'prompt_eval_count': 224, 'prompt_eval_duration': 15685440000, 'eval_coun
t': 41, 'eval_duration': 7140964000}
```


TotalSpent
49.5
49
48.5
48
47.5
47
46.5
46
45.5
10
20
30
40
50
60
CustomerId

Out[37]:

```
('SELECT c.CustomerId, SUM(i.Total) as TotalSpent\nFROM "invoices" i\nJOIN "customers" c ON i.CustomerId =
c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5',
    CustomerId  TotalSpent
 0           6       49.62
 1          26       47.62
 2          57       46.62
 3          45       45.62
 4          46       45.62,
 Figure({
     'data': [{'hovertemplate': 'CustomerId=%{x}<br>TotalSpent=%{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'symbol': 'circle'},
               'mode': 'markers',
               'name': '',
               'orientation': 'v',
               'showlegend': False,
               'type': 'scatter',
               'x': array([ 6, 26, 57, 45, 46]),
               'xaxis': 'x',
               'y': array([49.62, 47.62, 46.62, 45.62, 45.62]),
               'yaxis': 'y'}],
     'layout': {'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'CustomerId'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalSpent'}}}
 }))
```

In [38]:

```python
question = """
     Get all playlists containing at least 10 tracks and the total duration of those tracks:
"""

vn.ask(question=question)
```

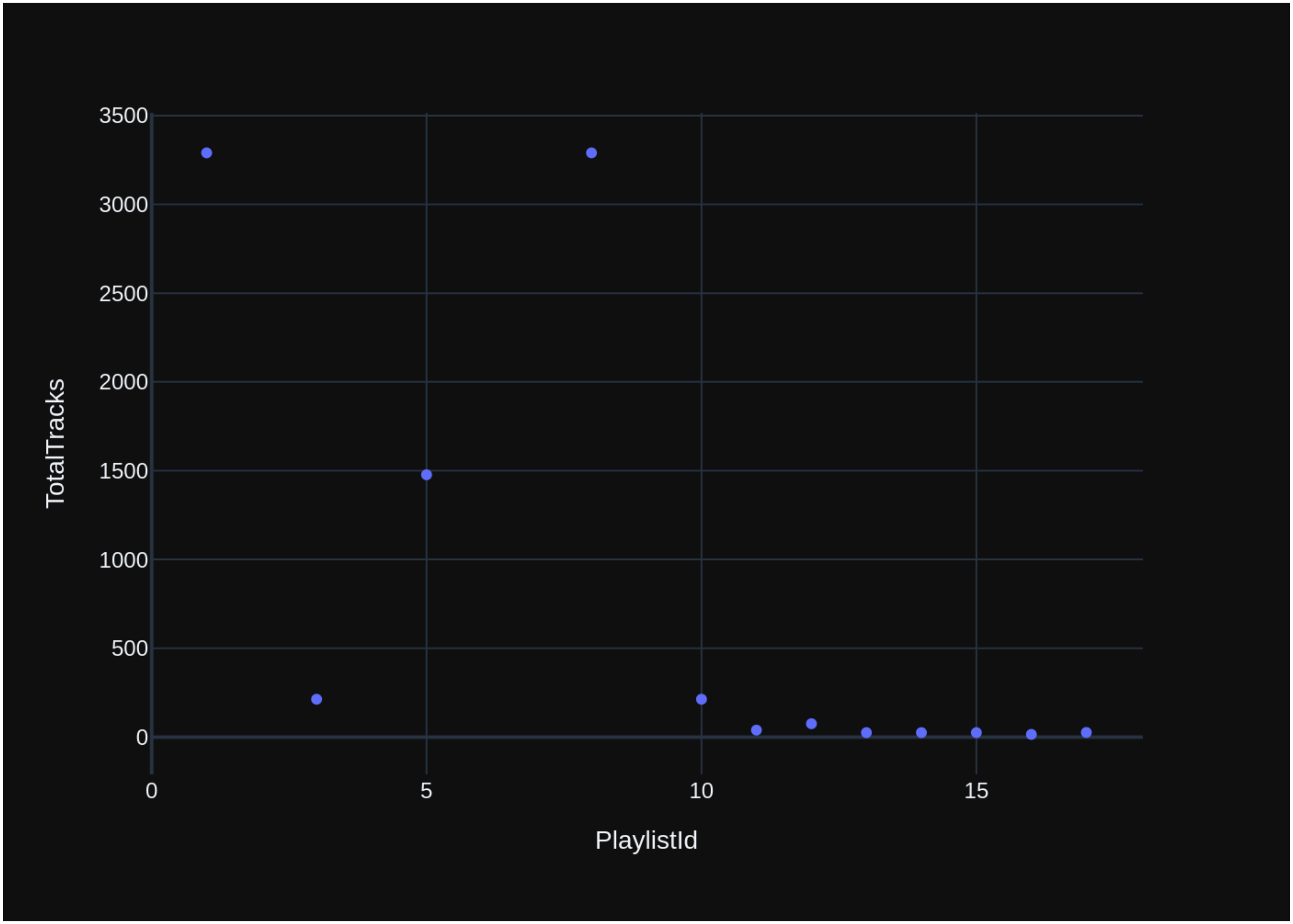
```
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCRE
ATE TABLE "playlists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCH
AR(120)\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTE
GER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (P
laylistId) REFERENCES "playlists" (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\nCRE
ATE 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 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" (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 DEL
ETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE IND
EX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n
\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INTEGE
R PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NU
LL,\r\n    FOREIGN KEY (ArtistId) REFERENCES "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 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===Re
sponse 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 st
rings 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). \n
5. If the question has been asked and answered before, please repeat the answer exactly as it was given bef
ore. \n'}, {'role': 'user', 'content': '  \n    List all genres and the number of tracks in each genr
e:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\n
JOIN "genres" g ON t.GenreId = g.GenreId\nGROUP BY g.Name'}, {'role': 'user', 'content': ' \n    There are
3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks ar
e 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.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN
"albums" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10'}, {'role':
'user', 'content': '  \n    Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role':
'assistant', 'content': 'SELECT *\nFROM "tracks"\nWHERE Name LIKE \'%what%\''}, {'role': 'user', 'content':
'  \n    Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content':
'SELECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Hint: al
bum 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.Qu
antity) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoices" i ON ii.InvoiceId = i.InvoiceId\n
JOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DES
```

```
C\nLIMIT 5'}, {'role': 'user', 'content': '  \n     Find the customer who bought the most albums in total q
uantity (across all invoices): \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(ii.Quantit
y) as TotalAlbumsPurchased\nFROM "invoice_items" ii\nJOIN "invoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN
"customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLI
MIT 1'}, {'role': 'user', 'content': '  \n    List all albums and their corresponding artist names  \n'},
{'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS ArtistName\nFROM "albums" a\nJOIN
"artists" ar ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': 'Can you list all tables in the SQL
ite database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {'role': 'user', 'content': '  \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 un
necessary \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, SUM(i.Total) as TotalSpent\nFROM "in
voices" i\nJOIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalSpent DE
SC\nLIMIT 5'}, {'role': 'user', 'content': '  \n     Find the customer with the most invoices \n'}, {'rol
e': 'assistant', 'content': 'SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJ
OIN "customers" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT
1'}, {'role': 'user', 'content': '  \n     Get all playlists containing at least 10 tracks and the total du
ration of those tracks:\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nC
REATE TABLE \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NV
ARCHAR(120)\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackI
d 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 \"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  NOT NULL,\r\n    GenreId INTEGE
R,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitP
rice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELET
E NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DE
LETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (Gen
reId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tr
acks\" (MediaTypeId)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\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\tO
N DELETE NO ACTION ON UPDATE NO ACTION\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 Context \n\nIn the chinook da
tabase invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please gen
erate a valid SQL query without any explanations for the question. \n2. If the provided context is almost s
ufficient but requires knowledge of a specific string in a particular column, please generate an intermedia
te SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermedi
ate_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 n
umber of tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name, COUNT(t.TrackId) as To
talTracks\nFROM \"tracks\" t\nJOIN \"genres\" g ON t.GenreId = g.GenreId\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 AlbumId,\n    Can you find the top 10 most popular artists bas
ed on the number of tracks\n"}, {"role": "assistant", "content": "SELECT a.ArtistId, COUNT(t.TrackId) as To
talTracks\nFROM \"tracks\" t\nJOIN \"albums\" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistId\nORDER BY Tot
alTracks DESC\nLIMIT 10"}, {"role": "user", "content": "  \n    Find all tracks with a name containing \"Wh
at\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE
'%what%'"}, {"role": "user", "content": "  \n    Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\nLIMIT 5"},
{"role": "user", "content": "  \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 invoices):\n"}, {"role": "assistan
t", "content": "SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_items\" ii\nJ
OIN \"invoices\" i ON ii.InvoiceId = i.InvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROU
P BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 5"}, {"role": "user", "content": "  \n     Fin
d the customer who bought the most albums in total quantity (across all invoices): \n"}, {"role": "assistan
t", "content": "SELECT c.CustomerId, SUM(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_items\" ii\nJ
OIN \"invoices\" i ON ii.InvoiceId = i.InvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROU
P BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 1"}, {"role": "user", "content": "  \n    List
all albums and their corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Title, a.
ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" ar ON a.ArtistId = ar.ArtistId"}, {"ro
le": "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 customers who spent the most money overall, \n     \n     Hint: order total can be found on invoi
ces table, calculation using invoice_items detail table is unnecessary \n"}, {"role": "assistant", "conten
t": "SELECT c.CustomerId, SUM(i.Total) as TotalSpent\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.Custom
erId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5"}, {"role": "user", "conten
t": "  \n     Find the customer with the most invoices \n"}, {"role": "assistant", "content": "SELECT c.Cus
tomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId =
c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1"}, {"role": "user", "content": "
\n     Get all playlists containing at least 10 tracks and the total duration of those tracks:\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:32:28.30901568Z', 'message': {'role': 'assistant',
'content': 'SELECT p.PlaylistId, COUNT(pt.TrackId) as TotalTracks, SUM(t.Milliseconds) / 60000 as TotalDura
```

```
tion\nFROM "playlist_track" pt\nJOIN "playlists" p ON pt.PlaylistId = p.PlaylistId\nJOIN "tracks" t ON pt.T
rackId = t.TrackId\nGROUP BY p.PlaylistId\nHAVING COUNT(pt.TrackId) >= 10'}, 'done_reason': 'stop', 'done':
True, 'total_duration': 117824925580, 'load_duration': 771123, 'prompt_eval_count': 1390, 'prompt_eval_dura
tion': 101400930000, 'eval_count': 84, 'eval_duration': 15757476000}
SELECT p.PlaylistId, COUNT(pt.TrackId) as TotalTracks, SUM(t.Milliseconds) / 60000 as TotalDuration
FROM "playlist_track" pt
JOIN "playlists" p ON pt.PlaylistId = p.PlaylistId
JOIN "tracks" t ON pt.TrackId = t.TrackId
GROUP BY p.PlaylistId
HAVING COUNT(pt.TrackId) >= 10
SELECT p.PlaylistId, COUNT(pt.TrackId) as TotalTracks, SUM(t.Milliseconds) / 60000 as TotalDuration
FROM "playlist_track" pt
JOIN "playlists" p ON pt.PlaylistId = p.PlaylistId
JOIN "tracks" t ON pt.TrackId = t.TrackId
GROUP BY p.PlaylistId
HAVING COUNT(pt.TrackId) >= 10
    PlaylistId  TotalTracks  TotalDuration
0            1         3290          14628
1            3          213           8351
2            5         1477           6645
3            8         3290          14628
4           10          213           8351
5           11           39            158
6           12           75            362
7           13           25            112
8           14           25            126
9           15           25            123
10          16           15             68
11          17           26            136
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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     Get all playlists containing at least 10 tracks and the
total duration of those tracks:\n'\n\nThe DataFrame was produced using this query: SELECT p.PlaylistId, COU
NT(pt.TrackId) as TotalTracks, SUM(t.Milliseconds) / 60000 as TotalDuration\nFROM \"playlist_track\" pt\nJO
IN \"playlists\" p ON pt.PlaylistId = p.PlaylistId\nJOIN \"tracks\" t ON pt.TrackId = t.TrackId\nGROUP BY
p.PlaylistId\nHAVING COUNT(pt.TrackId) >= 10\n\nThe following is information about the resulting pandas Dat
aFrame 'df': \nRunning df.dtypes gives:\n PlaylistId       int64\nTotalTracks      int64\nTotalDuration
int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the r
```

```
esults of the dataframe? Assume the data is in a pandas dataframe called 'df'. If there is only one value i
n the dataframe, use an Indicator. Respond with only Python code. Do not answer with any explanations -- ju
st the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:32:55.125231676Z', 'message': {'role': 'assistant',
'content': '```\nimport plotly.express as px\nfig = px.scatter(df, x=\'TotalTracks\', y=\'TotalDuration\',
hover_name=\'PlaylistId\')\nfig.update_layout(title_text="Playlists by Total Tracks and Duration", title_fo
nt_size=16)\nfig.show()'}, 'done_reason': 'stop', 'done': True, 'total_duration': 26794701378, 'load_durati
on': 681450, 'prompt_eval_count': 243, 'prompt_eval_duration': 16961402000, 'eval_count': 56, 'eval_duratio
n': 9701133000}
```


3500
3000
2500
2000
1500
1000
500
0
TotalTracks
0
5
10
15
PlaylistId

Out[38]:

```
('SELECT p.PlaylistId, COUNT(pt.TrackId) as TotalTracks, SUM(t.Milliseconds) / 60000 as TotalDuration\nFRO
M "playlist_track" pt\nJOIN "playlists" p ON pt.PlaylistId = p.PlaylistId\nJOIN "tracks" t ON pt.TrackId =
t.TrackId\nGROUP BY p.PlaylistId\nHAVING COUNT(pt.TrackId) >= 10',
     PlaylistId  TotalTracks  TotalDuration
 0            1         3290          14628
 1            3          213           8351
 2            5         1477           6645
 3            8         3290          14628
 4           10          213           8351
 5           11           39            158
 6           12           75            362
 7           13           25            112
 8           14           25            126
 9           15           25            123
 10          16           15             68
 11          17           26            136,
 Figure({
     'data': [{'hovertemplate': 'PlaylistId=%{x}<br>TotalTracks=%{y}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'symbol': 'circle'},
               'mode': 'markers',
               'name': '',
               'orientation': 'v',
               'showlegend': False,
               'type': 'scatter',
               'x': array([ 1,  3,  5,  8, 10, 11, 12, 13, 14, 15, 16, 17]),
               'xaxis': 'x',
               'y': array([3290,  213, 1477, 3290,  213,   39,   75,   25,   25,   25,   15,   26]),
               'yaxis': 'y'}],
     'layout': {'legend': {'tracegroupgap': 0},
                'margin': {'t': 60},
                'template': '...',
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'PlaylistId'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'TotalTracks'}}}
 }))
```

In [39]:

```
question = """
     Identify artists who have albums with tracks appearing in multiple genres:


"""
```

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

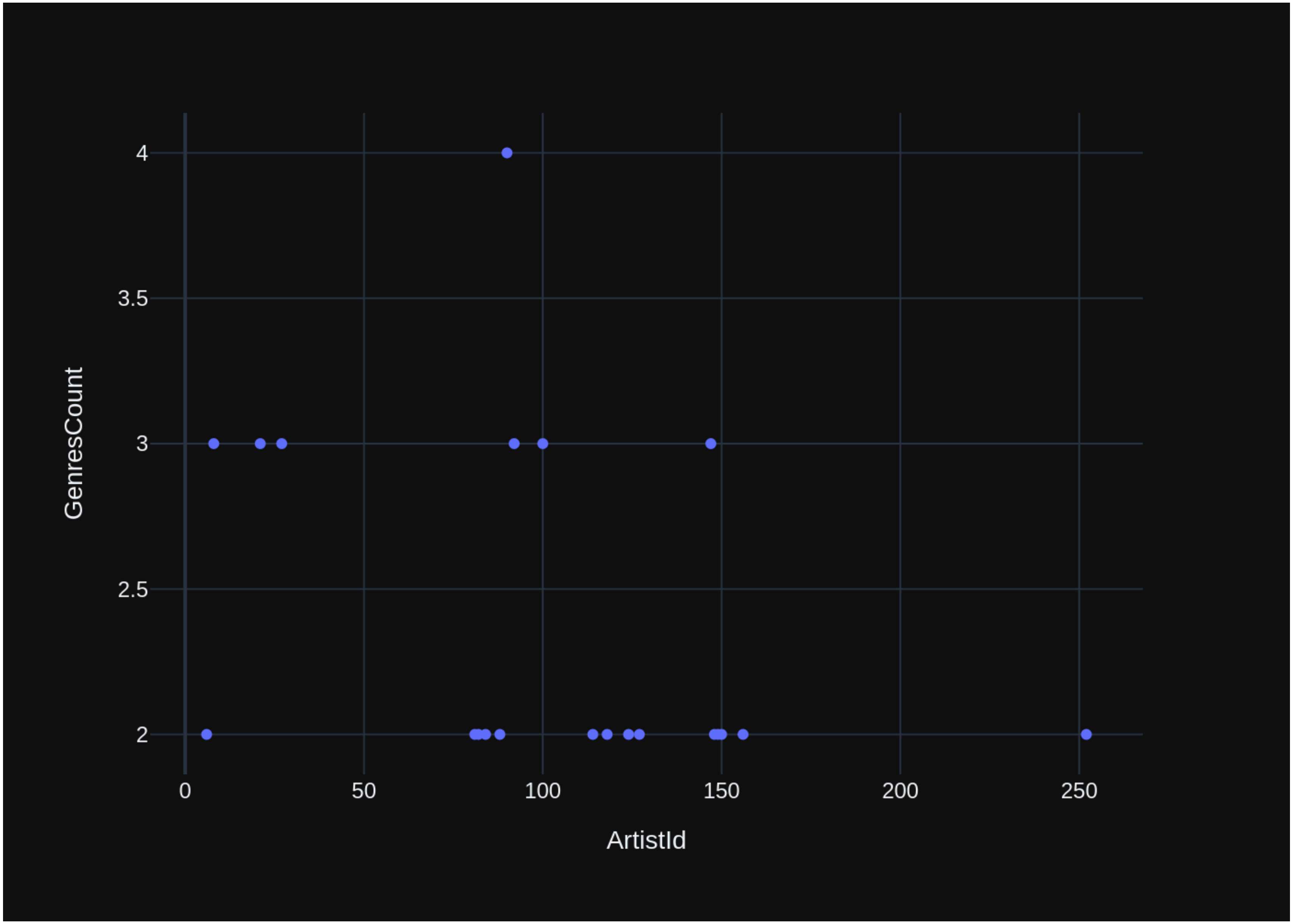
```
Number of requested results 10 is greater than number of elements in index 1, updating n_results = 1
```

```
[{'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 fo
rmat instructions. \n===Tables \nCREATE 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 NU
LL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    By
tes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (Al
bumId) \r\n\t\tON DELETE NO ACTION 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 ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistI
d ON "albums" (ArtistId)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX IFK_TrackAl
bumId ON "tracks" (AlbumId)\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 (Arti
stId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX
IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\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 "pla
ylist_track" (TrackId)\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 "playlist_track"\r\n(\r\n    PlaylistId INTEGER  NOT N
ULL,\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 UPDA
TE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Respons
e Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any expl
anations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a sp
ecific string in a particular column, please generate an intermediate SQL query to find the distinct string
s 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. I
f 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.ArtistId, COU
NT(t.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "albums" a ON t.AlbumId = a.AlbumId\nGROUP BY a.ArtistI
d\nORDER BY TotalTracks DESC\nLIMIT 10'}, {'role': 'user', 'content': '  \n    List all albums and their co
rresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT a.Title, a.ArtistId, ar.Name AS Ar
tistName\nFROM "albums" a\nJOIN "artists" 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.TrackId) as TotalTracks\nFROM "tracks" t\nJOIN "genres" g ON t.GenreId = g.GenreId\nGROUP B
Y g.Name'}, {'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\nFROM "invoice_item
s" ii\nJOIN "invoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.CustomerId\n
GROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n
Get all playlists containing at least 10 tracks and the total duration of those tracks:\n'}, {'role': 'assi
```

```
stant', 'content': 'SELECT p.PlaylistId, COUNT(pt.TrackId) as TotalTracks, SUM(t.Milliseconds) / 60000 as T
otalDuration\nFROM "playlist_track" pt\nJOIN "playlists" p ON pt.PlaylistId = p.PlaylistId\nJOIN "tracks" t
ON pt.TrackId = t.TrackId\nGROUP BY p.PlaylistId\nHAVING COUNT(pt.TrackId) >= 10'}, {'role': 'user', 'conte
nt': '   \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\nFROM "invo
ice_items" ii\nJOIN "invoices" i ON ii.InvoiceId = i.InvoiceId\nJOIN "customers" c ON i.CustomerId = c.Cust
omerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT 1'}, {'role': 'user', 'content': '
\n    Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SE
LECT *\nFROM "tracks"\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': '  \n    Find all tr
acks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': 'SELECT *\nFROM
"tracks"\nWHERE 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     Find the customer with the most invoices \n'}, {'role': 'assistant',
'content': 'SELECT c.CustomerId, COUNT(i.InvoiceId) as TotalInvoices\nFROM "invoices" i\nJOIN "customers" c
ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1'}, {'role': 'us
er', 'content': '  \n     Identify artists who have albums with tracks appearing in multiple genres:\n\n
\n'}]
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
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 fo
rmat instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREM
ENT 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_Albu
mArtistId 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 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 ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE TABLE \"genres\"\r\n(\r\n    Ge
nreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_Playli
stTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE TABLE \"artists\"\r\n(\r\n    ArtistId INTEGER PRI
MARY 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\tO
N 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 invoi
ce means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a vali
d SQL query without any explanations for the question. \n2. If the provided context is almost sufficient bu
t 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 exactl
y 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", "conten
t": "SELECT a.ArtistId, COUNT(t.TrackId) as TotalTracks\nFROM \"tracks\" t\nJOIN \"albums\" a ON t.AlbumId
= a.AlbumId\nGROUP BY a.ArtistId\nORDER BY TotalTracks DESC\nLIMIT 10"}, {"role": "user", "content": "  \n
List all albums and their corresponding artist names  \n"}, {"role": "assistant", "content": "SELECT a.Titl
e, a.ArtistId, ar.Name AS ArtistName\nFROM \"albums\" a\nJOIN \"artists\" 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.TrackId) as TotalTracks\nFROM \"tracks\" t\nJOIN \"genres\"
g ON t.GenreId = g.GenreId\nGROUP BY g.Name"}, {"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 Tota
lAlbumsPurchased\nFROM \"invoice_items\" ii\nJOIN \"invoices\" i ON ii.InvoiceId = i.InvoiceId\nJOIN \"cust
omers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsPurchased DESC\nLIMIT
5"}, {"role": "user", "content": "  \n     Get all playlists containing at least 10 tracks and the total du
ration of those tracks:\n"}, {"role": "assistant", "content": "SELECT p.PlaylistId, COUNT(pt.TrackId) as To
talTracks, SUM(t.Milliseconds) / 60000 as TotalDuration\nFROM \"playlist_track\" pt\nJOIN \"playlists\" p O
N pt.PlaylistId = p.PlaylistId\nJOIN \"tracks\" t ON pt.TrackId = t.TrackId\nGROUP BY p.PlaylistId\nHAVING
COUNT(pt.TrackId) >= 10"}, {"role": "user", "content": "  \n     Find the customer who bought the most albu
ms in total quantity (across all invoices): \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, SU
M(ii.Quantity) as TotalAlbumsPurchased\nFROM \"invoice_items\" ii\nJOIN \"invoices\" i ON ii.InvoiceId = i.
InvoiceId\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalAlbums
Purchased DESC\nLIMIT 1"}, {"role": "user", "content": "  \n    Find the top 5 most expensive tracks (based
on unit price):\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nORDER BY UnitPrice DESC\n
LIMIT 5"}, {"role": "user", "content": "  \n    Find all tracks with a name containing \"What\" (case-insen
sitive)\n"}, {"role": "assistant", "content": "SELECT *\nFROM \"tracks\"\nWHERE Name LIKE '%what%'"}, {"rol
e": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "c
ontent": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": "  \n     Find t
he customer with the most invoices \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, COUNT(i.Inv
oiceId) as TotalInvoices\nFROM \"invoices\" i\nJOIN \"customers\" c ON i.CustomerId = c.CustomerId\nGROUP B
Y c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1"}, {"role": "user", "content": "  \n     Identify arti
sts who have albums with tracks appearing in multiple genres:\n\n\n"}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:34:52.106611935Z', 'message': {'role': 'assistant',
'content': 'SELECT a.ArtistId, COUNT(DISTINCT g.Name) as GenresCount\nFROM "tracks" t\nJOIN "albums" a ON
```

```
t.AlbumId = a.AlbumId\nJOIN "genres" g ON t.GenreId = g.GenreId\nGROUP BY a.ArtistId\nHAVING COUNT(DISTINCT
g.Name) > 1'}, 'done_reason': 'stop', 'done': True, 'total_duration': 116896182010, 'load_duration': 78663
8, 'prompt_eval_count': 1398, 'prompt_eval_duration': 102025399000, 'eval_count': 76, 'eval_duration': 1419
0509000}
SELECT a.ArtistId, COUNT(DISTINCT g.Name) as GenresCount
FROM "tracks" t
JOIN "albums" a ON t.AlbumId = a.AlbumId
JOIN "genres" g ON t.GenreId = g.GenreId
GROUP BY a.ArtistId
HAVING COUNT(DISTINCT g.Name) > 1
SELECT a.ArtistId, COUNT(DISTINCT g.Name) as GenresCount
FROM "tracks" t
JOIN "albums" a ON t.AlbumId = a.AlbumId
JOIN "genres" g ON t.GenreId = g.GenreId
GROUP BY a.ArtistId
HAVING COUNT(DISTINCT g.Name) > 1
    ArtistId  GenresCount
0          6            2
1          8            3
2         21            3
3         27            3
4         81            2
5         82            2
6         84            2
7         88            2
8         90            4
9         92            3
10       100            3
11       114            2
12       118            2
13       124            2
14       127            2
15       147            3
16       148            2
17       149            2
18       150            2
19       156            2
20       252            2
Ollama parameters:
model=llama3:latest,
options={},
keep_alive=None
```

```
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     Identify artists who have albums with tracks appearing
in multiple genres:\n\n\n'\n\nThe DataFrame was produced using this query: SELECT a.ArtistId, COUNT(DISTINC
T g.Name) as GenresCount\nFROM \"tracks\" t\nJOIN \"albums\" a ON t.AlbumId = a.AlbumId\nJOIN \"genres\" g
ON t.GenreId = g.GenreId\nGROUP BY a.ArtistId\nHAVING COUNT(DISTINCT g.Name) > 1\n\nThe following is inform
ation about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n ArtistId       int64\nGenresC
ount    int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to cha
rt 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 explanation
s -- just the code."}]
Ollama Response:
{'model': 'llama3:latest', 'created_at': '2024-06-15T22:35:14.357611341Z', 'message': {'role': 'assistant',
'content': "```\nimport plotly.express as px\nimport matplotlib.pyplot as plt\n\nfig = px.bar(df, x='Artist
Id', y='GenresCount')\nplt.show()\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 2222569388
7, 'load_duration': 612856, 'prompt_eval_count': 224, 'prompt_eval_duration': 15611991000, 'eval_count': 3
7, 'eval_duration': 6483321000}
```


GenresCount
4
3.5
3
2.5
2
0
50
100
150
200
250
ArtistId

```
Out[39]: ('SELECT a.ArtistId, COUNT(DISTINCT g.Name) as GenresCount\nFROM "tracks" t\nJOIN "albums" a ON t.AlbumId
         = a.AlbumId\nJOIN "genres" g ON t.GenreId = g.GenreId\nGROUP BY a.ArtistId\nHAVING COUNT(DISTINCT g.Name)
         > 1',
              ArtistId  GenresCount
          0          6            2
          1          8            3
          2         21            3
          3         27            3
          4         81            2
          5         82            2
          6         84            2
          7         88            2
          8         90            4
          9         92            3
          10       100            3
          11       114            2
          12       118            2
          13       124            2
          14       127            2
          15       147            3
          16       148            2
          17       149            2
          18       150            2
          19       156            2
          20       252            2,
          Figure({
              'data': [{'hovertemplate': 'ArtistId=%{x}<br>GenresCount=%{y}<extra></extra>',
                        'legendgroup': '',
                        'marker': {'color': '#636efa', 'symbol': 'circle'},
                        'mode': 'markers',
                        'name': '',
                        'orientation': 'v',
                        'showlegend': False,
                        'type': 'scatter',
                        'x': array([  6,   8,  21,  27,  81,  82,  84,  88,  90,  92, 100, 114, 118, 124,
                                    127, 147, 148, 149, 150, 156, 252]),
                        'xaxis': 'x',
                        'y': array([2, 3, 3, 3, 2, 2, 2, 2, 4, 3, 3, 2, 2, 2, 2, 3, 2, 2, 2, 2, 2]),
                        'yaxis': 'y'}],
              'layout': {'legend': {'tracegroupgap': 0},
                         'margin': {'t': 60},
                         'template': '...',
                         'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'text': 'ArtistId'}},
                         'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'text': 'GenresCount'}}}
          }))
```

Check completion time

In [ ]:

```
```

In [40]:

```
ts_stop = time()

elapsed_time = ts_stop - ts_start
print(f"test running on '{hostname}' with '{model_name}' LLM took : {elapsed_time:.2f} sec")
```

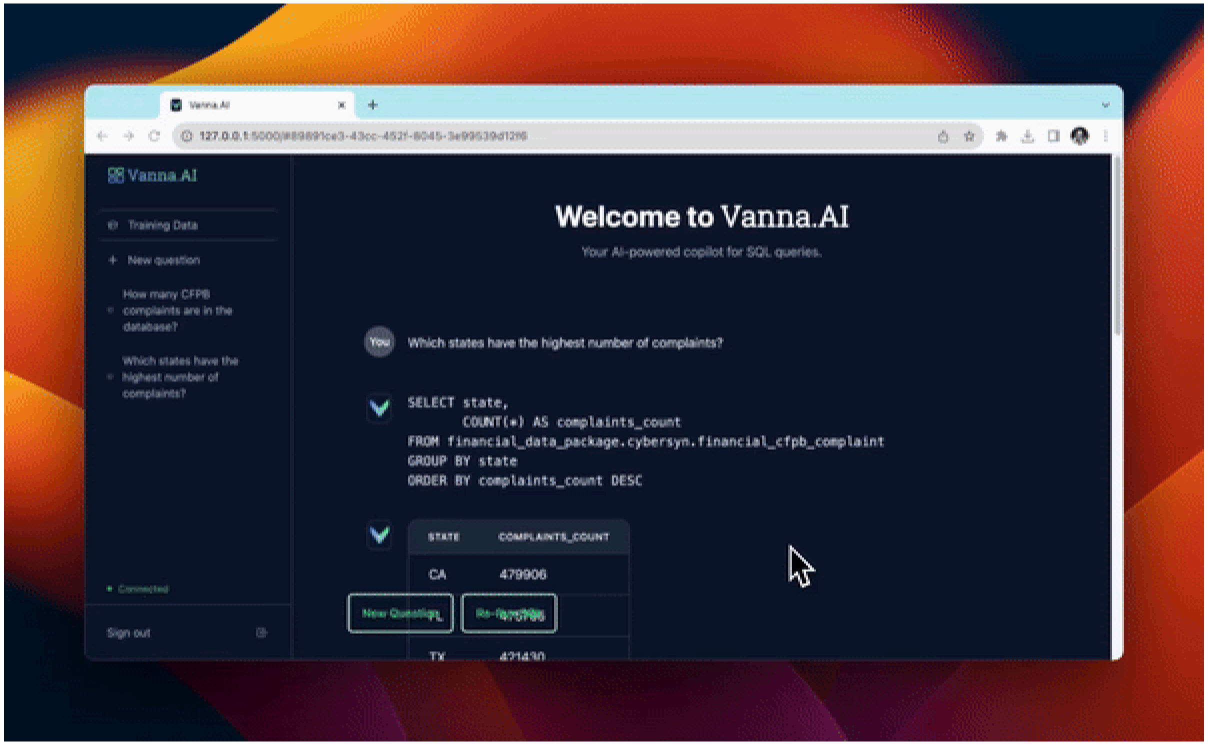
```
test running on 'ducklover1' with 'llama3' LLM took : 2704.21 sec
```

In [41]:

```
from datetime import datetime
print(datetime.now())
```

```
2024-06-15 18:35:14.426379
```

Launch the User Interface



Welcome to Vanna.AI

Your AI-powered copilot for SQL queries.

How many CFPB complaints are in the database?

Which states have the highest number of complaints?

Which states have the highest number of complaints?

```
SELECT state,
       COUNT(*) AS complaints_count
FROM financial_data_package.cybersyn.financial_cfpb_complaint
GROUP BY state
ORDER BY complaints_count DESC
```

STATE	COMPLAINTS_COUNT
CA	479906
FL	[illegible]
TX	421430

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