Generating SQL for SQLite using Ollama, ChromaDB

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

Which LLM do you want to use?

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

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

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

Setup

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

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

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

import os

import re
from time import time

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

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

In [3]:
```
file_db = "~/Downloads/chinook.sqlite"
model_name = 'deepseek-coder-v2'

clean_and_train = True  # False
```

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

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

```
Hostname: ducklover1
```

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

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

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

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

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

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

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

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

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

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

Training

SQLite sample database

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

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

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

Out[12]:

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

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

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

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

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

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

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

Out[14]:

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

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

Asking the AI

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

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

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

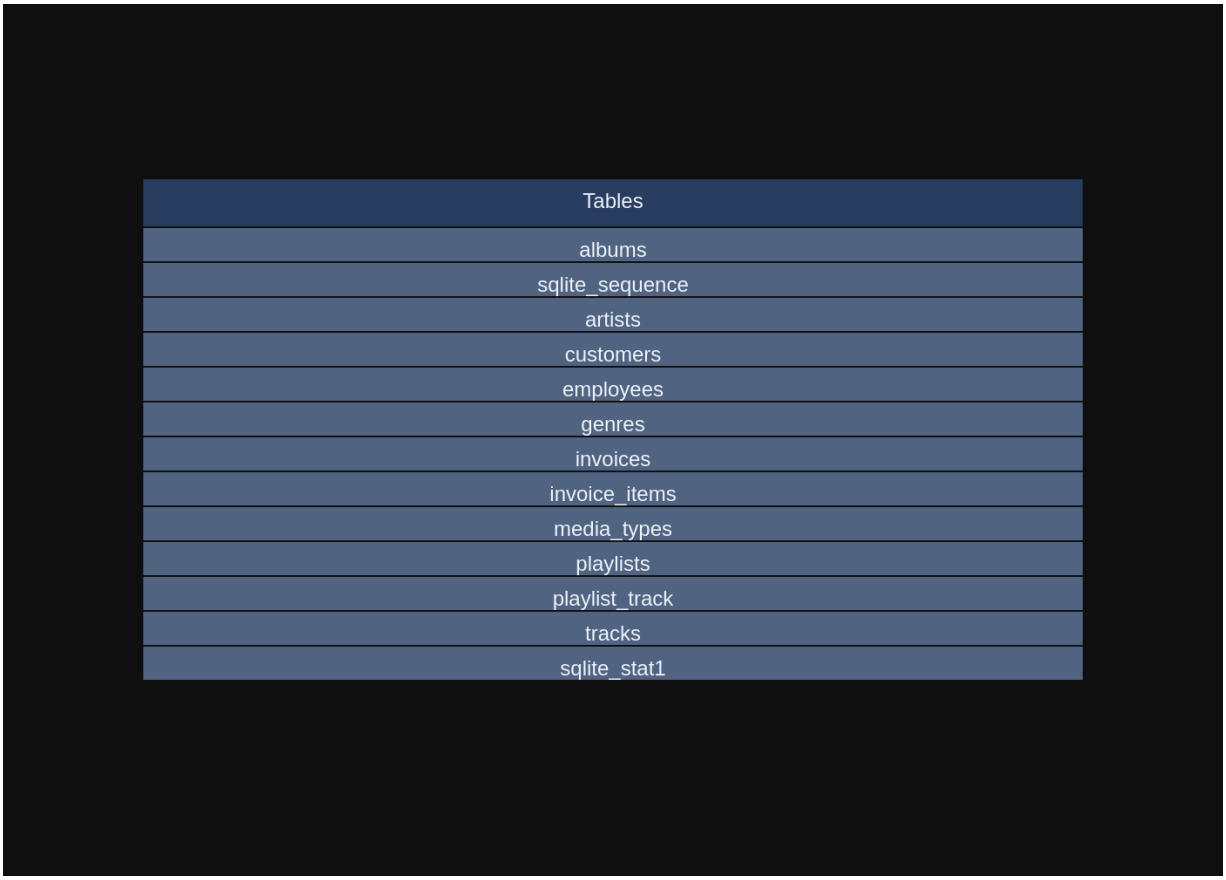
In [16]:
```
vn.ask(question="Can you list all tables in the SQLite database catalog?")
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCR
EATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "playlists"\r\n(\r\n
PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR
(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "trac
ks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    N
ame NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTE
GER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Genr
eId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaType
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "me
dia_types"\r\n(\r\n    MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "artists"\r\n(\r\n    Arti
stId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r
\n)\n\nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMAR
Y KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    Tra
ckId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Qua
ntity INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoice
s" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREI
GN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n    Playlis
tId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT P
K_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (Play
listId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON U
PDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "album
s"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Ti
tle NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREI
GN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook dat
abase invoice means order\n\n===Response Guidelines \n1. If the provided con
text is sufficient, please generate a valid SQL query without any explanatio
ns for the question. \n2. If the provided context is almost sufficient but r
equires knowledge of a specific string in a particular column, please genera
te an intermediate SQL query to find the distinct strings in that column. Pr
epend the query with a comment saying intermediate_sql \n3. If the provided
context is insufficient, please explain why it can\'t be generated. \n4. Ple
ase use the most relevant table(s). \n5. If the question has been asked and
answered before, please repeat the answer exactly as it was given before.
\n'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite dat
abase catalog?'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE TABLE s
```

qlite_sequence(name,seq)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   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 AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId)\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n   MediaTypeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(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\" (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 (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:10:40.460748342Z', 'message': {'role': 'assistant', 'content': " ```sql\nSELECT name FROM sqlite_master WHERE type='table';\n```"}, 'done_reason': 'stop', 'done': True, 'total_duration': 21354378613, 'load_duration': 926684319, 'prompt_eval_count': 934, 'prompt_eval_duration': 19160278000, 'eval_count': 17, 'eval_duration': 1211112000}
LLM Response: ```sql
SELECT name FROM sqlite_master WHERE type='table';
```
Info: Output from LLM:  ```sql
SELECT name FROM sqlite_master WHERE type='table';
```

```
Extracted SQL: SELECT name FROM sqlite_master WHERE type='table'
SELECT name FROM sqlite_master WHERE type='table'
              name
0           albums
1   sqlite_sequence
2          artists
3        customers
4        employees
5           genres
6         invoices
7     invoice_items
8       media_types
9         playlists
10   playlist_track
11           tracks
12     sqlite_stat1
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: 'Ca
n you list all tables in the SQLite database catalog?'\n\nThe DataFrame was
produced using this query: SELECT name FROM sqlite_master WHERE type='tabl
e'\n\nThe following is information about the resulting pandas DataFrame 'd
f': \nRunning df.dtypes gives:\n name    object\ndtype: object"}, {"role":
"user", "content": "Can you generate the Python plotly code to chart the res
ults of the dataframe? Assume the data is in a pandas dataframe called 'df'.
If there is only one value in the dataframe, use an Indicator. Respond with
only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:10:52.224
537865Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\n# Assuming df is your DataF
rame\nif len(df) == 1:\n    fig = go.Figure(go.Indicator(\n        mode="num
ber",\n        value=df[\'name\'].iloc[0],\n        title={"text": "Single T
able in Database"}\n    ))\nelse:\n    fig = go.Figure(data=[go.Table(header
=dict(values=[\'Tables\']), cells=dict(values=[[i for i in df[\'name
\']]))])\n\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_
duration': 11731095855, 'load_duration': 47023938, 'prompt_eval_count': 152,
'prompt_eval_duration': 2909838000, 'eval_count': 132, 'eval_duration': 8722
386000}
```



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': [{'cells': {'values': [['albums', 'sqlite_sequence', 'artist
s',
                                     'customers', 'employees', 'genres', 'in
voices',
                                     'invoice_items', 'media_types', 'playli
sts',
                                     'playlist_track', 'tracks', 'sqlite_sta
t1']]},
                'header': {'values': ['Tables']},
                'type': 'table'}],
     'layout': {'template': '...'}
 }))
```

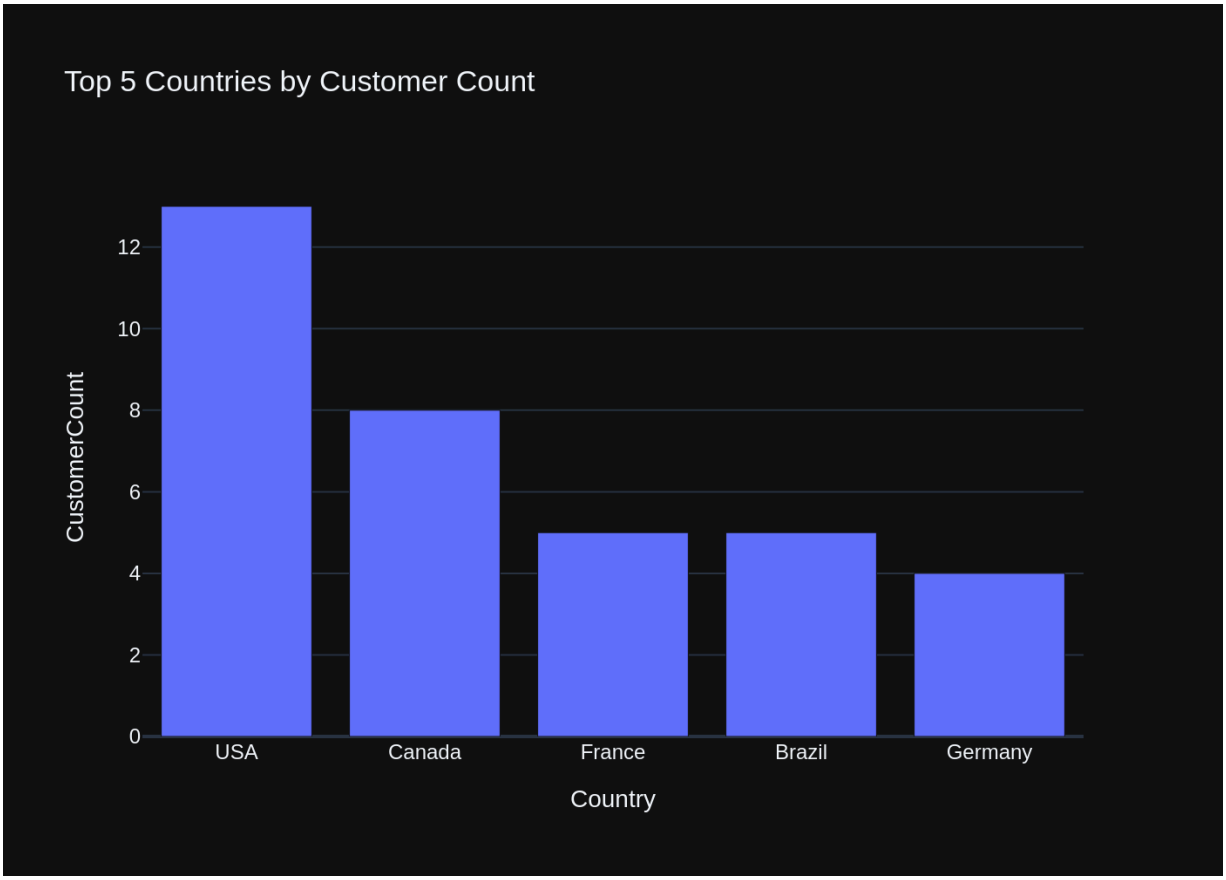
In [17]:
```
vn.ask(question="which table stores customer's orders")
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId I
NTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity
INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (Inv
oiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NUL
L,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n
Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r
\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVA
RCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n
SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employee
s" (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 AUTOINCREMEN
T 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    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r
\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR
(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax
NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFE
RENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "playlist
s"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(120)\r\n)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE
TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NUL
L,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "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 DELE
TE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "t
racks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCRE
ATE TABLE "media_types"\r\n(\r\n    MediaTypeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n
\nIn the chinook database invoice means order\n\n===Response Guidelines \n1.
If the provided context is sufficient, please generate a valid SQL query wit
hout any explanations for the question. \n2. If the provided context is almo
st sufficient but requires knowledge of a specific string in a particular co
lumn, please generate an intermediate SQL query to find the distinct strings
in that column. Prepend the query with a comment saying intermediate_sql \n
3. If the provided context is insufficient, please explain why it can\'t be
generated. \n4. Please use the most relevant table(s). \n5. If the question
has been asked and answered before, please repeat the answer exactly as it w
as given before. \n'}, {'role': 'user', 'content': 'Can you list all tables
```

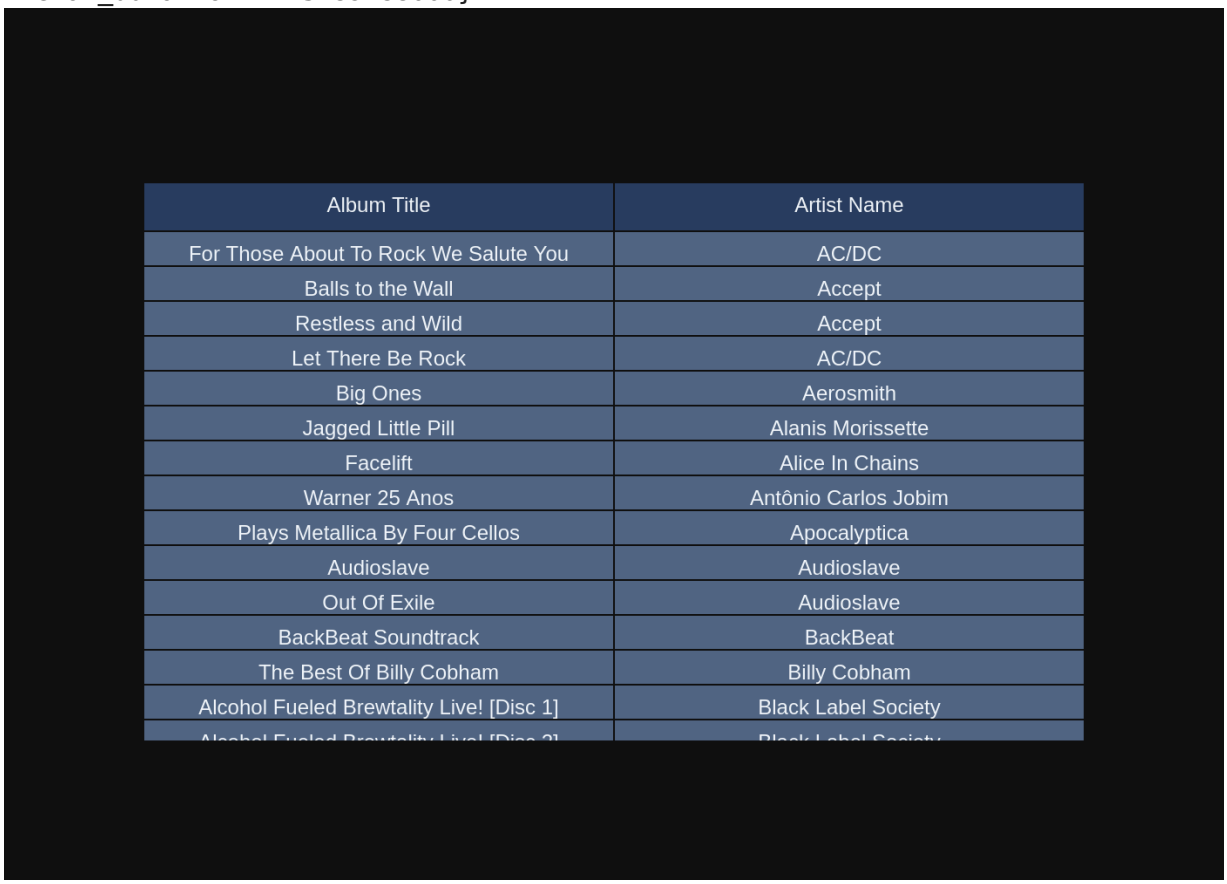
in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT
name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': "w
hich table stores customer's orders"}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRI
MARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r
\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   A
ddress NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r
\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n
SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVA
RCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r
\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR
(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NV
ARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo)
REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE \"p
laylists\"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)
\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGE
R NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playl
ist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, Track
Id),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n   MediaTypeId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===A
dditional Context \n\nIn the chinook database invoice means order\n\n===Resp
onse Guidelines \n1. If the provided context is sufficient, please generate

```
a valid SQL query without any explanations for the question. \n2. If the pro
vided context is almost sufficient but requires knowledge of a specific stri
ng in a particular column, please generate an intermediate SQL query to find
the distinct strings in that column. Prepend the query with a comment saying
intermediate_sql \n3. If the provided context is insufficient, please explai
n why it can't be generated. \n4. Please use the most relevant table(s). \n
5. If the question has been asked and answered before, please repeat the ans
wer exactly as it was given before. \n"}, {"role": "user", "content": "Can y
ou list all tables in the SQLite database catalog?"}, {"role": "assistant",
"content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "u
ser", "content": "which table stores customer's orders"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:11:25.268
169082Z', 'message': {'role': 'assistant', 'content': ' intermediate_sql\n``
`sql\nSELECT DISTINCT InvoiceId FROM invoices;\n```'}, 'done_reason': 'sto
p', 'done': True, 'total_duration': 32677783680, 'load_duration': 864596, 'p
rompt_eval_count': 1192, 'prompt_eval_duration': 31299304000, 'eval_count':
17, 'eval_duration': 1239210000}
LLM Response:  intermediate_sql
```sql
SELECT DISTINCT InvoiceId FROM invoices;
```
The LLM is not allowed to see the data in your database. Your question requi
res database introspection to generate the necessary SQL. Please set allow_l
lm_to_see_data=True to enable this.
Couldn't run sql: Execution failed on sql 'The LLM is not allowed to see th
e data in your database. Your question requires database introspection to ge
nerate the necessary SQL. Please set allow_llm_to_see_data=True to enable th
is.': near "The": syntax error
```

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

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE INDEX IFK_CustomerSupportRepId ON "customers" (SupportRepId)\n\nCRE
ATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREME
NT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCH
AR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(7
0),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVAR
CHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_
InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "invoice_item
s"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPr
ice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREI
GN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks"
(TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IND
EX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE
"albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOR
EIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACT
ION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invo
ice_items" (TrackId)\n\nCREATE TABLE "employees"\r\n(\r\n    EmployeeId INTE
GER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NU
LL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n
ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n
Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r
\n    Country NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVA
RCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIG
N KEY (ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlists"\r\n(\r\n    Playl
istId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)
\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means o
rder\n\n===Response Guidelines \n1. If the provided context is sufficient, p
lease generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can\'t be generated. \n4. Please use the most rele
vant table(s). \n5. If the question has been asked and answered before, plea
se repeat the answer exactly as it was given before. \n'}, {'role': 'user',
'content': 'Can you list all tables in the SQLite database catalog?'}, {'rol
e': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {'role': 'user', 'content': 'How many customers are there'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
```

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

```
WHERE type='table'"}, {"role": "user", "content": "How many customers are th
ere"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:11:50.725
509885Z', 'message': {'role': 'assistant', 'content': ' ```sql\nSELECT COUNT
(*) AS NumberOfCustomers FROM customers;\n```'}, 'done_reason': 'stop', 'don
e': True, 'total_duration': 25409335272, 'load_duration': 1090848, 'prompt_e
val_count': 901, 'prompt_eval_duration': 24081830000, 'eval_count': 16, 'eva
l_duration': 1184569000}
LLM Response:  ```sql
SELECT COUNT(*) AS NumberOfCustomers FROM customers;
```
Info: Output from LLM: ```sql
SELECT COUNT(*) AS NumberOfCustomers FROM customers;
```
Extracted SQL: SELECT COUNT(*) AS NumberOfCustomers FROM customers
SELECT COUNT(*) AS NumberOfCustomers FROM customers
   NumberOfCustomers
0                 59
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: 'Ho
w many customers are there'\n\nThe DataFrame was produced using this query:
SELECT COUNT(*) AS NumberOfCustomers FROM customers\n\nThe following is info
rmation about the resulting pandas DataFrame 'df': \nRunning df.dtypes give
s:\n NumberOfCustomers    int64\ndtype: object"}, {"role": "user", "conten
t": "Can you generate the Python plotly code to chart the results of the dat
aframe? Assume the data is in a pandas dataframe called 'df'. If there is on
ly one value in the dataframe, use an Indicator. Respond with only Python co
de. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:12:05.926
627247Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\n# Assuming df is your DataF
rame\nif len(df) == 1:\n    fig = go.Figure(go.Indicator(\n        mode="num
ber",\n        value=df[\'NumberOfCustomers\'].values[0],\n        title={"t
ext": "Number of Customers"},\n        domain={\'x\': [0, 1], \'y\': [0, 1]}
\n    ))\nelse:\n    fig = go.Figure(go.Bar(\n        x=[\'NumberOfCustomers
\'],\n        y=[df[\'NumberOfCustomers\'].values[0]],\n        marker_color
=\'rgb(55, 200, 132)\'\n    ))\n\nfig.update_layout(title_text="Number of Cu
stomers")\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_du
ration': 15175881939, 'load_duration': 665925, 'prompt_eval_count': 149, 'pr
ompt_eval_duration': 2837657000, 'eval_count': 184, 'eval_duration': 1220361
6000}
```

Out[18]:
```
('SELECT COUNT(*) AS NumberOfCustomers FROM customers',
    NumberOfCustomers
 0                 59,
 Figure({
     'data': [{'domain': {'x': [0, 1], 'y': [0, 1]},
               'mode': 'number',
               'title': {'text': 'Number of Customers'},
               'type': 'indicator',
               'value': 59}],
     'layout': {'template': '...', 'title': {'text': 'Number of Customer
s'}}
 }))
```

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

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOIN
CREMENT 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(40),\r\n    State NVARCHAR(40),\r\n    Country NV
ARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "inv
oice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NUL
L,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NU
LL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFE
RENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE "media_types"\r\n(\r\n    MediaTypeId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(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 NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r
\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIM
E,\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    Posta
lCode 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 TA
BLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NUL
L,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "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 DELE
TE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "t
racks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCRE
ATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE "tracks"\r\n(\r\n    Tra
ckId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)
NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n
GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER
NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r
\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE
NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES "genr
es" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIG
N 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 chi
nook database invoice means order\n\n===Response Guidelines \n1. If the prov
ided context is sufficient, please generate a valid SQL query without any ex
```

planations for the question. \n2. If the provided context is almost sufficie
nt but requires knowledge of a specific string in a particular column, pleas
e generate an intermediate SQL query to find the distinct strings in that co
lumn. Prepend the query with a comment saying intermediate_sql \n3. If the p
rovided context is insufficient, please explain why it can\'t be generated.
\n4. Please use the most relevant table(s). \n5. If the question has been as
ked and answered before, please repeat the answer exactly as it was given be
fore. \n'}, {'role': 'user', 'content': 'How many customers are there'}, {'r
ole': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers FROM cus
tomers'}, {'role': 'user', 'content': 'Can you list all tables in the SQLite
database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqli
te_master WHERE type='table'"}, {'role': 'user', 'content': 'what are the to
p 5 countries that customers come from?'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR
(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r
\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR
(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax
NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEG
ER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (EmployeeId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoi
ce_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r
\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NUL
L,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFE
RENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n   MediaTypeId INTEGER PRIMAR
Y KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX
IFK_CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"e
mployees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NO
T NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDat
e DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   Ci
ty NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2
4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e
mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGE
R NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId)

```
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playl
ist_track\"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER
NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, Track
Id),\r\n    FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE \"trac
ks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INT
EGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES \"albums\" (Albu
mId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Ge
nreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (Medi
aTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additi
onal Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can't be generated. \n4. Please use the most relevant table(s). \n5. If
the question has been asked and answered before, please repeat the answer ex
actly as it was given before. \n"}, {"role": "user", "content": "How many cu
stomers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) AS Nu
mberOfCustomers FROM customers"}, {"role": "user", "content": "Can you list
all tables in the SQLite database catalog?"}, {"role": "assistant", "conten
t": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user",
"content": "what are the top 5 countries that customers come from?"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:12:47.444
594991Z', 'message': {'role': 'assistant', 'content': ' ```sql\nSELECT Billi
ngCountry, COUNT(*) AS CustomerCount\nFROM customers\nGROUP BY BillingCountr
y\nORDER BY CustomerCount DESC\nLIMIT 5;\n```'}, 'done_reason': 'stop', 'don
e': True, 'total_duration': 41422469283, 'load_duration': 671375, 'prompt_ev
al_count': 1388, 'prompt_eval_duration': 38565320000, 'eval_count': 34, 'eva
l_duration': 2629828000}
LLM Response:  ```sql
SELECT BillingCountry, COUNT(*) AS CustomerCount
FROM customers
GROUP BY BillingCountry
ORDER BY CustomerCount DESC
LIMIT 5;
```
Info: Output from LLM: ```sql
SELECT BillingCountry, COUNT(*) AS CustomerCount
FROM customers
GROUP BY BillingCountry
ORDER BY CustomerCount DESC
LIMIT 5;
```
Extracted SQL: SELECT BillingCountry, COUNT(*) AS CustomerCount
FROM customers
GROUP BY BillingCountry
```

```
ORDER BY CustomerCount DESC
LIMIT 5
SELECT BillingCountry, COUNT(*) AS CustomerCount
FROM customers
GROUP BY BillingCountry
ORDER BY CustomerCount DESC
LIMIT 5
Couldn't run sql:  Execution failed on sql 'SELECT BillingCountry, COUNT(*)
AS CustomerCount
FROM customers
GROUP BY BillingCountry
ORDER BY CustomerCount DESC
LIMIT 5': no such column: BillingCountry
```

In [43]:
```
vn.ask(question="""
Hint: customers table already has a column called "country".
What are the top 5 countries that customers come from?

""")
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "customers"\r\n(\r\n    Customer
Id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)
NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(8
0),\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    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT
NULL,\r\n    SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFEREN
CES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY A
UTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDa
te DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCit
y NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCountry NVAR
CHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)
NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "emp
loyees"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NU
LL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DA
TETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City N
VARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(2
4),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENCES "em
ployees" (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 I
NTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity
INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (Inv
oiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMAR
Y KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    A
rtistId 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 "media_types"\r\n(\r\n    MediaTypeId 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 KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(12
0)\r\n)\n\nCREATE TABLE "artists"\r\n(\r\n    ArtistId INTEGER PRIMARY KEY A
UTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "playl
ists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "custome
rs" (SupportRepId)\n\n\n===Additional Context \n\nIn the chinook database in
voice means order\n\n===Response Guidelines \n1. If the provided context is
sufficient, please generate a valid SQL query without any explanations for t
he 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 th
e 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'}, {'rol
e': 'user', 'content': '  \n     Find the top 5 customers who spent the most
money overall, \n     \n     Hint: order total can be found on invoices tabl
e, calculation using invoice_items detail table is unnecessary \n'}, {'rol
```

e': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, S
UM(i.Total) AS TotalSpent\nFROM customers c\nJOIN invoices i ON c.CustomerId
= i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5'},
{'role': 'user', 'content': ' \n   List all customers from Canada and the
ir email addresses:\n'}, {'role': 'assistant', 'content': "SELECT FirstName,
LastName, Email\nFROM customers\nWHERE Country = 'Canada'"}, {'role': 'use
r', '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.Custo
merId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought
\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN inv
oice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY
TotalAlbumQuantityBought DESC\nLIMIT 5'}, {'role': 'user', 'content': ' \n
Find the customer with the most invoices \n'}, {'role': 'assistant', 'conten
t': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS Tot
alInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.Custo
merId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1'}, {'rol
e': 'user', 'content': ' \n   Find the total number of invoices per countr
y:\n'}, {'role': 'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS
TotalInvoices\nFROM invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'c
ontent': ' \n   Find the customer who bought the most albums in total qua
ntity (across all invoices): \n'}, {'role': 'assistant', 'content': 'SELECT
c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought
\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN inv
oice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY
TotalAlbumsBought DESC\nLIMIT 1'}, {'role': 'user', 'content': ' \n   Get
the total number of invoices for each customer\n'}, {'role': 'assistant', 'c
ontent': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) A
S TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.
CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   The
re are 3 tables: artists, albums and tracks, where albums and artists are li
nked by ArtistId, albums and tracks are linked by AlbumId,\n   Can you find
the top 10 most popular artists based on the number of tracks\n'}, {'role':
'assistant', 'content': 'SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS N
umberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId = al.Arti
stId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nOR
DER BY NumberOfTracks DESC\nLIMIT 10'}, {'role': 'user', 'content': ' \n
Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'a
ssistant', 'content': 'SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPri
ce DESC\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are ther
e'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers
FROM customers'}, {'role': 'user', 'content': '\nHint: customers table alrea
dy has a column called "country".\nWhat are the top 5 countries that custome
rs come from?\n\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n
Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r

\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n
SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETI
ME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHA
R(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(4
0),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT N
ULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"employ
yees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NU
LL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DA
TETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City N
VARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2
4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e
mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   Track
Id INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quant
ity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices
\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREI
GN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION
ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTE
GER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NUL
L,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENC
ES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE \"media_types\"\r\n(\r\n   MediaTypeId 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 \"artists\"\r\n(\r\n   Artist
Id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r
\n)\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_C
ustomerSupportRepId ON \"customers\" (SupportRepId)\n\n\n===Additional Conte
xt \n\nIn the chinook database invoice means order\n\n===Response Guidelines
\n1. If the provided context is sufficient, please generate a valid SQL quer
y without any explanations for the question. \n2. If the provided context is
almost sufficient but requires knowledge of a specific string in a particula
r column, please generate an intermediate SQL query to find the distinct str
ings in that column. Prepend the query with a comment saying intermediate_sq
l \n3. If the provided context is insufficient, please explain why it can't
be generated. \n4. Please use the most relevant table(s). \n5. If the questi
on has been asked and answered before, please repeat the answer exactly as i
t was given before. \n"}, {"role": "user", "content": " \n   Find the top
5 customers who spent the most money overall, \n   \n   Hint: order tota
l can be found on invoices table, calculation using invoice_items detail tab
le is unnecessary \n"}, {"role": "assistant", "content": "SELECT c.CustomerI
d, c.FirstName, c.LastName, SUM(i.Total) AS TotalSpent\nFROM customers c\nJO
IN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\nORDER B
Y TotalSpent DESC\nLIMIT 5"}, {"role": "user", "content": " \n   List all
customers from Canada and their email addresses:\n"}, {"role": "assistant",
"content": "SELECT FirstName, LastName, Email\nFROM customers\nWHERE Country
= 'Canada'"}, {"role": "user", "content": " \n   Hint: album quantity is f

ound in invoice_items, \n   \n   Find the top 5 customers who bought the m
ost albums in total quantity (across all invoices):\n"}, {"role": "assistan
t", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantit
y) AS TotalAlbumQuantityBought\nFROM customers c\nJOIN invoices i ON c.Custo
merId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nG
ROUP BY c.CustomerId\nORDER BY TotalAlbumQuantityBought DESC\nLIMIT 5"}, {"r
ole": "user", "content": " \n   Find the customer with the most invoices
\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.
LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN i
nvoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY To
talInvoices DESC\nLIMIT 1"}, {"role": "user", "content": " \n   Find the t
otal number of invoices per country:\n"}, {"role": "assistant", "content":
"SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY B
illingCountry"}, {"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, c.FirstName, c.LastName, S
UM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c\nJOIN invoices i ON
c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.Invoi
ceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMIT 1"}, {"r
ole": "user", "content": " \n   Get the total number of invoices for each
customer\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.First
Name, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEF
T JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"},
{"role": "user", "content": " \n   There are 3 tables: artists, albums and
tracks, where albums and artists are linked by ArtistId, albums and tracks a
re linked by AlbumId,\n   Can you find the top 10 most popular artists base
d on the number of tracks\n"}, {"role": "assistant", "content": "SELECT ar.N
ame AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT
JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumI
d = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 1
0"}, {"role": "user", "content": " \n   Find the top 5 most expensive trac
ks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Nam
e, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "use
r", "content": "How many customers are there"}, {"role": "assistant", "conte
nt": "SELECT COUNT(*) AS NumberOfCustomers FROM customers"}, {"role": "use
r", "content": "\nHint: customers table already has a column called \"countr
y\".\nWhat are the top 5 countries that customers come from?\n\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:51:15.636
713239Z', 'message': {'role': 'assistant', 'content': ' SELECT Country, COUN
T(*) AS CustomerCount\nFROM customers\nGROUP BY Country\nORDER BY CustomerCo
unt DESC\nLIMIT 5'}, 'done_reason': 'stop', 'done': True, 'total_duration':
54794674001, 'load_duration': 639414, 'prompt_eval_count': 1894, 'prompt_eva
l_duration': 51943370000, 'eval_count': 26, 'eval_duration': 2163721000}
LLM Response: SELECT Country, COUNT(*) AS CustomerCount
FROM customers
GROUP BY Country
ORDER BY CustomerCount DESC
LIMIT 5
 SELECT Country, COUNT(*) AS CustomerCount
FROM customers
GROUP BY Country
ORDER BY CustomerCount DESC
LIMIT 5
   Country CustomerCount
0   USA   13

```
1   Canada              8
2   France              5
3   Brazil              5
4  Germany              4
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '\n
Hint: customers table already has a column called \"country\".\nWhat are the
top 5 countries that customers come from?\n\n'\n\nThe DataFrame was produced
using this query:  SELECT Country, COUNT(*) AS CustomerCount\nFROM customers
\nGROUP BY Country\nORDER BY CustomerCount DESC\nLIMIT 5\n\nThe following is
information about the resulting pandas DataFrame 'df': \nRunning df.dtypes g
ives:\n Country          object\nCustomerCount     int64\ndtype: object"},
{"role": "user", "content": "Can you generate the Python plotly code to char
t the results of the dataframe? Assume the data is in a pandas dataframe cal
led 'df'. If there is only one value in the dataframe, use an Indicator. Res
pond with only Python code. Do not answer with any explanations -- just the
code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:51:26.604
760317Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.express as px\n\nif len(df) == 1:\n    fig = px.indicator(title="Top Co
untry by Customer Count", value=df[\'CustomerCount\'].iloc[0], dimension=
{\'reference\': df[\'CustomerCount\'].max()})\nelse:\n    fig = px.bar(df, x
=\'Country\', y=\'CustomerCount\', title=\'Top 5 Countries by Customer Count
\')\n\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_durati
on': 10946380979, 'load_duration': 646301, 'prompt_eval_count': 192, 'prompt
_eval_duration': 3913821000, 'eval_count': 102, 'eval_duration': 6897336000}
```


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

Out[43]:
```
(' SELECT Country, COUNT(*) AS CustomerCount\nFROM customers\nGROUP BY Coun
try\nORDER BY CustomerCount DESC\nLIMIT 5',
    Country  CustomerCount
 0      USA             13
 1   Canada              8
 2   France              5
 3   Brazil              5
 4  Germany              4,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'Country=%{x}<br>CustomerCount=%{y}<extra>
</extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['USA', 'Canada', 'France', 'Brazil', 'Germany'],
dtype=object),
               'xaxis': 'x',
               'y': array([13,  8,  5,  5,  4]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Top 5 Countries by Customer Count'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Country'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'CustomerCount'}}}
 }))
```

More SQL questions

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

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

vn.ask(question=question)
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_AlbumArtistId ON "albums" (A
rtistId)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AU
TOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId
INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (Artis
tId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "t
racks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INT
EGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Genr
eId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaType
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK
_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "artists"\r\n(\r\n    Ar
tistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)
\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX
IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE TABLE "play
lists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n
\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\n\n===Additi
onal Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can\'t be generated. \n4. Please use the most relevant table(s). \n5. I
f the question has been asked and answered before, please repeat the answer
exactly as it was given before. \n'}, {'role': 'user', 'content': 'Can you l
ist all tables in the SQLite database catalog?'}, {'role': 'assistant', 'con
tent': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'use
r', 'content': 'How many customers are there'}, {'role': 'assistant', 'conte
nt': 'SELECT COUNT(*) AS NumberOfCustomers FROM customers'}, {'role': 'use
r', 'content': '  \n    List all albums and their corresponding artist names
\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n
\nCREATE TABLE \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER
NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks
\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Na
me NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEG
```

```
ER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES \"albums\" (Albu
mId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Ge
nreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (Medi
aTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDE
X IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n
(\r\n    ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NV
ARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n
\nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nC
REATE TABLE \"playlists\"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINC
REMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"genres\"\r
\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name N
VARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaT
ypeId)\n\n\n===Additional Context \n\nIn the chinook database invoice means
order\n\n===Response Guidelines \n1. If the provided context is sufficient,
please generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can't be generated. \n4. Please use the most relev
ant table(s). \n5. If the question has been asked and answered before, pleas
e repeat the answer exactly as it was given before. \n"}, {"role": "user",
"content": "Can you list all tables in the SQLite database catalog?"}, {"rol
e": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {"role": "user", "content": "How many customers are there"}, {"role":
"assistant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customer
s"}, {"role": "user", "content": "  \n    List all albums and their correspo
nding artist names  \n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:13:08.312
936655Z', 'message': {'role': 'assistant', 'content': ' ```sql\nSELECT a.Tit
le AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON
a.ArtistId = ar.ArtistId;\n```'}, 'done_reason': 'stop', 'done': True, 'tota
l_duration': 20826275373, 'load_duration': 658593, 'prompt_eval_count': 715,
'prompt_eval_duration': 17852234000, 'eval_count': 39, 'eval_duration': 2786
774000}
LLM Response:  ```sql
SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName
FROM albums a
JOIN artists ar ON a.ArtistId = ar.ArtistId;
```
Info: Output from LLM: ```sql
SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName
FROM albums a
JOIN artists ar ON a.ArtistId = ar.ArtistId;
```
Extracted SQL: SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName
FROM albums a
JOIN artists ar ON a.ArtistId = ar.ArtistId
SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName
FROM albums a
JOIN artists ar ON a.ArtistId = ar.ArtistId
                                          AlbumTitle  \
```

```
0                For Those About To Rock We Salute You
1                                    Balls to the Wall
2                                    Restless and Wild
3                                    Let There Be Rock
4                                             Big Ones
..                                                 ...
342                            Respighi:Pines of Rome
343  Schubert: The Late String Quartets & String Qu...
344                               Monteverdi: L'Orfeo
345                             Mozart: Chamber Music
346  Koyaanisqatsi (Soundtrack from the Motion Pict...

                                          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 2 columns]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all albums and their corresponding artist names  \n'\n\nThe DataF
rame was produced using this query: SELECT a.Title AS AlbumTitle, ar.Name AS
ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId\n\nTh
e following is information about the resulting pandas DataFrame 'df': \nRunn
ing df.dtypes gives:\n AlbumTitle    object\nArtistName    object\ndtype: ob
ject"}, {"role": "user", "content": "Can you generate the Python plotly code
to chart the results of the dataframe? Assume the data is in a pandas datafr
ame called 'df'. If there is only one value in the dataframe, use an Indicat
or. Respond with only Python code. Do not answer with any explanations -- ju
st the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:13:25.195
766878Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\n# Assuming df is already de
fined and contains the columns \'AlbumTitle\' and \'ArtistName\'\nif len(df)
== 1:\n    fig = go.Figure(go.Indicator(\n        mode="number",\n        va
lue=1,\n        title={"text": f"{df[\'AlbumTitle\'].iloc[0]} by {df[\'Artis
tName\'].iloc[0]}"}\n    ))\nelse:\n    fig = go.Figure(data=[go.Table(heade
r=dict(values=[\'Album Title\', \'Artist Name\']),\n
cells=dict(values=[[row[\'AlbumTitle\'] for idx, row in df.iterrows()],\n
[row[\'ArtistName\'] for idx, row in df.iterrows()]])\n
)\n                        ])\n\nfig.show()\n```'}, 'done_reason': 'stop',
'done': True, 'total_duration': 16856521973, 'load_duration': 41195033, 'pro
```

mpt_eval_count': 182, 'prompt_eval_duration': 3581099000, 'eval_count': 194, 'eval_duration': 13189788000}



Album Title	Artist Name
For Those About To Rock We Salute You	AC/DC
Balls to the Wall	Accept
Restless and Wild	Accept
Let There Be Rock	AC/DC
Big Ones	Aerosmith
Jagged Little Pill	Alanis Morissette
Facelift	Alice In Chains
Warner 25 Anos	Antônio Carlos Jobim
Plays Metallica By Four Cellos	Apocalyptica
Audioslave	Audioslave
Out Of Exile	Audioslave
BackBeat Soundtrack	BackBeat
The Best Of Billy Cobham	Billy Cobham
Alcohol Fueled Brewtality Live! [Disc 1]	Black Label Society
Alcohol Fueled Brewtality Live! [Disc 2]	Black Label Society

Out[20]: ('SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId',

```
                                          AlbumTitle  \
 0               For Those About To Rock We Salute You
 1                                   Balls to the Wall
 2                                   Restless and Wild
 3                                   Let There Be Rock
 4                                            Big Ones
 ..                                                ...
 342                           Respighi:Pines of Rome
 343  Schubert: The Late String Quartets & String Qu...
 344                              Monteverdi: L'Orfeo
 345                            Mozart: Chamber Music
 346  Koyaanisqatsi (Soundtrack from the Motion Pict...

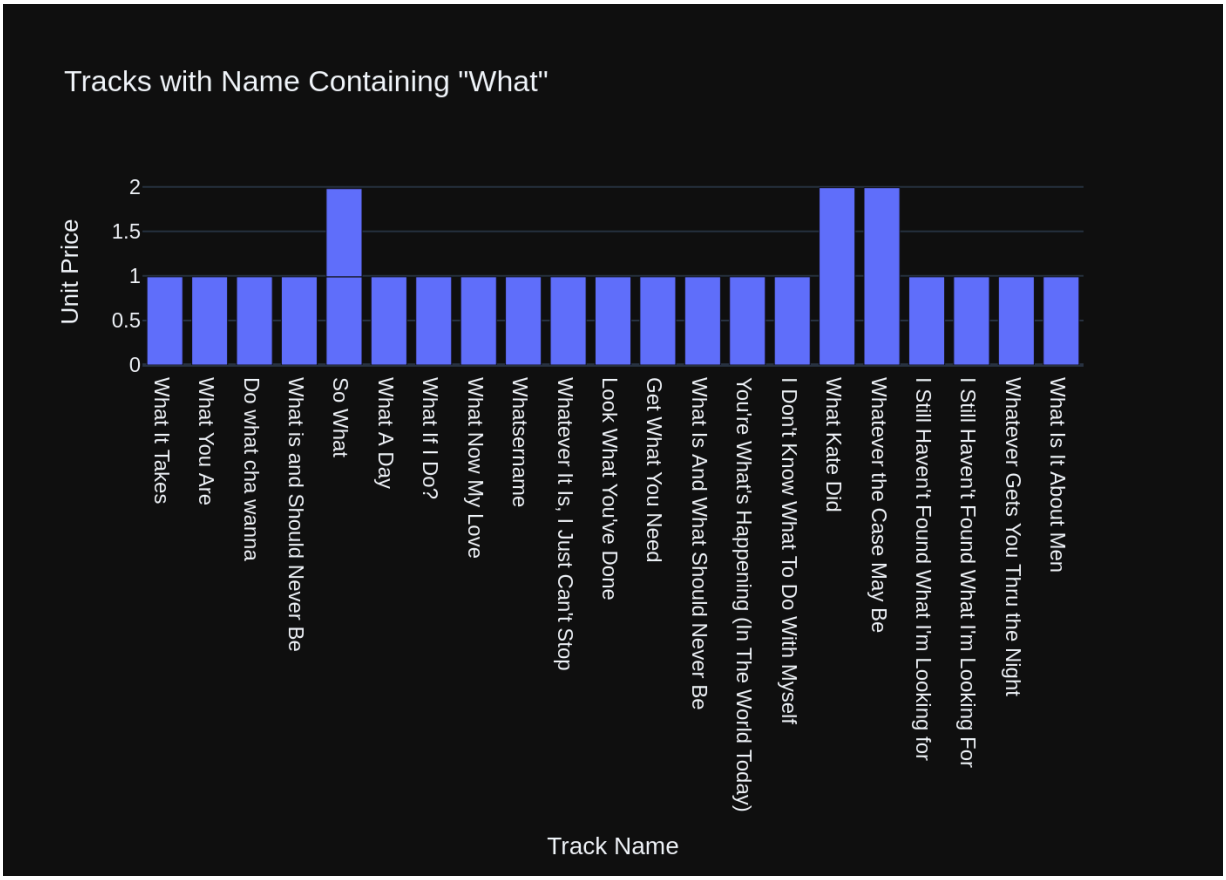
                                          ArtistName
 0                                             AC/DC
 1                                            Accept
 2                                            Accept
 3                                             AC/DC
 4                                           Aerosmith
 ..                   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

 [347 rows x 2 columns],
 Figure({
     'data': [{'cells': {'values': [['For Those About To Rock We Salute Yo
u',
                                     'Balls to the Wall', 'Restless and Wil
d', 'Let
                                     There Be Rock', 'Big Ones', 'Jagged Lit
tle
                                     Pill', 'Facelift', 'Warner 25 Anos', 'P
lays
                                     Metallica By Four Cellos', 'Audioslav
e', 'Out Of
                                     Exile', 'BackBeat Soundtrack', 'The Bes
t Of
                                     Billy Cobham', 'Alcohol Fueled Brewtali
ty Live!
                                     [Disc 1]', 'Alcohol Fueled Brewtality L
ive!
                                     [Disc 2]', 'Black Sabbath', 'Black Sabb
ath Vol.
                                     4 (Remaster)', 'Body Count', 'Chemical
Wedding',
                                     'The Best Of Buddy Guy - The Millenium
                                     Collection', 'Prenda Minha', 'Sozinho R
emix Ao
                                     Vivo', 'Minha Historia', 'Afrociberdeli
a', 'Da
                                     Lama Ao Caos', 'Acústico MTV [Live]',
```

'Cidade
001',
Fury',
1)',
(Disc
s', 'Heart
s', 'Into
tória',
'Physical
001',
'The
ssential
rts (Disc
- Sem
l. 1',
leção Sem
imite
Purple
Back
0's",
s And
ages On',

'Djavan
istória',
lbum Of
y Fool
xa
Your
Negra - Hits', 'Na Pista', 'Axé Bahia 2
'BBC Sessions [Disc 1] [Live]', 'Bongo
'Carnaval 2001', 'Chill: Brazil (Disc
'Chill: Brazil (Disc 2)', 'Garage Inc.
1)', 'Greatest Hits II', 'Greatest Kis
of the Night', 'International Superhit
The Light', 'Meus Momentos', 'Minha His
'MK III The Final Concerts [Disc 1]',
Graffiti [Disc 1]', 'Sambas De Enredo 2
'Supernatural', 'The Best of Ed Motta',
Essential Miles Davis [Disc 1]', 'The E
Miles Davis [Disc 2]', 'The Final Conce
2)', "Up An' Atom", 'Vinícius De Moraes
Limite', 'Vozes do MPB', 'Chronicle, Vo
'Chronicle, Vol. 2', 'Cássia Eller - Co
Limite [Disc 2]', 'Cássia Eller - Sem L
[Disc 1]', 'Come Taste The Band', 'Deep
In Rock', 'Fireball', "Knocking at Your
Door: The Best Of Deep Purple in the 8
'Machine Head', 'Purpendicular', 'Slave
Masters', 'Stormbringer', 'The Battle R
"Vault: Def Leppard's Greatest Hits",
'Outbreak', 'Djavan Ao Vivo - Vol. 02',
Ao Vivo - Vol. 1', 'Elis Regina-Minha H
'The Cream Of Clapton', 'Unplugged', 'A
The Year', 'Angel Dust', 'King For A Da
For A Lifetime', 'The Real Thing', 'Dei
Entrar', 'In Your Honor [Disc 1]', 'In

```
our And
Sinatra
de Eu Tu
'Quanta
celess',
tion',
on II',
th', 'A
ve New
Dark',
eath',
e At
r The
'Rock In
th Son of
he Number
XI',
th',
Cowboy',
rfing
n Jor 25
'Living
BC
ses Of
'IV', 'Led
ppelin
esence',
'The Song
```

```
Honor [Disc 2]', 'One By One', 'The Col
The Shape', 'My Way: The Best Of Frank
[Disc 1]', 'Roda De Funk', 'As Canções
Eles', 'Quanta Gente Veio Ver (Live)',
Gente Veio ver--Bônus De Carnaval', 'Fa
'American Idiot', 'Appetite for Destruc
'Use Your Illusion I', 'Use Your Illusi
'Blue Moods', 'A Matter of Life and Dea
Real Dead One', 'A Real Live One', 'Bra
World', 'Dance Of Death', 'Fear Of The
'Iron Maiden', 'Killers', 'Live After D
'Live At Donington 1992 (Disc 1)', 'Liv
Donington 1992 (Disc 2)', 'No Prayer Fo
Dying', 'Piece Of Mind', 'Powerslave',
Rio [CD1]', 'Rock In Rio [CD2]', 'Seven
a Seventh Son', 'Somewhere in Time', 'T
of The Beast', 'The X Factor', 'Virtual
'Sex Machine', 'Emergency On Planet Ear
'Synkronized', 'The Return Of The Space
'Get Born', 'Are You Experienced?', 'Su
with the Alien (Remastered)', 'Jorge Be
Anos', 'Jota Quest-1995', 'Cafezinho',
After Midnight', 'Unplugged [Live]', 'B
Sessions [Disc 2] [Live]', 'Coda', 'Hou
The Holy', 'In Through The Out Door',
Zeppelin I', 'Led Zeppelin II', 'Led Ze
III', 'Physical Graffiti [Disc 2]', 'Pr
'The Song Remains The Same (Disc 1)',
```

s', 'Mais
s - RCA
antos -
splaced
Shall
he Best
Inc.
ter Of
g', 'St.
s Ahead',
'Ace Of
atest
hkah
'Olodum',
Os
n
y of a

'The
'Pearl
ide Of
na Não
World',
In Hi-
s', 'Cesta
x Magik',
spective

Remains The Same (Disc 2)', 'A TempestadeTempestade Ou O Livro Dos Dia
Do Mesmo', 'Greatest Hits', 'Lulu Santo
100 Anos De Música - Álbum 01', 'Lulu S
RCA 100 Anos De Música - Álbum 02', 'Mi
Childhood', 'Barulhinho Bom', 'Seek And
Find: More Of The Best (1963-1981)', 'T
Of Men At Work', 'Black Album', 'Garage
(Disc 2)', "Kill 'Em All", 'Load', 'Mas
Puppets', 'ReLoad', 'Ride The Lightnin
Anger', '...And Justice For All', 'Mile
'Milton Nascimento Ao Vivo', 'Minas',
Spades', 'Demorou...', 'Motley Crue Gre
Hits', 'From The Muddy Banks Of The Wis
[Live]', 'Nevermind', 'Compositores',
'Acústico MTV', 'Arquivo II', 'Arquivo
Paralamas Do Sucesso', 'Bark at the Moo
(Remastered)', 'Blizzard of Ozz', 'Diar
Madman (Remastered)', 'No More Tears (Remastered)', 'Tribute', 'Walking Into Clarksdale', 'Original Soundtracks 1',
Beast Live', 'Live On Two Legs [Live]',
Jam', 'Riot Act', 'Ten', 'Vs.', 'Dark S
The Moon', 'Os Cães Ladram Mas A Carava
Pára', 'Greatest Hits I', 'News Of The
'Out Of Time', 'Green', 'New Adventures
Fi', 'The Best Of R.E.M.: The IRS Year
Básica', 'Raul Seixas', 'Blood Sugar Se
'By The Way', 'Californication', 'Retro
I (1974-1980)', 'Santana - As Years Go

By',
Poconé',
n Apples:
ce', 'In
ck Light
2]',
'Pure
s, Ravers,
'The
-1971
e',
ion - The
e (Disc
tico',
he Story
3',
'Lost,
aby',
ides
Bomb',

Best Of
Best Of
alen
s', 'Ao
'The
3', 'Un-
c), Season
sty

'Santana Live', 'Maquinarama', 'O Samba
'Judas 0: B-Sides and Rarities', 'Rotte
Greatest Hits', 'A-Sides', 'Morning Dan
Step', 'Core', 'Mezmerize', '[1997] Bla
Syndrome', 'Live [Disc 1]', 'Live [Disc
'The Singles', 'Beyond Good And Evil',
Cult: The Best Of The Cult (For Rocker
Lovers & Sinners) [UK]', 'The Doors',
Police Greatest Hits', 'Hot Rocks, 1964
(Disc 1)', 'No Security', 'Voodoo Loung
'Tangents', 'Transmission', 'My Generat
Very Best Of The Who', 'Serie Sem Limit
1)', 'Serie Sem Limite (Disc 2)', 'Acús
'Volume Dois', 'Battlestar Galactica: T
So Far', 'Battlestar Galactica, Season
'Heroes, Season 1', 'Lost, Season 3',
Season 1', 'Lost, Season 2', 'Achtung B
"All That You Can't Leave Behind", 'B-S
1980-1990', 'How To Dismantle An Atomic
'Pop', 'Rattle And Hum', 'The Best Of
1980-1990', 'War', 'Zooropa', 'UB40 The
- Volume Two [UK]', 'Diver Down', 'The
Van Halen, Vol. I', 'Van Halen', 'Van H
III', 'Contraband', 'Vinicius De Morae
Vivo [IMPORT]', 'The Office, Season 1',
Office, Season 2', 'The Office, Season
Led-Ed', 'Battlestar Galactica (Classi
1', 'Aquaman', 'Instant Karma: The Amne

'Speak
The
pions',
a Jovem
o', 'Cake:
4', 'Quiet
nd of
of
On',
Chant

'Vivaldi:
tos',
Cello
ts)',
ir
rt: Wind
4',
oprano
ner:
Ravel:
cracker',
i: Madama
anets,

r
na
I', 'Bach:

International Campaign to Save Darfur',
of the Devil', '20th Century Masters -
Millennium Collection: The Best of Scor
'House of Pain', 'Radio Brasil (O Som d
Vanguarda) - Seleccao de Henrique Amar
B-Sides and Rarities', 'LOST, Season
Songs', 'Muso Ko', 'Realize', 'Every Ki
Light', 'Duos II', 'Worlds', 'The Best
Beethoven', 'Temple of the Dog', 'Carry
'Revelations', 'Adorate Deum: Gregorian
from the Proper of the Mass', 'Allegri:
Miserere', 'Pachelbel: Canon & Gigue',
The Four Seasons', 'Bach: Violin Concer
'Bach: Goldberg Variations', 'Bach: The
Suites', 'Handel: The Messiah (Highligh
'The World of Classical Favourites', 'S
Neville Marriner: A Celebration', 'Moza
Concertos', 'Haydn: Symphonies 99 - 10
'Beethoven: Symhonies Nos. 5 & 6', 'A S
Inspired', 'Great Opera Choruses', 'Wag
Favourite Overtures', 'Fauré: Requiem,
Pavane & Others', 'Tchaikovsky: The Nut
'The Last Night of the Proms', 'Puccin
Butterfly - Highlights', 'Holst: The Pl
Op. 32 & Vaughan Williams: Fantasies',
"Pavarotti's Opera Made Easy", "Great
Performances - Barber's Adagio and Othe
Romantic Favorites for Strings", 'Carmi
Burana', 'A Copland Celebration, Vol.
Toccata & Fugue in D Minor', 'Prokofie

v:
The
Concertos
icana',
Sonatas:
gs of the
e',
ms:
Dances',
p.49,
on's
elli &
Juliet',
s',
or the
9)',
lléas et
s',
sic from
rt:
k',
'Pastoral'
s',
m", 'Bach:
tier:
th
o. 3',
Symphony No.1', 'Scheherazade', 'Bach:
Brandenburg Concertos', 'Chopin: Piano
Nos. 1 & 2', 'Mascagni: Cavalleria Rust
'Sibelius: Finlandia', 'Beethoven Piano
Moonlight & Pastorale', 'Great Recordin
Century - Mahler: Das Lied von der Erd
'Elgar: Cello Concerto & Vaughan Willia
Fantasias', 'Adams, John: The Chairman
"Tchaikovsky: 1812 Festival Overture, O
Capriccio Italien & Beethoven: Wellingt
Victory", 'Palestrina: Missa Papae Marc
Allegri: Miserere', 'Prokofiev: Romeo &
'Strauss: Waltzes', 'Berlioz: Symphonie
Fantastique', 'Bizet: Carmen Highlight
'English Renaissance', 'Handel: Music f
Royal Fireworks (Original Version 174
'Grieg: Peer Gynt Suites & Sibelius: Pe
Mélisande', 'Mozart Gala: Famous Aria
'SCRIABIN: Vers la flamme', 'Armada: Mu
the Courts of England and Spain', 'Moza
Symphonies Nos. 40 & 41', 'Back to Blac
'Frank', 'Carried to Dust (Bonus Track
Version)', "Beethoven: Symphony No. 6
Etc.", 'Bartok: Violin & Viola Concerto
"Mendelssohn: A Midsummer Night's Drea
Orchestral Suites Nos. 1 - 4', 'Charpen
Divertissements, Airs & Concerts', 'Sou
American Getaway', 'Górecki: Symphony N
'Purcell: The Fairy Queen', 'The Ultima

te
Relexation Album', 'Purcell: Music for
the Queen
Mary', 'Weill: The Seven Deadly Sins',
'J.S.
Bach: Chaconne, Suite in E Minor, Parti
ta in E
Major & Prelude, Fugue and Allegro', 'P
rokofiev:
Symphony No.5 & Stravinksy: Le Sacre Du
Printemps', 'Szymanowski: Piano Works,
Vol. 1',
'Nielsen: The Six Symphonies', "Great R
ecordings
of the Century: Paganini's 24 Caprice
s", "Liszt
- 12 Études D'Execution Transcendante",
'Great
Recordings of the Century - Shubert:
Concertos
Schwanengesang, 4 Lieder', 'Locatelli:
3',
for Violin, Strings and Continuo, Vol.
e Late
'Respighi:Pines of Rome', "Schubert: Th
D's)",
String Quartets & String Quintet (3 C
Music',
"Monteverdi: L'Orfeo", 'Mozart: Chamber
ion
'Koyaanisqatsi (Soundtrack from the Mot
t',
Picture)'], ['AC/DC', 'Accept', 'Accep
e',
'AC/DC', 'Aerosmith', 'Alanis Morissett
m',
'Alice In Chains', 'Antônio Carlos Jobi
ve',
'Apocalyptica', 'Audioslave', 'Audiosla
l
'BackBeat', 'Billy Cobham', 'Black Labe
Society', 'Black Label Society', 'Black
t', 'Bruce
Sabbath', 'Black Sabbath', 'Body Coun
Dickinson', 'Buddy Guy', 'Caetano Velos
o',
'Caetano Veloso', 'Chico Buarque', 'Chi
co
Science & Nação Zumbi', 'Chico Science
& Nação
Zumbi', 'Cidade Negra', 'Cidade Negra',
'Cláudio
Zoli', 'Various Artists', 'Led Zeppeli
n', 'Frank
Zappa & Captain Beefheart', 'Various Ar
tists',

```
                                                'Marcos Valle', 'Antônio Carlos Jobim',
                                                'Metallica', 'Queen', 'Kiss', 'Spyro Gy
ra',
                                                'Green Day', 'David Coverdale', 'Gonzag
uinha',
                                                'Os Mutantes', 'Deep Purple', 'Led Zepp
elin',
                                                'Various Artists', 'Santana', 'Ed Mott
a', 'Miles
                                                Davis', 'Miles Davis', 'Deep Purple',
'Gene
                                                Krupa', 'Toquinho & Vinícius', 'Various
                                                Artists', 'Creedence Clearwater Reviva
l',
                                                'Creedence Clearwater Revival', 'Cássia
Eller',
                                                'Cássia Eller', 'Deep Purple', 'Deep Pu
rple',
                                                'Deep Purple', 'Deep Purple', 'Deep Pur
ple',
                                                'Deep Purple', 'Deep Purple', 'Deep Pur
ple',
                                                'Deep Purple', 'Def Leppard', 'Dennis C
hambers',
                                                'Djavan', 'Djavan', 'Elis Regina', 'Eri
c
                                                Clapton', 'Eric Clapton', 'Faith No Mor
e',
                                                'Faith No More', 'Faith No More', 'Fait
h No
                                                More', 'Falamansa', 'Foo Fighters', 'Fo
o
                                                Fighters', 'Foo Fighters', 'Foo Fighter
s',
                                                'Frank Sinatra', 'Funk Como Le Gusta',
'Gilberto
                                                Gil', 'Gilberto Gil', 'Gilberto Gil',
                                                'Godsmack', 'Green Day', "Guns N' Rose
s", "Guns
                                                N' Roses", "Guns N' Roses", 'Incognit
o', 'Iron
                                                Maiden', 'Iron Maiden', 'Iron Maiden',
'Iron
                                                Maiden', 'Iron Maiden', 'Iron Maiden',
'Iron
                                                Maiden', 'Iron Maiden', 'Iron Maiden',
'Iron
                                                Maiden', 'Iron Maiden', 'Iron Maiden',
'Iron
                                                Maiden', 'Iron Maiden', 'Iron Maiden',
'Iron
                                                Maiden', 'Iron Maiden', 'Iron Maiden',
'James
                                                Brown', 'Jamiroquai', 'Jamiroquai',
```

```
                                                'Jamiroquai', 'JET', 'Jimi Hendrix', 'J
oe
                                                Satriani', 'Jorge Ben', 'Jota Quest',
'João
                                                Suplicy', 'Judas Priest', 'Kiss', 'Led
                                                Zeppelin', 'Led Zeppelin', 'Led Zeppeli
n', 'Led
                                                Zeppelin', 'Led Zeppelin', 'Led Zeppeli
n', 'Led
                                                Zeppelin', 'Led Zeppelin', 'Led Zeppeli
n', 'Led
                                                Zeppelin', 'Led Zeppelin', 'Led Zeppeli
n',
                                                'Legião Urbana', 'Legião Urbana', 'Lenn
y
                                                Kravitz', 'Lulu Santos', 'Lulu Santos',
                                                'Marillion', 'Marisa Monte', 'Marvin Ga
ye', 'Men
                                                At Work', 'Metallica', 'Metallica', 'Me
tallica',
                                                'Metallica', 'Metallica', 'Metallica',
                                                'Metallica', 'Metallica', 'Metallica',
'Miles
                                                Davis', 'Milton Nascimento', 'Milton
                                                Nascimento', 'Motörhead', 'Mônica Maria
nno',
                                                'Mötley Crüe', 'Nirvana', 'Nirvana', 'O
Terço',
                                                'Olodum', 'Os Paralamas Do Sucesso', 'O
s
                                                Paralamas Do Sucesso', 'Os Paralamas Do
                                                Sucesso', 'Ozzy Osbourne', 'Ozzy Osbour
ne',
                                                'Ozzy Osbourne', 'Ozzy Osbourne', 'Ozzy
                                                Osbourne', 'Page & Plant', 'Passenger
s', "Paul
                                                D'Ianno", 'Pearl Jam', 'Pearl Jam', 'Pe
arl Jam',
                                                'Pearl Jam', 'Pearl Jam', 'Pink Floyd',
'Planet
                                                Hemp', 'Queen', 'Queen', 'R.E.M. Feat.
Kate
                                                Pearson', 'R.E.M.', 'R.E.M.', 'R.E.M.',
                                                'Raimundos', 'Raul Seixas', 'Red Hot Ch
ili
                                                Peppers', 'Red Hot Chili Peppers', 'Red
Hot
                                                Chili Peppers', 'Rush', 'Santana', 'San
tana',
                                                'Skank', 'Skank', 'Smashing Pumpkins',
'Smashing
                                                Pumpkins', 'Soundgarden', 'Spyro Gyra',
'Stevie
                                                Ray Vaughan & Double Trouble', 'Stone T
emple
                                                Pilots', 'System Of A Down', 'Terry Boz
```

```
zio, Tony
                                                Levin & Steve Stevens', 'The Black Crow
es', 'The
                                                Black Crowes', 'The Clash', 'The Cult',
'The
                                                Cult', 'The Doors', 'The Police', 'The
Rolling
                                                Stones', 'The Rolling Stones', 'The Rol
ling
                                                Stones', 'The Tea Party', 'The Tea Part
y', 'The
                                                Who', 'Tim Maia', 'Tim Maia', 'Titãs',
'Titãs',
                                                'Battlestar Galactica', 'Battlestar Gal
actica',
                                                'Heroes', 'Lost', 'Lost', 'Lost', 'U2',
'U2',
                                                'U2', 'U2', 'U2', 'U2', 'U2', 'U2', 'U
2',
                                                'UB40', 'Van Halen', 'Van Halen', 'Van
Halen',
                                                'Van Halen', 'Velvet Revolver', 'Viníci
us De
                                                Moraes', 'Zeca Pagodinho', 'The Offic
e', 'The
                                                Office', 'The Office', 'Dread Zeppeli
n',
                                                'Battlestar Galactica (Classic)', 'Aqua
man',
                                                'U2', 'Ozzy Osbourne', 'Scorpions', 'Ho
use Of
                                                Pain', 'O Rappa', 'Cake', 'Lost', 'Aish
a Duo',
                                                'Habib Koité and Bamada', 'Karsh Kale',
'The
                                                Posies', 'Luciana Souza/Romero Lubamb
o', 'Aaron
                                                Goldberg', 'Nicolaus Esterhazy Sinfoni
a',
                                                'Temple of the Dog', 'Chris Cornell',
                                                'Audioslave', 'Alberto Turco & Nova Sch
ola
                                                Gregoriana', 'Richard Marlow & The Choi
r of
                                                Trinity College, Cambridge', 'English C
oncert &
                                                Trevor Pinnock', 'Anne-Sophie Mutter, H
erbert
                                                Von Karajan & Wiener Philharmoniker',
'Hilary
                                                Hahn, Jeffrey Kahane, Los Angeles Chamb
er
                                                Orchestra & Margaret Batjer', 'Wilhelm
Kempff',
                                                'Yo-Yo Ma', 'Scholars Baroque Ensembl
e',
```

```
Sir
in in the
arriner',
o &
estra &
ionnaire
'Britten
t',
ony
g Solti &
Martin
Marriner
estra &
th & BBC
n,
'Eugene
ston
on
'Ton
rkanov',
ner',
t',
hia
io
dt,
beck de
```

```
'Academy of St. Martin in the Fields &
Neville Marriner', 'Academy of St. Mart
Fields Chamber Ensemble & Sir Neville M
'Berliner Philharmoniker, Claudio Abbad
Sabine Meyer', 'Royal Philharmonic Orch
Sir Thomas Beecham', 'Orchestre Révolut
et Romantique & John Eliot Gardiner',
Sinfonia, Ivor Bolton & Lesley Garret
'Chicago Symphony Chorus, Chicago Symph
Orchestra & Sir Georg Solti', 'Sir Geor
Wiener Philharmoniker', 'Academy of St.
in the Fields, John Birch, Sir Neville
& Sylvia McNair', 'London Symphony Orch
Sir Charles Mackerras', 'Barry Wordswor
Concert Orchestra', 'Herbert Von Karaja
Mirella Freni & Wiener Philharmoniker',
Ormandy', 'Luciano Pavarotti', 'Leonard
Bernstein & New York Philharmonic', 'Bo
Symphony Orchestra & Seiji Ozawa', 'Aar
Copland & London Symphony Orchestra',
Koopman', 'Sergei Prokofiev & Yuri Temi
'Chicago Symphony Orchestra & Fritz Rei
'Orchestra of The Age of Enlightenmen
'Emanuel Ax, Eugene Ormandy & Philadelp
Orchestra', 'James Levine', 'Berliner
Philharmoniker & Hans Rosbaud', 'Mauriz
Pollini', 'Gustav Mahler', 'Felix Schmi
London Symphony Orchestra & Rafael Früh
Burgos', 'Edo de Waart & San Francisco
Symphony', 'Antal Doráti & London Symph
```

ony
& Simon
Francisco
ilson
der
& Wiener
'English

'Sir
niker',
liner
'Amy
o', 'Otto
ehudi
Neville
e Fields,
'Les Arts
2
'Adrian
gton,
oit &
'Equale
unich
Nagano
ian
ert Von
ymfoniker
ele
nbaum,

Orchestra', 'Choir Of Westminster Abbey
Preston', 'Michael Tilson Thomas & San
Symphony', 'Eugene Ormandy', 'Michael T
Thomas & San Francisco Symphony', 'Chor
Wiener Staatsoper, Herbert Von Karajan
Philharmoniker', "The King's Singers",
Concert & Trevor Pinnock', 'Berliner
Philharmoniker & Herbert Von Karajan',
Georg Solti, Sumi Jo & Wiener Philharmo
"Christopher O'Riley", 'Fretwork', 'Ber
Philharmoniker & Herbert Von Karajan',
Winehouse', 'Amy Winehouse', 'Calexic
Klemperer & Philharmonia Orchestra', 'Y
Menuhin', 'Philharmonia Orchestra & Sir
Marriner', 'Academy of St. Martin in th
Sir Neville Marriner & Thurston Dart',
Florissants & William Christie', 'The 1
Cellists of The Berlin Philharmonic',
Leaper & Doreen de Feis', 'Roger Norrin
London Classical Players', "Charles Dut
L'Orchestre Symphonique de Montréal",
Brass Ensemble, John Eliot Gardiner & M
Monteverdi Orchestra and Choir', "Kent
and Orchestre de l'Opéra de Lyon", 'Jul
Bream', 'Berliner Philharmoniker & Herb
Karajan', 'Martin Roscoe', 'Göteborgs S
& Neeme Järvi', 'Itzhak Perlman', 'Mich
Campanella', 'Gerald Moore', 'Mela Tene

```
                                          Pro Musica Prague & Richard Kapp', 'Eug
ene
                                          Ormandy', 'Emerson String Quartet', 'C.
                                          Monteverdi, Nigel Rogers - Chiaroscuro;
London
                                          Baroque; London Cornett & Sackbu', 'Nas
h
                                          Ensemble', 'Philip Glass Ensemble']]},
                'header': {'values': ['Album Title', 'Artist Name']},
                'type': 'table'}],
     'layout': {'template': '...'}
 }))
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON "tracks" (Ge
nreId)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackI
d)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGE
R,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Comp
oser NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTE
GER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId)
REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DEL
ETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENC
ES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTI
ON\r\n)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE IND
EX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE TABLE "playlist_
track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT
NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackI
d),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n
\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) RE
FERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION
\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n
\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE TABLE "pla
ylists"\r\n(\r\n    PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n    Name NVARCHAR(120)\r\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n
\n\n===Additional Context \n\nIn the chinook database invoice means order\n
\n===Response Guidelines \n1. If the provided context is sufficient, please
generate a valid SQL query without any explanations for the question. \n2. I
f the provided context is almost sufficient but requires knowledge of a spec
ific string in a particular column, please generate an intermediate SQL quer
y to find the distinct strings in that column. Prepend the query with a comm
ent saying intermediate_sql \n3. If the provided context is insufficient, pl
ease explain why it can\'t be generated. \n4. Please use the most relevant t
able(s). \n5. If the question has been asked and answered before, please rep
eat the answer exactly as it was given before. \n'}, {'role': 'user', 'conte
nt': '  \n    List all albums and their corresponding artist names  \n'},
{'role': 'assistant', 'content': 'SELECT a.Title AS AlbumTitle, ar.Name AS A
rtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId'}, {'r
ole': 'user', 'content': 'Can you list all tables in the SQLite database cat
alog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WH
ERE type='table'"}, {'role': 'user', 'content': 'How many customers are ther
e'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers
FROM customers'}, {'role': 'user', 'content': '  \n    Find all tracks with
a name containing "What" (case-insensitive)\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\n
CREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCRE
ATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT N
```

OT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n
MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARC
HAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDEX
IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE TABLE \"playlist_
track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NO
T NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackI
d),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId)
REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackI
d)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\nCREATE TABL
E \"playlists\"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"genres\"\r\n(\r\n
GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(12
0)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means
order\n\n===Response Guidelines \n1. If the provided context is sufficient,
please generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can't be generated. \n4. Please use the most relev
ant table(s). \n5. If the question has been asked and answered before, pleas
e repeat the answer exactly as it was given before. \n"}, {"role": "user",
"content": " \n   List all albums and their corresponding artist names
\n"}, {"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Nam
e AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistI
d"}, {"role": "user", "content": "Can you list all tables in the SQLite data
base catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_m
aster WHERE type='table'"}, {"role": "user", "content": "How many customers
are there"}, {"role": "assistant", "content": "SELECT COUNT(*) AS NumberOfCu
stomers FROM customers"}, {"role": "user", "content": " \n   Find all trac
ks with a name containing \"What\" (case-insensitive)\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:13:47.828
568393Z', 'message': {'role': 'assistant', 'content': "```sql\nSELECT * \nFR
OM tracks \nWHERE LOWER(Name) LIKE '%what%';\n```"}, 'done_reason': 'stop',
'done': True, 'total_duration': 22507278436, 'load_duration': 574083, 'promp
t_eval_count': 834, 'prompt_eval_duration': 20527774000, 'eval_count': 25,
'eval_duration': 1751281000}
LLM Response: ```sql
SELECT *
FROM tracks
WHERE LOWER(Name) LIKE '%what%';
```
Info: Output from LLM: ```sql
SELECT *
FROM tracks
WHERE LOWER(Name) LIKE '%what%';
```

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

    MediaTypeId  GenreId                                         Composer
\
0             1        1            Steven Tyler, Joe Perry, Desmond Child
1             1        1                          Audioslave/Chris Cornell
2             1        2                                       George Duke
3             1        1                        Jimmy Page/Robert Plant
4             1        2                                       Miles Davis
5             1        1             Mike Bordin, Billy Gould, Mike Patton
6             1        1  Dave Grohl, Taylor Hawkins, Nate Mendel, Chris...
7             1       12         carl sigman/gilbert becaud/pierre leroyer
8             1        4                                        Green Day
9             1        1                                  Jay Kay/Kay, Jay
10            1        4                                        N. Cester
11            1        4                      C. Cester/C. Muncey/N. Cester
12            1        1                           Jimmy Page, Robert Plant
13            1       14             Allen Story/George Gordy/Robert Gordy
14            1        3                                  Culmer/Exalt
15            1        7                                          None
16            3       19                                          None
17            3       19                                          None
18            1        1     Bono/Clayton, Adam/Mullen Jr., Larry/The Edge
19            1        1                                            U2
20            2        9                                          None
21            2        9  Delroy "Chris" Cooper, Donovan Jackson, Earl C...

    Milliseconds      Bytes  UnitPrice
```

```
0         310622   10144730       0.99
1         249391    5988186       0.99
2         274155    9018565       0.99
3         260675    8497116       0.99
4         564009   18360449       0.99
5         158275    5203430       0.99
6         302994    9929799       0.99
7         149995    4913383       0.99
8         252316    8244843       0.99
9         247222    8249453       0.99
10        230974    7517083       0.99
11        247719    8043765       0.99
12        287973    9369385       0.99
13        142027    4631104       0.99
14        189152    6162894       0.99
15        221387    7251478       0.99
16       2610250  484583988       1.99
17       2616410  183867185       1.99
18        353567   11542247       0.99
19        280764    9306737       0.99
20        215084    3499018       0.99
21        209573    3426106       0.99
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find all tracks with a name containing \"What\" (case-insensitive)
\n'\n\nThe DataFrame was produced using this query: SELECT * \nFROM tracks
\nWHERE LOWER(Name) LIKE '%what%'\n\nThe following is information about the
resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n TrackId
int64\nName             object\nAlbumId           int64\nMediaTypeId       i
nt64\nGenreId           int64\nComposer         object\nMilliseconds      in
t64\nBytes             int64\nUnitPrice       float64\ndtype: object"}, {"ro
le": "user", "content": "Can you generate the Python plotly code to chart th
e results of the dataframe? Assume the data is in a pandas dataframe called
'df'. If there is only one value in the dataframe, use an Indicator. Respond
with only Python code. Do not answer with any explanations -- just the cod
e."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:14:02.816
506036Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\n# Check if the DataFrame ha
s more than one row\nif len(df) > 1:\n    fig = go.Figure(data=[go.Bar(x=df
[\'Name\'], y=df[\'UnitPrice\'])])\nelse:\n    fig = go.Figure(data=[go.Indi
cator(mode="number", value=df[\'UnitPrice\'].iloc[0], number={\'font\': {\'s
ize\': 48}})] )\n\nfig.update_layout(title=\'Tracks with Name Containing "Wh
at"\', xaxis_title=\'Track Name\', yaxis_title=\'Unit Price\')\nfig.show()\n
```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 14958827577,
'load_duration': 598725, 'prompt_eval_count': 219, 'prompt_eval_duration': 4
442005000, 'eval_count': 155, 'eval_duration': 10424408000}
```
```


Tracks with Name Containing "What"
Unit Price
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
Track Name

Out[21]: ("SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'",

	TrackId	Name	AlbumId	MediaTypeId	GenreId	Composer
0	26	What It Takes	5	1	1	Steven Tyler, Joe Perry, Desmond Child
1	88	What You Are	10	1	1	Audioslave/Chris Cornell
2	130	Do what cha wanna	13	1	2	George Duke
3	342	What is and Should Never Be	30	1	1	Jimmy Page/Robert Plant
4	607	So What	48	1	2	Miles Davis
5	960	What A Day	76	1	1	Mike Bordin, Billy Gould, Mike Patton
6	1000	What If I Do?	80	1	1	Dave Grohl, Taylor Hawkins, Nate Mendel, Chris...
7	1039	What Now My Love	83	1	12	carl sigman/gilbert becaud/pierre leroyer
8	1145	Whatsername	89	1	4	Green Day
9	1440	Whatever It Is, I Just Can't Stop	116	1	1	Jay Kay/Kay, Jay
10	1469	Look What You've Done	119	1	4	N. Cester
11	1470	Get What You Need	119	1	4	C. Cester/C. Muncey/N. Cester
12	1628	What Is And What Should Never Be	133	1	1	Jimmy Page, Robert Plant
13	1778	You're What's Happening (In The World Today)	146	1	14	Allen Story/George Gordy/Robert Gordy
14	1823	So What	149	1	3	Culmer/Exal

```
t
 15            1        7                                                  Non
e
 16            3       19                                                  Non
e
 17            3       19                                                  Non
e
 18            1        1      Bono/Clayton, Adam/Mullen Jr., Larry/The Edg
e
 19            1        1                                                    U
2
 20            2        9                                                  Non
e
 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': [{'type': 'bar',
               'x': array(['What It Takes', 'What You Are', 'Do what cha wa
nna',
                           '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 Kat
e 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),
                '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])}],
     'layout': {'template': '...',
                'title': {'text': 'Tracks with Name Containing "What"'},
                'xaxis': {'title': {'text': 'Track Name'}},
                'yaxis': {'title': {'text': 'Unit Price'}}}
 }))
```

In [22]:
```
question = """
    Get the total number of invoices for each customer
"""

vn.ask(question=question)
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE IN
DEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE
"invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NU
LL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT N
ULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFE
RENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nC
REATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCRE
MENT 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(7
0),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVAR
CHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_
CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employee
s"\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 DATETIM
E,\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    Posta
lCode 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 IN
DEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "track
s"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Na
me NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEG
ER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Genr
eId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaType
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional
Context \n\nIn the chinook database invoice means order\n\n===Response Guide
lines \n1. If the provided context is sufficient, please generate a valid SQ
L query without any explanations for the question. \n2. If the provided cont
ext is almost sufficient but requires knowledge of a specific string in a pa
rticular column, please generate an intermediate SQL query to find the disti
nct strings in that column. Prepend the query with a comment saying intermed
iate_sql \n3. If the provided context is insufficient, please explain why it
can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the
question has been asked and answered before, please repeat the answer exactl
y as it was given before. \n'}, {'role': 'user', 'content': 'How many custom
```

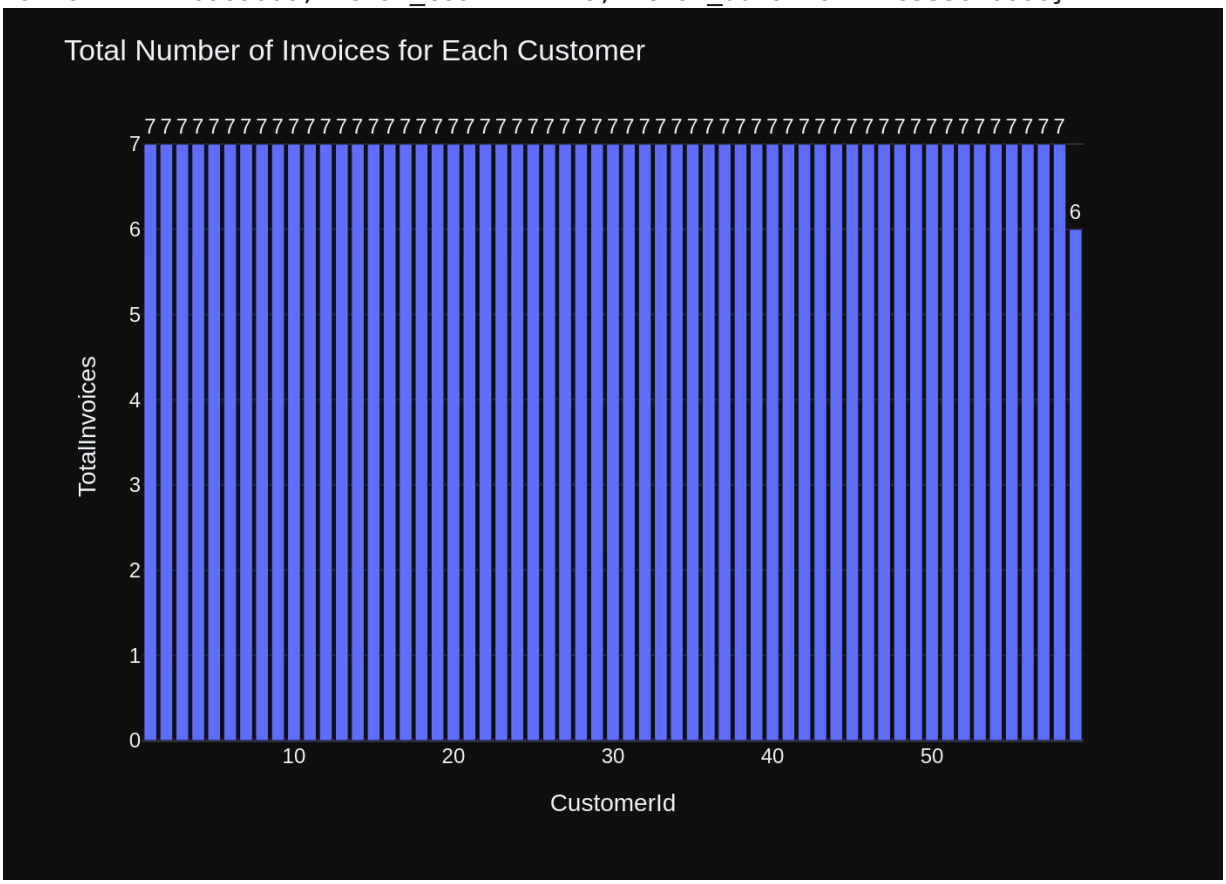
ers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) AS Number OfCustomers FROM customers'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a \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 LOWER (Name) LIKE '%what%'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLite database catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene rate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructi ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK _InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"inv oice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NUL L,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NU LL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n \t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) RE FERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO N\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId) \n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY AU TOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   LastNa me NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVARC HAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n   SupportRepId I NTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (Employee Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK _CustomerSupportRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"empl oyees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r \n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NU LL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DA TETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City N VARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(2 4),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"e mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n) \n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\nCREAT E TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   Me diaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHA

```
R(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means ord
er\n\n===Response Guidelines \n1. If the provided context is sufficient, ple
ase generate a valid SQL query without any explanations for the question. \n
2. If the provided context is almost sufficient but requires knowledge of a
specific string in a particular column, please generate an intermediate SQL
query to find the distinct strings in that column. Prepend the query with a
comment saying intermediate_sql \n3. If the provided context is insufficien
t, please explain why it can't be generated. \n4. Please use the most releva
nt table(s). \n5. If the question has been asked and answered before, please
repeat the answer exactly as it was given before. \n"}, {"role": "user", "co
ntent": "How many customers are there"}, {"role": "assistant", "content": "S
ELECT COUNT(*) AS NumberOfCustomers FROM customers"}, {"role": "user", "cont
ent": "  \n    List all albums and their corresponding artist names  \n"},
{"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS A
rtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"r
ole": "user", "content": "  \n    Find all tracks with a name containing \"W
hat\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \nF
ROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content":
"Can you list all tables in the SQLite database catalog?"}, {"role": "assist
ant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"rol
e": "user", "content": "  \n    Get the total number of invoices for each cu
stomer\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:14:42.613
977236Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId,
c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers
c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerI
d'}, 'done_reason': 'stop', 'done': True, 'total_duration': 39681295874, 'lo
ad_duration': 615529, 'prompt_eval_count': 1315, 'prompt_eval_duration': 352
93718000, 'eval_count': 52, 'eval_duration': 4112080000}
LLM Response: SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceI
d) AS TotalInvoices
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInv
oices
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
    CustomerId  FirstName      LastName  TotalInvoices
0            1       Luís     Gonçalves             7
1            2     Leonie        Köhler             7
2            3   François      Tremblay             7
3            4      Bjørn        Hansen             7
4            5  František   Wichterlová             7
5            6     Helena          Holý             7
6            7     Astrid        Gruber             7
7            8       Daan       Peeters             7
8            9       Kara       Nielsen             7
```

9	10	Eduardo	Martins	7
10	11	Alexandre	Rocha	7
11	12	Roberto	Almeida	7
12	13	Fernanda	Ramos	7
13	14	Mark	Philips	7
14	15	Jennifer	Peterson	7
15	16	Frank	Harris	7
16	17	Jack	Smith	7
17	18	Michelle	Brooks	7
18	19	Tim	Goyer	7
19	20	Dan	Miller	7
20	21	Kathy	Chase	7
21	22	Heather	Leacock	7
22	23	John	Gordon	7
23	24	Frank	Ralston	7
24	25	Victor	Stevens	7
25	26	Richard	Cunningham	7
26	27	Patrick	Gray	7
27	28	Julia	Barnett	7
28	29	Robert	Brown	7
29	30	Edward	Francis	7
30	31	Martha	Silk	7
31	32	Aaron	Mitchell	7
32	33	Ellie	Sullivan	7
33	34	João	Fernandes	7
34	35	Madalena	Sampaio	7
35	36	Hannah	Schneider	7
36	37	Fynn	Zimmermann	7
37	38	Niklas	Schröder	7
38	39	Camille	Bernard	7
39	40	Dominique	Lefebvre	7
40	41	Marc	Dubois	7
41	42	Wyatt	Girard	7
42	43	Isabelle	Mercier	7
43	44	Terhi	Hämäläinen	7
44	45	Ladislav	Kovács	7
45	46	Hugh	O'Reilly	7
46	47	Lucas	Mancini	7
47	48	Johannes	Van der Berg	7
48	49	Stanisław	Wójcik	7
49	50	Enrique	Muñoz	7
50	51	Joakim	Johansson	7
51	52	Emma	Jones	7
52	53	Phil	Hughes	7
53	54	Steve	Murray	7
54	55	Mark	Taylor	7
55	56	Diego	Gutiérrez	7
56	57	Luis	Rojas	7
57	58	Manoj	Pareek	7
58	59	Puja	Srivastava	6

```
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
```

tains the results of the query that answers the question the user asked: '
\n   Get the total number of invoices for each customer\n'\n\nThe DataFrame
was produced using this query: SELECT c.CustomerId, c.FirstName, c.LastName,
COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i
ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\n\nThe following is in
formation about the resulting pandas DataFrame 'df': \nRunning df.dtypes giv
es:\n CustomerId   int64\nFirstName   object\nLastName   obj
ect\nTotalInvoices   int64\ndtype: object"}, {"role": "user", "content":
"Can you generate the Python plotly code to chart the results of the datafra
me? Assume the data is in a pandas dataframe called 'df'. If there is only o
ne value in the dataframe, use an Indicator. Respond with only Python code.
Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:14:55.497
215004Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.express as px\n\nif len(df) == 1:\n   fig = px.indicator(value=df[\'To
talInvoices\'].iloc[0], title="Total Number of Invoices")\nelse:\n   fig =
px.bar(df, x=\'CustomerId\', y=\'TotalInvoices\', text=\'TotalInvoices\')\n
fig.update_traces(texttemplate=\'%{text}\', textposition=\'outside\')\n   f
ig.update_layout(title="Total Number of Invoices for Each Customer")\n\nfig.
show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 128563
23883, 'load_duration': 43644151, 'prompt_eval_count': 214, 'prompt_eval_dur
ation': 4226909000, 'eval_count': 128, 'eval_duration': 8533016000}



Out[22]: ('SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS Total Invoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId',

	CustomerId	FirstName	LastName	TotalInvoices
0	1	Luís	Gonçalves	7
1	2	Leonie	Köhler	7
2	3	François	Tremblay	7
3	4	Bjørn	Hansen	7
4	5	František	Wichterlová	7
5	6	Helena	Holý	7
6	7	Astrid	Gruber	7
7	8	Daan	Peeters	7
8	9	Kara	Nielsen	7
9	10	Eduardo	Martins	7
10	11	Alexandre	Rocha	7
11	12	Roberto	Almeida	7
12	13	Fernanda	Ramos	7
13	14	Mark	Philips	7
14	15	Jennifer	Peterson	7
15	16	Frank	Harris	7
16	17	Jack	Smith	7
17	18	Michelle	Brooks	7
18	19	Tim	Goyer	7
19	20	Dan	Miller	7
20	21	Kathy	Chase	7
21	22	Heather	Leacock	7
22	23	John	Gordon	7
23	24	Frank	Ralston	7
24	25	Victor	Stevens	7
25	26	Richard	Cunningham	7
26	27	Patrick	Gray	7
27	28	Julia	Barnett	7
28	29	Robert	Brown	7
29	30	Edward	Francis	7
30	31	Martha	Silk	7
31	32	Aaron	Mitchell	7
32	33	Ellie	Sullivan	7
33	34	João	Fernandes	7
34	35	Madalena	Sampaio	7
35	36	Hannah	Schneider	7
36	37	Fynn	Zimmermann	7
37	38	Niklas	Schröder	7
38	39	Camille	Bernard	7
39	40	Dominique	Lefebvre	7
40	41	Marc	Dubois	7
41	42	Wyatt	Girard	7
42	43	Isabelle	Mercier	7
43	44	Terhi	Hämäläinen	7
44	45	Ladislav	Kovács	7
45	46	Hugh	O'Reilly	7
46	47	Lucas	Mancini	7
47	48	Johannes	Van der Berg	7
48	49	Stanisław	Wójcik	7
49	50	Enrique	Muñoz	7
50	51	Joakim	Johansson	7
51	52	Emma	Jones	7

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

In [23]:
```
question = """
    Find the total number of invoices per country:
"""
```

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

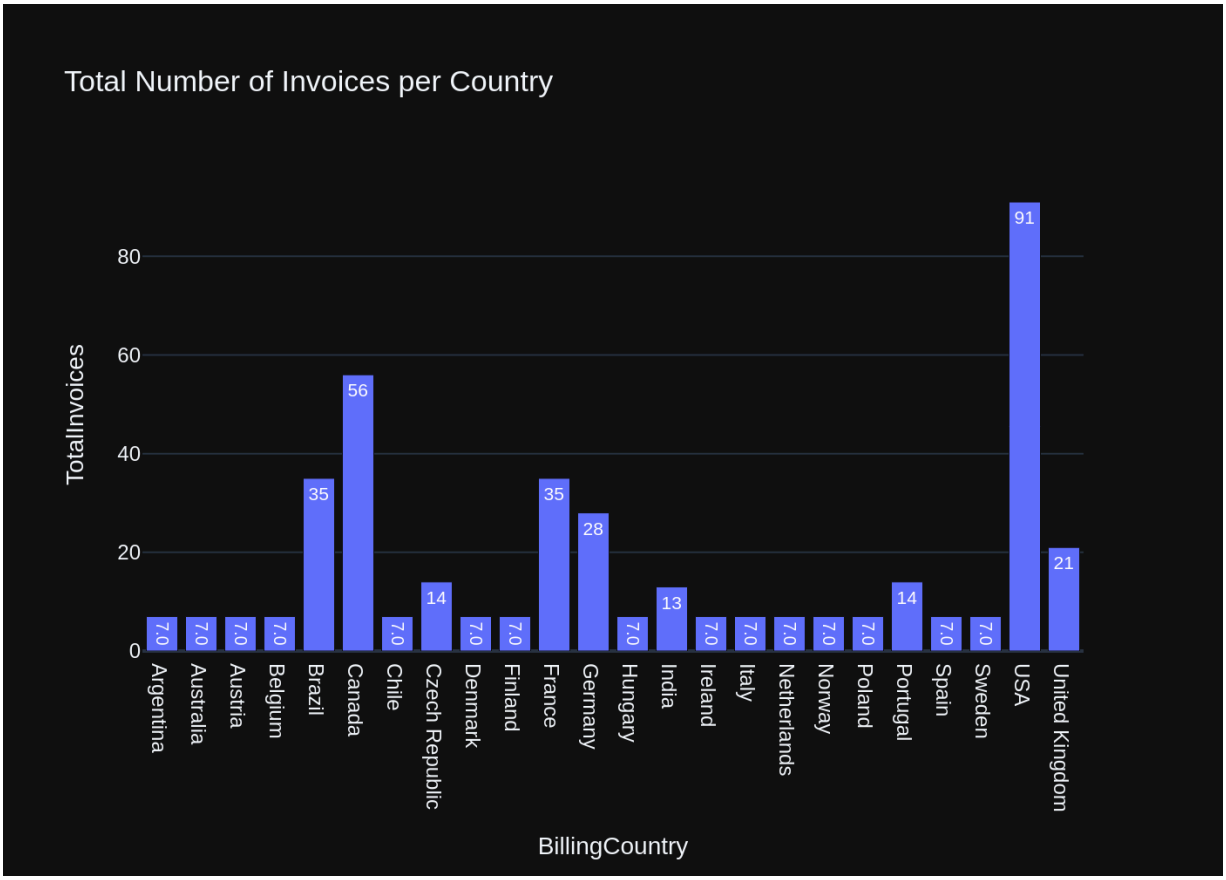
```
Number of requested results 10 is greater than number of elements in index
5, updating n_results = 5
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

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

lease use the most relevant table(s). \n5. If the question has been asked an
d answered before, please repeat the answer exactly as it was given before.
\n'}, {'role': 'user', 'content': ' \n   Get the total number of invoices
for each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerI
d, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM custom
ers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.Custo
merId'}, {'role': 'user', 'content': 'How many customers are there'}, {'rol
e': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers FROM custo
mers'}, {'role': 'user', 'content': ' \n   List all albums and their corre
sponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT a.Titl
e AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.
ArtistId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   Find all trac
ks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistan
t', 'content': "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"},
{'role': 'user', 'content': 'Can you list all tables in the SQLite database
catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master
WHERE type='table'"}, {'role': 'user', 'content': ' \n   Find the total nu
mber of invoices per country:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerI
d)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCR
EATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVA
RCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r
\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR
(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NV
ARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo)
REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n
Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r
\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n

```
SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"albums\"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT
NULL,\r\n    FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r
\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name N
VARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER
NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Mill
iseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(1
0,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreI
d) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTy
peId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX I
FK_EmployeeReportsTo ON \"employees\" (ReportsTo)\n\n\n===Additional Context
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n
1. If the provided context is sufficient, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is a
lmost sufficient but requires knowledge of a specific string in a particular
column, please generate an intermediate SQL query to find the distinct strin
gs in that column. Prepend the query with a comment saying intermediate_sql
\n3. If the provided context is insufficient, please explain why it can't be
generated. \n4. Please use the most relevant table(s). \n5. If the question
has been asked and answered before, please repeat the answer exactly as it w
as given before. \n"}, {"role": "user", "content": "  \n    Get the total nu
mber of invoices for each customer\n"}, {"role": "assistant", "content": "SE
LECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoi
ces\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\n
GROUP BY c.CustomerId"}, {"role": "user", "content": "How many customers are
there"}, {"role": "assistant", "content": "SELECT COUNT(*) AS NumberOfCustom
ers FROM customers"}, {"role": "user", "content": "  \n    List all albums a
nd their corresponding artist names  \n"}, {"role": "assistant", "content":
"SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN ar
tists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": "  \n
Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"ro
le": "assistant", "content": "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIK
E '%what%'"}, {"role": "user", "content": "Can you list all tables in the SQ
Lite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM
sqlite_master WHERE type='table'"}, {"role": "user", "content": "  \n    Fin
d the total number of invoices per country:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:15:37.477
017044Z', 'message': {'role': 'assistant', 'content': 'SELECT BillingCountr
y, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry'}, 'don
e_reason': 'stop', 'done': True, 'total_duration': 41853683474, 'load_durati
on': 589025, 'prompt_eval_count': 1447, 'prompt_eval_duration': 40095296000,
'eval_count': 19, 'eval_duration': 1436337000}
LLM Response: SELECT BillingCountry, COUNT(*) AS TotalInvoices
FROM invoices
GROUP BY BillingCountry
SELECT BillingCountry, COUNT(*) AS TotalInvoices
FROM invoices
GROUP BY BillingCountry
    BillingCountry  TotalInvoices
0        Argentina             7
```

```
1        Australia             7
2          Austria             7
3          Belgium             7
4           Brazil            35
5           Canada            56
6            Chile             7
7   Czech Republic            14
8          Denmark             7
9          Finland             7
10          France            35
11         Germany            28
12         Hungary             7
13           India            13
14         Ireland             7
15           Italy             7
16     Netherlands             7
17          Norway             7
18          Poland             7
19        Portugal            14
20           Spain             7
21          Sweden             7
22             USA            91
23  United Kingdom            21
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find the total number of invoices per country:\n'\n\nThe DataFrame was
produced using this query: SELECT BillingCountry, COUNT(*) AS TotalInvoices
\nFROM invoices\nGROUP BY BillingCountry\n\nThe following is information abo
ut the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Billing
Country    object\nTotalInvoices      int64\ndtype: object"}, {"role": "use
r", "content": "Can you generate the Python plotly code to chart the results
of the dataframe? Assume the data is in a pandas dataframe called 'df'. If t
here is only one value in the dataframe, use an Indicator. Respond with only
Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:15:47.906
666737Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.express as px\n\nif df[\'TotalInvoices\'].nunique() == 1:\n    fig = p
x.indicator(value=df[\'TotalInvoices\'].iloc[0], title="Total Invoices")\nel
se:\n    fig = px.bar(df, x=\'BillingCountry\', y=\'TotalInvoices\', text_au
to=\'.2s\', title=\'Total Number of Invoices per Country\')\n\nfig.show()\n`
``'}, 'done_reason': 'stop', 'done': True, 'total_duration': 10409726168, 'l
oad_duration': 43525444, 'prompt_eval_count': 171, 'prompt_eval_duration': 3
409517000, 'eval_count': 105, 'eval_duration': 6905154000}
```


Total Number of Invoices per Country
TotalInvoices
0
20
40
60
80
7.0
7.0
7.0
7.0
35
56
7.0
14
7.0
7.0
35
28
7.0
13
7.0
7.0
7.0
7.0
7.0
14
7.0
7.0
91
21
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
BillingCountry

Out[23]:
```
('SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY
BillingCountry',
     BillingCountry  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': 'BillingCountry=%{x}<br>TotalInvoices=%{y}<
extra></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'texttemplate': '%{y:.2s}',
               'type': 'bar',
               'x': array(['Argentina', 'Australia', 'Austria', 'Belgium',
'Brazil', 'Canada',
                           'Chile', 'Czech Republic', 'Denmark', 'Finland',
'France', 'Germany',
                           'Hungary', 'India', 'Ireland', 'Italy', 'Netherl
ands', 'Norway',
                           'Poland', 'Portugal', 'Spain', 'Sweden', 'USA',
'United Kingdom'],
                          dtype=object),
               'xaxis': 'x',
               'y': array([ 7,  7,  7,  7, 35, 56,  7, 14,  7,  7, 35, 28,
7, 13,  7,  7,  7,  7,
                            7, 14,  7,  7, 91, 21]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
```

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

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

vn.ask(question=question)
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoice_items"\r\n(\r\n    Invo
iceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEG
ER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,
2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (Invoice
Id) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLi
neInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE "invoices"\r\n(\r
\n    InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerI
d INTEGER  NOT NULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAd
dress NVARCHAR(70),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVA
RCHAR(40),\r\n    BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVAR
CHAR(10),\r\n    Total NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (Customer
Id) REFERENCES "customers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items"
(TrackId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n
\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n
MediaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARC
HAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES
"albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    F
OREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_t
ypes" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n
CREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABL
E "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\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    C
ity NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r
\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHA
R(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEGER,\r\n
FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r\n\t\tON DE
LETE 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 NVARC
HAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Title NVA
RCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n    HireD
ate 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 NVARC
HAR(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_Cus
tomerSupportRepId ON "customers" (SupportRepId)\n\n\n===Additional Context
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n
1. If the provided context is sufficient, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is a
lmost sufficient but requires knowledge of a specific string in a particular
column, please generate an intermediate SQL query to find the distinct strin
gs in that column. Prepend the query with a comment saying intermediate_sql
\n3. If the provided context is insufficient, please explain why it can\'t b
e generated. \n4. Please use the most relevant table(s). \n5. If the questio
n has been asked and answered before, please repeat the answer exactly as it
was given before. \n'}, {'role': 'user', 'content': '  \n    Get the total n
```

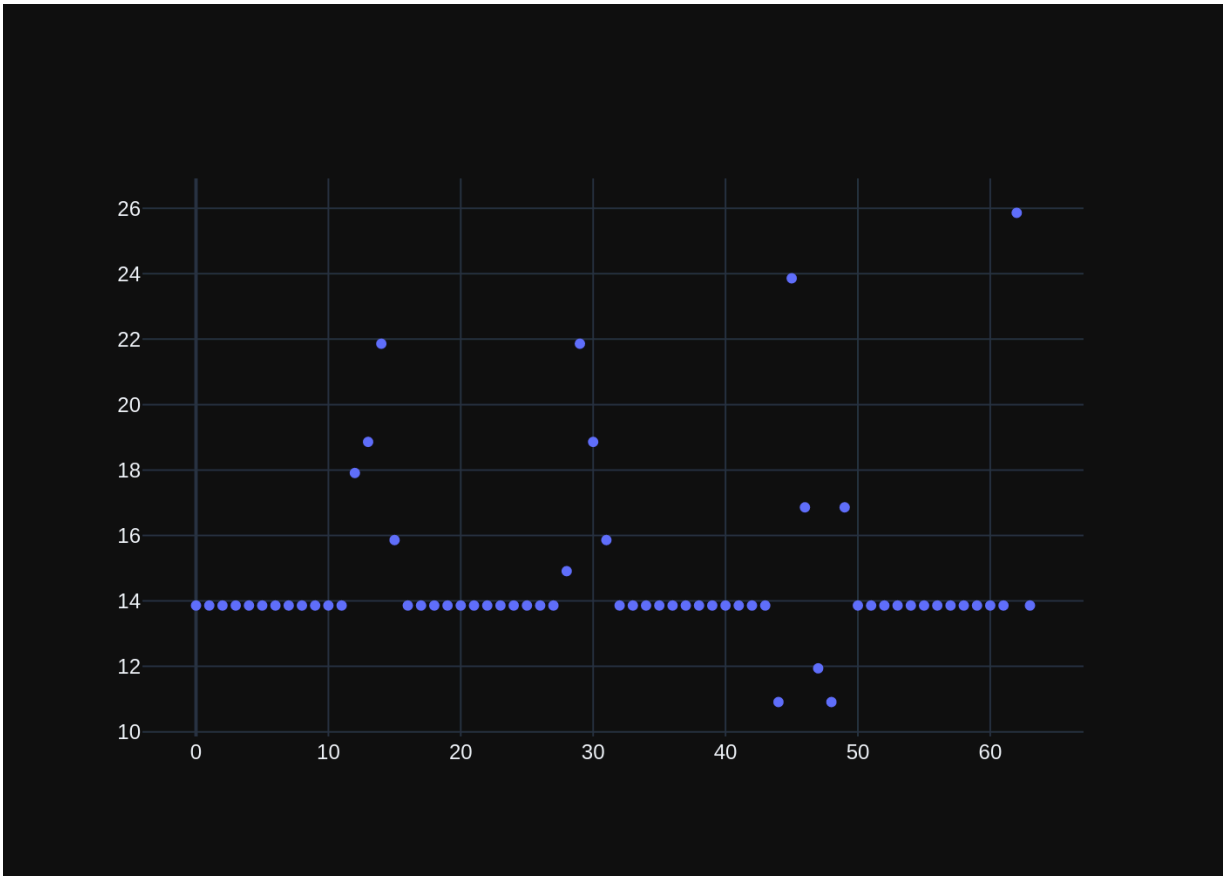
umber of invoices for each customer\n'}, {'role': 'assistant', 'content': 'S
ELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvo
ices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId
\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   Find the tot
al number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SE
LECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY Bill
ingCountry'}, {'role': 'user', 'content': 'How many customers are there'},
{'role': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers FROM
customers'}, {'role': 'user', 'content': ' \n   List all albums and their
corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT
a.Title AS AlbumTitle, 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': 'assis
tant', 'content': "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%wha
t%'"}, {'role': 'user', 'content': 'Can you list all tables in the SQLite da
tabase catalog?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite
_master WHERE type='table'"}, {'role': 'user', 'content': ' \n   List all
invoices with a total exceeding $10:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NU
LL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NU
LL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERE
NCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON D
ELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvo
iceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n
InvoiceId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTE
GER NOT NULL,\r\n   InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress
NVARCHAR(70),\r\n   BillingCity NVARCHAR(40),\r\n   BillingState NVARCHAR
(40),\r\n   BillingCountry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR
(10),\r\n   Total NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId)
REFERENCES \"customers\" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\"
(TrackId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)
\n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGE
R,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Comp
oser NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTE
GER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId)
REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFER
ENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsT
o)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r\n   Last
Name NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   Address NVA
RCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Count

```
ry NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    Sup
portRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES \"employees
\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREA
TE TABLE \"employees\"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARC
HAR(20)  NOT NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n
BirthDate DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r
\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR
(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax
NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFE
RENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO A
CTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (Suppor
tRepId)\n\n\n===Additional Context \n\nIn the chinook database invoice means
order\n\n===Response Guidelines \n1. If the provided context is sufficient,
please generate a valid SQL query without any explanations for the question.
\n2. If the provided context is almost sufficient but requires knowledge of
a specific string in a particular column, please generate an intermediate SQ
L query to find the distinct strings in that column. Prepend the query with
a comment saying intermediate_sql \n3. If the provided context is insufficie
nt, please explain why it can't be generated. \n4. Please use the most relev
ant table(s). \n5. If the question has been asked and answered before, pleas
e repeat the answer exactly as it was given before. \n"}, {"role": "user",
"content": "  \n    Get the total number of invoices for each customer\n"},
{"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastNa
me, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoice
s i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "use
r", "content": "  \n    Find the total number of invoices per country:\n"},
{"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) AS TotalIn
voices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "conten
t": "How many customers are there"}, {"role": "assistant", "content": "SELEC
T COUNT(*) AS NumberOfCustomers FROM customers"}, {"role": "user", "conten
t": "  \n    List all albums and their corresponding artist names  \n"}, {"r
ole": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS Arti
stName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"rol
e": "user", "content": "  \n    Find all tracks with a name containing \"Wha
t\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \nFRO
M tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "C
an you list all tables in the SQLite database catalog?"}, {"role": "assistan
t", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"rol
e": "user", "content": "  \n    List all invoices with a total exceeding $1
0:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:16:28.861
322909Z', 'message': {'role': 'assistant', 'content': 'SELECT * \nFROM invoi
ces \nWHERE Total > 10'}, 'done_reason': 'stop', 'done': True, 'total_durati
on': 40816682939, 'load_duration': 686440, 'prompt_eval_count': 1424, 'promp
t_eval_duration': 39257957000, 'eval_count': 15, 'eval_duration': 112330900
0}
LLM Response: SELECT *
FROM invoices
WHERE Total > 10
SELECT *
FROM invoices
WHERE Total > 10
    InvoiceId  CustomerId          InvoiceDate             BillingAddress  \
```

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

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

[64 rows x 9 columns]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all invoices with a total exceeding $10:\n'\n\nThe DataFrame was
produced using this query: SELECT * \nFROM invoices \nWHERE Total > 10\n\nTh
e following is information about the resulting pandas DataFrame 'df': \nRunn
ing df.dtypes gives:\n InvoiceId              int64\nCustomerId
int64\nInvoiceDate           object\nBillingAddress        object\nBillingCi
ty           object\nBillingState          object\nBillingCountry        obj
ect\nBillingPostalCode     object\nTotal                float64\ndtype: obje
ct"}, {"role": "user", "content": "Can you generate the Python plotly code t
o chart the results of the dataframe? Assume the data is in a pandas datafra
me called 'df'. If there is only one value in the dataframe, use an Indicato
r. Respond with only Python code. Do not answer with any explanations -- jus
t the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:16:41.957
344809Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\nif df.shape[0] == 1:\n    f
ig = go.Figure(go.Indicator(\n        mode="number",\n        value=df[\'Tot
al\'].values[0],\n        title={"text": "Invoice Total"}\n    ))\nelse:\n
fig = go.Figure()\n    fig.add_trace(go.Scatter(x=df.index, y=df[\'Total\'],
mode=\'markers\', name=\'Invoice Totals\'))\n\nfig.show()\n```'}, 'done_reas
on': 'stop', 'done': True, 'total_duration': 13073817851, 'load_duration': 4
0924004, 'prompt_eval_count': 213, 'prompt_eval_duration': 4290135000, 'eval
_count': 129, 'eval_duration': 8689332000}
```


26
24
22
20
18
16
14
12
10
0
10
20
30
40
50
60

Out[24]:
```
('SELECT * \nFROM invoices \nWHERE Total > 10',
    InvoiceId  CustomerId          InvoiceDate             BillingAddress
\
 0           5          23  2009-01-11 00:00:00            69 Salem Street
 1          12           2  2009-02-11 00:00:00    Theodor-Heuss-Straße 34
 2          19          40  2009-03-14 00:00:00             8, Rue Hanovre
 3          26          19  2009-04-14 00:00:00            1 Infinite Loop
 4          33          57  2009-05-15 00:00:00            Calle Lira, 198
 ..        ...         ...                  ...                        ...
 59        383          10  2013-08-12 00:00:00  Rua Dr. Falcão Filho, 155
 60        390          48  2013-09-12 00:00:00      Lijnbaansgracht 120bg
 61        397          27  2013-10-13 00:00:00            1033 N Park Ave
 62        404           6  2013-11-13 00:00:00              Rilská 3174/6
 63        411          44  2013-12-14 00:00:00            Porthaninkatu 9

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

 [64 rows x 9 columns],
 Figure({
     'data': [{'mode': 'markers',
               'name': 'Invoice Totals',
               'type': 'scatter',
               'x': array([ 0,  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, 60, 61, 62, 63]),
              'y': array([13.86, 13.86, 13.86, 13.86, 13.86, 13.86, 13.86,
13.86, 13.86, 13.86,
                          13.86, 13.86, 17.91, 18.86, 21.86, 15.86, 13.86,
13.86, 13.86, 13.86,
                          13.86, 13.86, 13.86, 13.86, 13.86, 13.86, 13.86,
13.86, 14.91, 21.86,
                          18.86, 15.86, 13.86, 13.86, 13.86, 13.86, 13.86,
13.86, 13.86, 13.86,
                          13.86, 13.86, 13.86, 13.86, 10.91, 23.86, 16.86,
11.94, 10.91, 16.86,
                          13.86, 13.86, 13.86, 13.86, 13.86, 13.86, 13.86,
13.86, 13.86, 13.86,
                          13.86, 13.86, 25.86, 13.86])}],
     'layout': {'template': '...'}
 }))
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId I
NTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity
INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (Inv
oiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items"
(InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREA
TE TABLE "employees"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMEN
T 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    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r
\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR
(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax
NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFE
RENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n
LastName NVARCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address
NVARCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Co
untry NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    Sup
portRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees"
(EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    Media
TypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(2
20),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    Uni
tPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "alb
ums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREI
GN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_type
s" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCRE
ATE 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 NUL
L,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "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 DELE
TE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "t
racks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n=
==Additional Context \n\nIn the chinook database invoice means order\n\n===R
esponse Guidelines \n1. If the provided context is sufficient, please genera
te a valid SQL query without any explanations for the question. \n2. If the
```

provided context is almost sufficient but requires knowledge of a specific s
tring in a particular column, please generate an intermediate SQL query to f
ind the distinct strings in that column. Prepend the query with a comment sa
ying intermediate_sql \n3. If the provided context is insufficient, please e
xplain why it can\'t be generated. \n4. Please use the most relevant table
(s). \n5. If the question has been asked and answered before, please repeat
the answer exactly as it was given before. \n'}, {'role': 'user', 'content':
' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assist
ant', 'content': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'u
ser', 'content': ' \n   Find the total number of invoices per countr
y:\n'}, {'role': 'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS
TotalInvoices\nFROM invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'c
ontent': ' \n   Get the total number of invoices for each customer\n'},
{'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastNa
me, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoice
s i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'use
r', 'content': 'How many customers are there'}, {'role': 'assistant', 'conte
nt': 'SELECT COUNT(*) AS NumberOfCustomers FROM customers'}, {'role': 'use
r', 'content': ' \n   List all albums and their corresponding artist names
\n'}, {'role': 'assistant', 'content': 'SELECT a.Title AS AlbumTitle, ar.Nam
e AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistI
d'}, {'role': 'user', 'content': ' \n   Find all tracks with a name contai
ning "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT
* \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'conte
nt': 'Can you list all tables in the SQLite database catalog?'}, {'role': 'a
ssistant', 'content': "SELECT name FROM sqlite_master WHERE type='table'"},
{'role': 'user', 'content': ' \n   Find all invoices since 2010 and the to
tal amount invoiced:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (I
nvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCR
EATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   FirstName NVA
RCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r
\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address NVARCHAR

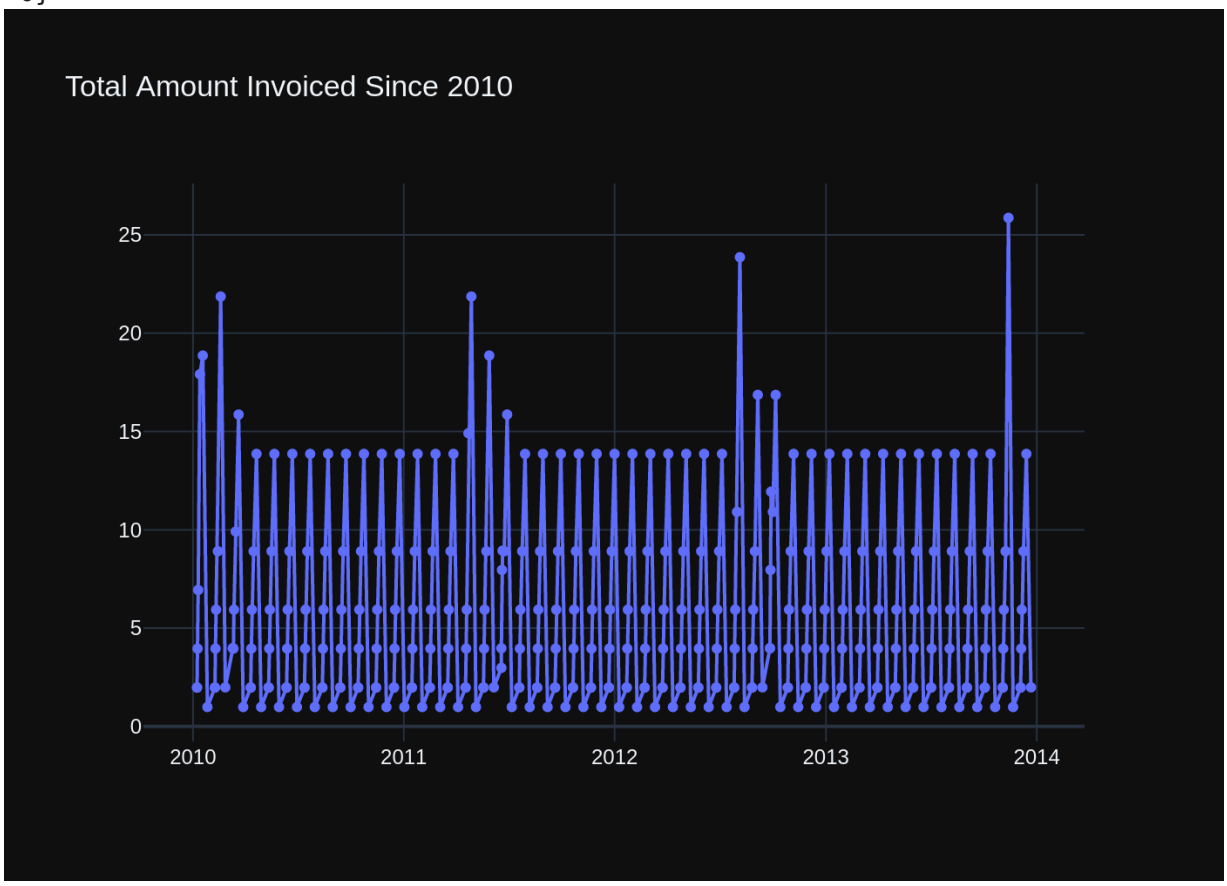
(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Country NV
ARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo)
REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NUL
L,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n
Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r
\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n
SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n
MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARC
HAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOI
NCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INT
EGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (Artist
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"p
laylist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INT
EGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId,
TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (Playlist
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (Tra
ckId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoic
e means order\n\n===Response Guidelines \n1. If the provided context is suff
icient, please generate a valid SQL query without any explanations for the q
uestion. \n2. If the provided context is almost sufficient but requires know
ledge of a specific string in a particular column, please generate an interm
ediate SQL query to find the distinct strings in that column. Prepend the qu
ery with a comment saying intermediate_sql \n3. If the provided context is i
nsufficient, please explain why it can't be generated. \n4. Please use the m
ost relevant table(s). \n5. If the question has been asked and answered befo
re, please repeat the answer exactly as it was given before. \n"}, {"role":
"user", "content": " \n   List all invoices with a total exceeding $1
0:\n"}, {"role": "assistant", "content": "SELECT * \nFROM invoices \nWHERE T
otal > 10"}, {"role": "user", "content": " \n   Find the total number of i
nvoices per country:\n"}, {"role": "assistant", "content": "SELECT BillingCo
untry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"},
{"role": "user", "content": " \n   Get the total number of invoices for ea
ch customer\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.Fi
rstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\n
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerI
d"}, {"role": "user", "content": "How many customers are there"}, {"role":
"assistant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customer
s"}, {"role": "user", "content": " \n   List all albums and their correspo
nding artist names \n"}, {"role": "assistant", "content": "SELECT a.Title A
S AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.Art
istId = ar.ArtistId"}, {"role": "user", "content": " \n   Find all tracks
with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistan

t", "content": "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:17:30.617310696Z', 'message': {'role': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, 'done_reason': 'stop', 'done': True, 'total_duration': 48572004727, 'load_duration': 1261013, 'prompt_eval_count': 1623, 'prompt_eval_duration': 46054505000, 'eval_count': 26, 'eval_duration': 2063678000}
LLM Response: SELECT InvoiceDate, Total
FROM invoices
WHERE InvoiceDate >= '2010-01-01'
SELECT InvoiceDate, Total
FROM invoices
WHERE InvoiceDate >= '2010-01-01'

	InvoiceDate	Total
0	2010-01-08 00:00:00	1.98
1	2010-01-08 00:00:00	1.98
2	2010-01-09 00:00:00	3.96
3	2010-01-10 00:00:00	6.94
4	2010-01-13 00:00:00	17.91
..	...	...
324	2013-12-05 00:00:00	3.96
325	2013-12-06 00:00:00	5.94
326	2013-12-09 00:00:00	8.91
327	2013-12-14 00:00:00	13.86
328	2013-12-22 00:00:00	1.99

[329 rows x 2 columns]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that contains the results of the query that answers the question the user asked: '\n   Find all invoices since 2010 and the total amount invoiced:\n'\n\nThe DataFrame was produced using this query: SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n InvoiceDate   object\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 with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:17:44.86358168Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport plotly.graph_objects as go\nimport pandas as pd\n\n# Assuming df is your DataFrame\nif len(df) == 1:\n   fig = go.Figure(go.Indicator(\n   mode="number",\n   value=df[\'Total\'].values[0],\n   title={"text": "Total Amount Invoiced"}\n   ))\nelse:\n   fig = go.Figure()\n   fig.add_trace(go.Scatter(x=df[\'InvoiceDate\'], y=df[\'Total\'], mode=\'lines+markers\', na

me=\'Invoices\'))\n\nfig.update_layout(title="Total Amount Invoiced Since 2010")\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 14218400802, 'load_duration': 43333352, 'prompt_eval_count': 183, 'prompt_eval_duration': 3508297000, 'eval_count': 160, 'eval_duration': 10568744000}



Out[25]:
```
("SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-0
1'",
              InvoiceDate  Total
 0    2010-01-08 00:00:00   1.98
 1    2010-01-08 00:00:00   1.98
 2    2010-01-09 00:00:00   3.96
 3    2010-01-10 00:00:00   6.94
 4    2010-01-13 00:00:00  17.91
 ..                   ...    ...
 324  2013-12-05 00:00:00   3.96
 325  2013-12-06 00:00:00   5.94
 326  2013-12-09 00:00:00   8.91
 327  2013-12-14 00:00:00  13.86
 328  2013-12-22 00:00:00   1.99

 [329 rows x 2 columns],
 Figure({
     'data': [{'mode': 'lines+markers',
               'name': 'Invoices',
               'type': 'scatter',
               'x': array(['2010-01-08 00:00:00', '2010-01-08 00:00:00', '2
010-01-09 00:00:00',
                           ..., '2013-12-09 00:00:00', '2013-12-14 00:00:0
0',
                           '2013-12-22 00:00:00'], dtype=object),
               'y': array([ 1.98,  1.98,  3.96, ...,  8.91, 13.86,  1.9
9])}],
     'layout': {'template': '...', 'title': {'text': 'Total Amount Invoiced
Since 2010'}}
 }))
```

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

vn.ask(question=question)
```

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

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

be generated. \n4. Please use the most relevant table(s). \n5. If the questi
on has been asked and answered before, please repeat the answer exactly as i
t was given before. \n'}, {'role': 'user', 'content': ' \n   Get the total
number of invoices for each customer\n'}, {'role': 'assistant', 'content':
'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalIn
voices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerI
d\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   Find all in
voices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant',
'content': "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >=
'2010-01-01'"}, {'role': 'user', 'content': ' \n   Find the total number o
f invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT Billin
gCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountr
y'}, {'role': 'user', 'content': ' \n   List all albums and their correspo
nding artist names \n'}, {'role': 'assistant', 'content': 'SELECT a.Title A
S AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.Art
istId = ar.ArtistId'}, {'role': 'user', 'content': ' \n   List all invoice
s with a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT
* \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'content': 'How man
y customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) A
S NumberOfCustomers FROM customers'}, {'role': 'user', 'content': 'Can you l
ist all tables in the SQLite database catalog?'}, {'role': 'assistant', 'con
tent': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'use
r', 'content': ' \n   Find all tracks with a name containing "What" (case-
insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \nFROM tracks
\nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': " \n   Li
st all employees and their reporting manager's name (if any):\n"}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (Repo
rtsTo)\n\nCREATE TABLE \"employees\"\r\n(\r\n   EmployeeId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   F
irstName NVARCHAR(20) NOT NULL,\r\n   Title NVARCHAR(30),\r\n   ReportsTo
INTEGER,\r\n   BirthDate DATETIME,\r\n   HireDate DATETIME,\r\n   Address
NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r\n   Co
untry NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVARCHAR(2
4),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60),\r\n   FOREIGN KEY
(ReportsTo) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"customers\"\r\n(\r\n   Custome
rId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(4
0) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHA
R(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State N
VARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r
\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60)
NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REF
ERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (Suppo
rtRepId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   Invo
iceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   Billi
ngCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry

NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(1
0,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (Cu
stomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TAB
LE \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCRE
MENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER N
OT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER
NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY
KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE
\"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r
\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaType
Id INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(22
0),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   Unit
Price NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"alb
ums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FORE
IGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_t
ypes\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER
NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE sqlite_st
at1(tbl,idx,stat)\n\n\n===Additional Context \n\nIn the chinook database inv
oice means order\n\n===Response Guidelines \n1. If the provided context is s
ufficient, please generate a valid SQL query without any explanations for th
e question. \n2. If the provided context is almost sufficient but requires k
nowledge of a specific string in a particular column, please generate an int
ermediate SQL query to find the distinct strings in that column. Prepend the
query with a comment saying intermediate_sql \n3. If the provided context is
insufficient, please explain why it can't be generated. \n4. Please use the
most relevant table(s). \n5. If the question has been asked and answered bef
ore, please repeat the answer exactly as it was given before. \n"}, {"role":
"user", "content": " \n   Get the total number of invoices for each custom
er\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName,
c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN
invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role":
"user", "content": " \n   Find all invoices since 2010 and the total amoun
t invoiced:\n"}, {"role": "assistant", "content": "SELECT InvoiceDate, Total
\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "cont
ent": " \n   Find the total number of invoices per country:\n"}, {"role":
"assistant", "content": "SELECT BillingCountry, COUNT(*) AS TotalInvoices\nF
ROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "content": " \n
List all albums and their corresponding artist names \n"}, {"role": "assist
ant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM
albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "c
ontent": " \n   List all invoices with a total exceeding $10:\n"}, {"rol
e": "assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"},
{"role": "user", "content": "How many customers are there"}, {"role": "assis
tant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customers"},
{"role": "user", "content": "Can you list all tables in the SQLite database
catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master
WHERE type='table'"}, {"role": "user", "content": " \n   Find all tracks w
ith a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant",
"content": "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {"ro

le": "user", "content": " \n   List all employees and their reporting mana
ger's name (if any):\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:18:35.509
064023Z', 'message': {'role': 'assistant', 'content': "SELECT e1.FirstName |
| ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName || ' '
|| e2.LastName, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JOIN e
mployees e2 ON e1.ReportsTo = e2.EmployeeId"}, 'done_reason': 'stop', 'don
e': True, 'total_duration': 50546402406, 'load_duration': 1315399, 'prompt_e
val_count': 1586, 'prompt_eval_duration': 44723870000, 'eval_count': 66, 'ev
al_duration': 5321352000}
LLM Response: SELECT e1.FirstName || ' ' || e1.LastName AS EmployeeName,
       COALESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS Manager
Name
FROM employees e1
LEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId
SELECT e1.FirstName || ' ' || e1.LastName AS EmployeeName,
       COALESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS Manager
Name
FROM employees e1
LEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId
       EmployeeName   ManagerName
0   Andrew Adams   No Manager
1   Nancy Edwards   Andrew Adams
2   Jane Peacock   Nancy Edwards
3   Margaret Park   Nancy Edwards
4   Steve Johnson   Nancy Edwards
5 Michael Mitchell   Andrew Adams
6   Robert King Michael Mitchell
7   Laura Callahan Michael Mitchell
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n   List all employees and their reporting manager's name (if any):\n'\n\n
The DataFrame was produced using this query: SELECT e1.FirstName || ' ' || e
1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName || ' ' || e2.Last
Name, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JOIN employees e
2 ON e1.ReportsTo = e2.EmployeeId\n\nThe following is information about the
resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n EmployeeName
object\nManagerName   object\ndtype: object"}, {"role": "user", "content":
"Can you generate the Python plotly code to chart the results of the datafra
me? Assume the data is in a pandas dataframe called 'df'. If there is only o
ne value in the dataframe, use an Indicator. Respond with only Python code.
Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:18:58.700
625787Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\n# Check if DataFrame has on
ly one row\nif df.shape[0] == 1:\n   fig = go.Figure(go.Indicator(\n
mode="number",\n   value=df[\'EmployeeName\'].iloc[0],\n   title=
{"text": "Employee and Manager"}\n   ))\nelse:\n   fig = go.Figure(data=[g
o.Sankey(\n   node=dict(\n   pad=15,\n   thickness=2

0,\n   line=dict(color="black", width=0.5),\n   label=df
[\'EmployeeName\'].tolist(),\n   color=["#FFD700"]*len(df) # Chang
e colors if needed\n   ),\n   link=dict(\n   source=[i fo
r i in range(len(df))],\n   target=[i+1 if i % 2 == 0 else i-1 for
i in range(len(df))],\n   value=df[\'ManagerName\'].tolist()\n
)\n   )])\n\nfig.update_layout(title_text="Employee Reporting Structure", f
ont_size=10)\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total
_duration': 23166174929, 'load_duration': 41920416, 'prompt_eval_count': 22
0, 'prompt_eval_duration': 4282375000, 'eval_count': 277, 'eval_duration': 1
8789741000}

Out[26]:
```
("SELECT e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n       COALE
SCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM
employees e1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId",
        EmployeeName       ManagerName
 0      Andrew Adams        No Manager
 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': [{'link': {'source': [0, 1, 2, 3, 4, 5, 6, 7],
                        'target': [1, 0, 3, 2, 5, 4, 7, 6],
                        'value': [No Manager, Andrew Adams, Nancy Edwards,
Nancy
                                  Edwards, Nancy Edwards, Andrew Adams, Mic
hael
                                  Mitchell, Michael Mitchell]},
               'node': {'color': [#FFD700, #FFD700, #FFD700, #FFD700, #FFD7
00,
                                  #FFD700, #FFD700, #FFD700],
                        'label': [Andrew Adams, Nancy Edwards, Jane Peacoc
k,
                                  Margaret Park, Steve Johnson, Michael Mit
chell,
                                  Robert King, Laura Callahan],
                        'line': {'color': 'black', 'width': 0.5},
                        'pad': 15,
                        'thickness': 20},
               'type': 'sankey'}],
     'layout': {'font': {'size': 10}, 'template': '...', 'title': {'text':
'Employee Reporting Structure'}}
 }))
```

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
```

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

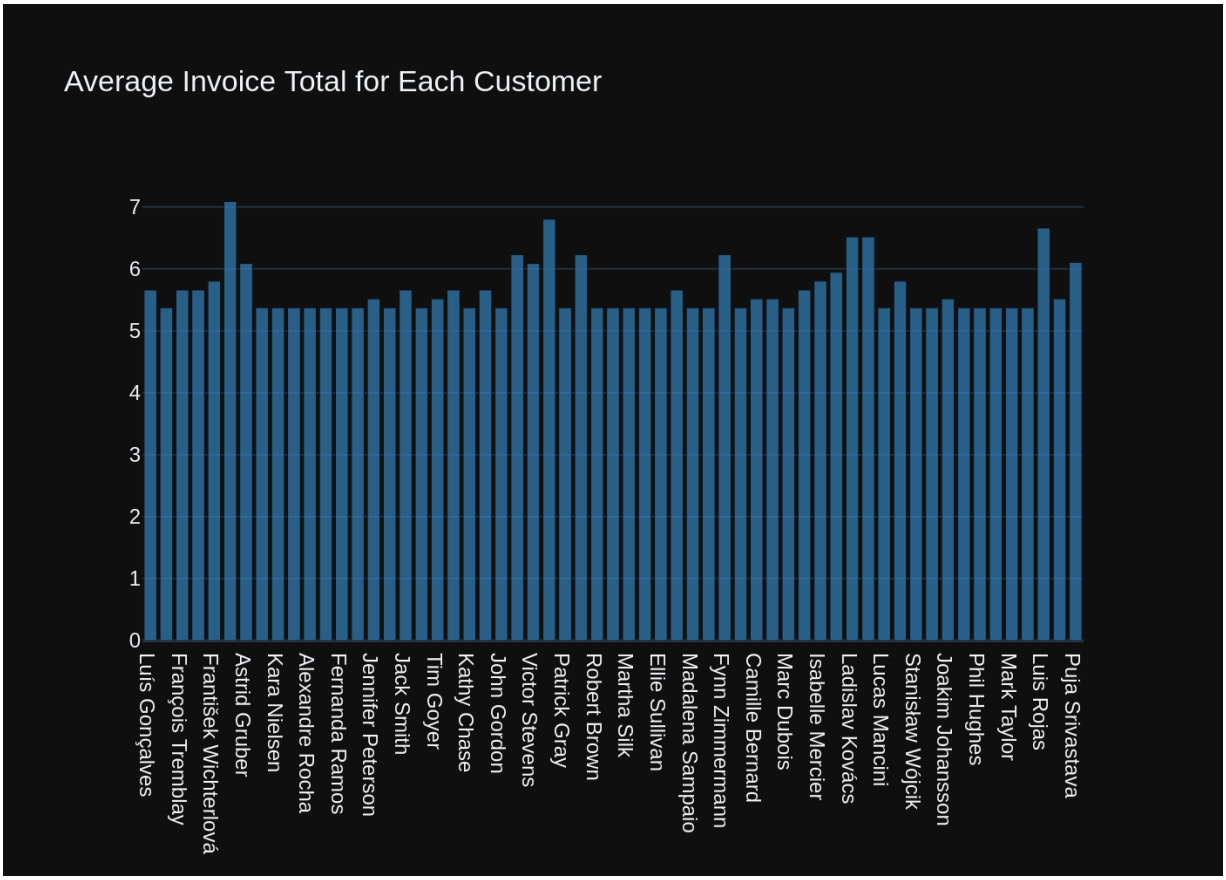
\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'cont
ent': ' \n   List all invoices with a total exceeding $10:\n'}, {'role':
'assistant', 'content': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'ro
le': 'user', 'content': 'How many customers are there'}, {'role': 'assistan
t', 'content': 'SELECT COUNT(*) AS NumberOfCustomers FROM customers'}, {'rol
e': 'user', 'content': " \n   List all employees and their reporting manag
er's name (if any):\n"}, {'role': 'assistant', 'content': "SELECT e1.FirstNa
me || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName ||
' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JO
IN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {'role': 'user', 'conten
t': ' \n   Find all tracks with a name containing "What" (case-insensitiv
e)\n'}, {'role': 'assistant', 'content': "SELECT * \nFROM tracks \nWHERE LOW
ER(Name) LIKE '%what%'"}, {'role': 'user', 'content': ' \n   List all albu
ms and their corresponding artist names \n'}, {'role': 'assistant', 'conten
t': 'SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOI
N 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': 'u
ser', 'content': ' \n   Get the average invoice total for each custome
r:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK
_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"inv
oice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NUL
L,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NU
LL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n
\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) RE
FERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N\r\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)
\n\nCREATE TABLE sqlite_stat1(tbl,idx,stat)\n\nCREATE INDEX IFK_CustomerSupp
ortRepId ON \"customers\" (SupportRepId)\n\nCREATE TABLE \"customers\"\r\n
(\r\n   CustomerId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   First
Name NVARCHAR(40) NOT NULL,\r\n   LastName NVARCHAR(20) NOT NULL,\r\n
Company NVARCHAR(80),\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(4
0),\r\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCod
e NVARCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   E
mail NVARCHAR(60) NOT NULL,\r\n   SupportRepId INTEGER,\r\n   FOREIGN KEY
(SupportRepId) REFERENCES \"employees\" (EmployeeId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"emp
loyees\" (ReportsTo)\n\nCREATE TABLE \"employees\"\r\n(\r\n   EmployeeId IN
TEGER 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 DATETIM
E,\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),\r\n
FOREIGN KEY (ReportsTo) REFERENCES \"employees\" (EmployeeId) \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 p
rovided context is sufficient, please generate a valid SQL query without any
explanations for the question. \n2. If the provided context is almost suffic
ient but requires knowledge of a specific string in a particular column, ple
ase generate an intermediate SQL query to find the distinct strings in that
column. Prepend the query with a comment saying intermediate_sql \n3. If the
provided context is insufficient, please explain why it can't be generated.
\n4. Please use the most relevant table(s). \n5. If the question has been as
ked and answered before, please repeat the answer exactly as it was given be
fore. \n"}, {"role": "user", "content": " \n   Get the total number of inv
oices for each customer\n"}, {"role": "assistant", "content": "SELECT c.Cust
omerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM c
ustomers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.
CustomerId"}, {"role": "user", "content": " \n   Find the total number of
invoices per country:\n"}, {"role": "assistant", "content": "SELECT BillingC
ountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"},
{"role": "user", "content": " \n   Find all invoices since 2010 and the to
tal amount invoiced:\n"}, {"role": "assistant", "content": "SELECT InvoiceDa
te, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "use
r", "content": " \n   List all invoices with a total exceeding $10:\n"},
{"role": "assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 1
0"}, {"role": "user", "content": "How many customers are there"}, {"role":
"assistant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customer
s"}, {"role": "user", "content": " \n   List all employees and their repor
ting manager's name (if any):\n"}, {"role": "assistant", "content": "SELECT
e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.Fir
stName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e
1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {"role": "use
r", "content": " \n   Find all tracks with a name containing \"What\" (cas
e-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \nFROM tracks
\nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": " \n   Li
st all albums and their corresponding artist names \n"}, {"role": "assistan
t", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM al
bums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "con
tent": "Can you list all tables in the SQLite database catalog?"}, {"role":
"assistant", "content": "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {"role": "user", "content": " \n   Get the average invoice total for
each customer:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:19:41.564
936713Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId,
c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers
c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerI
d'}, 'done_reason': 'stop', 'done': True, 'total_duration': 42779545500, 'lo
ad_duration': 751703, 'prompt_eval_count': 1408, 'prompt_eval_duration': 381
60083000, 'eval_count': 51, 'eval_duration': 4019549000}
LLM Response: SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS
AverageInvoiceTotal
FROM customers c

```
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoice
Total
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
```

	CustomerId	FirstName	LastName	AverageInvoiceTotal
0	1	Luís	Gonçalves	5.660000
1	2	Leonie	Köhler	5.374286
2	3	François	Tremblay	5.660000
3	4	Bjørn	Hansen	5.660000
4	5	František	Wichterlová	5.802857
5	6	Helena	Holý	7.088571
6	7	Astrid	Gruber	6.088571
7	8	Daan	Peeters	5.374286
8	9	Kara	Nielsen	5.374286
9	10	Eduardo	Martins	5.374286
10	11	Alexandre	Rocha	5.374286
11	12	Roberto	Almeida	5.374286
12	13	Fernanda	Ramos	5.374286
13	14	Mark	Philips	5.374286
14	15	Jennifer	Peterson	5.517143
15	16	Frank	Harris	5.374286
16	17	Jack	Smith	5.660000
17	18	Michelle	Brooks	5.374286
18	19	Tim	Goyer	5.517143
19	20	Dan	Miller	5.660000
20	21	Kathy	Chase	5.374286
21	22	Heather	Leacock	5.660000
22	23	John	Gordon	5.374286
23	24	Frank	Ralston	6.231429
24	25	Victor	Stevens	6.088571
25	26	Richard	Cunningham	6.802857
26	27	Patrick	Gray	5.374286
27	28	Julia	Barnett	6.231429
28	29	Robert	Brown	5.374286
29	30	Edward	Francis	5.374286
30	31	Martha	Silk	5.374286
31	32	Aaron	Mitchell	5.374286
32	33	Ellie	Sullivan	5.374286
33	34	João	Fernandes	5.660000
34	35	Madalena	Sampaio	5.374286
35	36	Hannah	Schneider	5.374286
36	37	Fynn	Zimmermann	6.231429
37	38	Niklas	Schröder	5.374286
38	39	Camille	Bernard	5.517143
39	40	Dominique	Lefebvre	5.517143
40	41	Marc	Dubois	5.374286
41	42	Wyatt	Girard	5.660000
42	43	Isabelle	Mercier	5.802857
43	44	Terhi	Hämäläinen	5.945714
44	45	Ladislav	Kovács	6.517143
45	46	Hugh	O'Reilly	6.517143
46	47	Lucas	Mancini	5.374286
47	48	Johannes	Van der Berg	5.802857

```
48          49  Stanisław        Wójcik            5.374286
49          50    Enrique         Muñoz            5.374286
50          51     Joakim     Johansson            5.517143
51          52       Emma         Jones            5.374286
52          53       Phil        Hughes            5.374286
53          54      Steve        Murray            5.374286
54          55       Mark        Taylor            5.374286
55          56      Diego     Gutiérrez            5.374286
56          57       Luis         Rojas            6.660000
57          58      Manoj        Pareek            5.517143
58          59       Puja    Srivastava            6.106667
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Get the average invoice total for each customer:\n'\n\nThe DataFrame w
as produced using this query: SELECT c.CustomerId, c.FirstName, c.LastName,
AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i
ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\n\nThe following is in
formation about the resulting pandas DataFrame 'df': \nRunning df.dtypes giv
es:\n CustomerId               int64\nFirstName              object\nLastNa
me               object\nAverageInvoiceTotal    float64\ndtype: object"},
{"role": "user", "content": "Can you generate the Python plotly code to char
t the results of the dataframe? Assume the data is in a pandas dataframe cal
led 'df'. If there is only one value in the dataframe, use an Indicator. Res
pond with only Python code. Do not answer with any explanations -- just the
code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:19:58.231
429029Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\n\nif len(df) == 1:\n    fig = go.Figure(go.Indicat
or(\n        mode="number",\n        value=df[\'AverageInvoiceTotal\'].value
s[0],\n        title={"text": "Average Invoice Total"},\n        domain={\'x
\': [0, 1], \'y\': [0, 1]}\n    ))\nelse:\n    fig = go.Figure(go.Bar(\n
x=df[\'FirstName\'] + \' \' + df[\'LastName\'],\n        y=df[\'AverageInvoi
ceTotal\'],\n        marker_color=\'rgba(55, 128, 191, 0.7)\'\n    ))\n\nfi
g.update_layout(title="Average Invoice Total for Each Customer")\nfig.show()
\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 16638337049,
'load_duration': 43402289, 'prompt_eval_count': 213, 'prompt_eval_duration':
4147758000, 'eval_count': 185, 'eval_duration': 12402378000}
```


Average Invoice Total for Each Customer
7
6
5
4
3
2
1
0
Luís Gonçalves
François Tremblay
František Wichterlová
Astrid Gruber
Kara Nielsen
Alexandre Rocha
Fernanda Ramos
Jennifer Peterson
Jack Smith
Tim Goyer
Kathy Chase
John Gordon
Victor Stevens
Patrick Gray
Robert Brown
Martha Silk
Ellie Sullivan
Madalena Sampaio
Fynn Zimmermann
Camille Bernard
Marc Dubois
Isabelle Mercier
Ladislav Kovács
Lucas Mancini
Stanisław Wójcik
Joakim Johansson
Phil Hughes
Mark Taylor
Luis Rojas
Puja Srivastava

Out[27]: ('SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId',

	CustomerId	FirstName	LastName	AverageInvoiceTotal
0	1	Luís	Gonçalves	5.660000
1	2	Leonie	Köhler	5.374286
2	3	François	Tremblay	5.660000
3	4	Bjørn	Hansen	5.660000
4	5	František	Wichterlová	5.802857
5	6	Helena	Holý	7.088571
6	7	Astrid	Gruber	6.088571
7	8	Daan	Peeters	5.374286
8	9	Kara	Nielsen	5.374286
9	10	Eduardo	Martins	5.374286
10	11	Alexandre	Rocha	5.374286
11	12	Roberto	Almeida	5.374286
12	13	Fernanda	Ramos	5.374286
13	14	Mark	Philips	5.374286
14	15	Jennifer	Peterson	5.517143
15	16	Frank	Harris	5.374286
16	17	Jack	Smith	5.660000
17	18	Michelle	Brooks	5.374286
18	19	Tim	Goyer	5.517143
19	20	Dan	Miller	5.660000
20	21	Kathy	Chase	5.374286
21	22	Heather	Leacock	5.660000
22	23	John	Gordon	5.374286
23	24	Frank	Ralston	6.231429
24	25	Victor	Stevens	6.088571
25	26	Richard	Cunningham	6.802857
26	27	Patrick	Gray	5.374286
27	28	Julia	Barnett	6.231429
28	29	Robert	Brown	5.374286
29	30	Edward	Francis	5.374286
30	31	Martha	Silk	5.374286
31	32	Aaron	Mitchell	5.374286
32	33	Ellie	Sullivan	5.374286
33	34	João	Fernandes	5.660000
34	35	Madalena	Sampaio	5.374286
35	36	Hannah	Schneider	5.374286
36	37	Fynn	Zimmermann	6.231429
37	38	Niklas	Schröder	5.374286
38	39	Camille	Bernard	5.517143
39	40	Dominique	Lefebvre	5.517143
40	41	Marc	Dubois	5.374286
41	42	Wyatt	Girard	5.660000
42	43	Isabelle	Mercier	5.802857
43	44	Terhi	Hämäläinen	5.945714
44	45	Ladislav	Kovács	6.517143
45	46	Hugh	O'Reilly	6.517143
46	47	Lucas	Mancini	5.374286
47	48	Johannes	Van der Berg	5.802857
48	49	Stanisław	Wójcik	5.374286
49	50	Enrique	Muñoz	5.374286
50	51	Joakim	Johansson	5.517143
51	52	Emma	Jones	5.374286

```
 52          53       Phil        Hughes             5.374286
 53          54      Steve        Murray             5.374286
 54          55       Mark        Taylor             5.374286
 55          56      Diego     Gutiérrez             5.374286
 56          57       Luis         Rojas             6.660000
 57          58      Manoj        Pareek             5.517143
 58          59       Puja    Srivastava             6.106667,
 Figure({
     'data': [{'marker': {'color': 'rgba(55, 128, 191, 0.7)'},
               'type': 'bar',
               'x': array(['Luís Gonçalves', 'Leonie Köhler', 'François Tre
mblay', 'Bjørn Hansen',
                           'František Wichterlová', 'Helena Holý', 'Astrid
Gruber', 'Daan Peeters',
                           'Kara Nielsen', 'Eduardo Martins', 'Alexandre Ro
cha', 'Roberto Almeida',
                           'Fernanda Ramos', 'Mark Philips', 'Jennifer Pete
rson', 'Frank Harris',
                           'Jack Smith', 'Michelle Brooks', 'Tim Goyer', 'D
an Miller',
                           'Kathy Chase', 'Heather Leacock', 'John Gordon',
'Frank Ralston',
                           'Victor Stevens', 'Richard Cunningham', 'Patrick
Gray', 'Julia Barnett',
                           'Robert Brown', 'Edward Francis', 'Martha Silk',
'Aaron Mitchell',
                           'Ellie Sullivan', 'João Fernandes', 'Madalena Sa
mpaio',
                           'Hannah Schneider', 'Fynn Zimmermann', 'Niklas S
chröder',
                           'Camille Bernard', 'Dominique Lefebvre', 'Marc D
ubois', 'Wyatt Girard',
                           'Isabelle Mercier', 'Terhi Hämäläinen', 'Ladisla
v Kovács',
                           "Hugh O'Reilly", 'Lucas Mancini', 'Johannes Van
der Berg',
                           'Stanisław Wójcik', 'Enrique Muñoz', 'Joakim Joh
ansson', 'Emma Jones',
                           'Phil Hughes', 'Steve Murray', 'Mark Taylor', 'D
iego Gutiérrez',
                           'Luis Rojas', 'Manoj Pareek', 'Puja Srivastav
a'], dtype=object),
               '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])}],
     'layout': {'template': '...', 'title': {'text': 'Average Invoice Total
for Each Customer'}}
 }))
```

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

vn.ask(question=question)
```

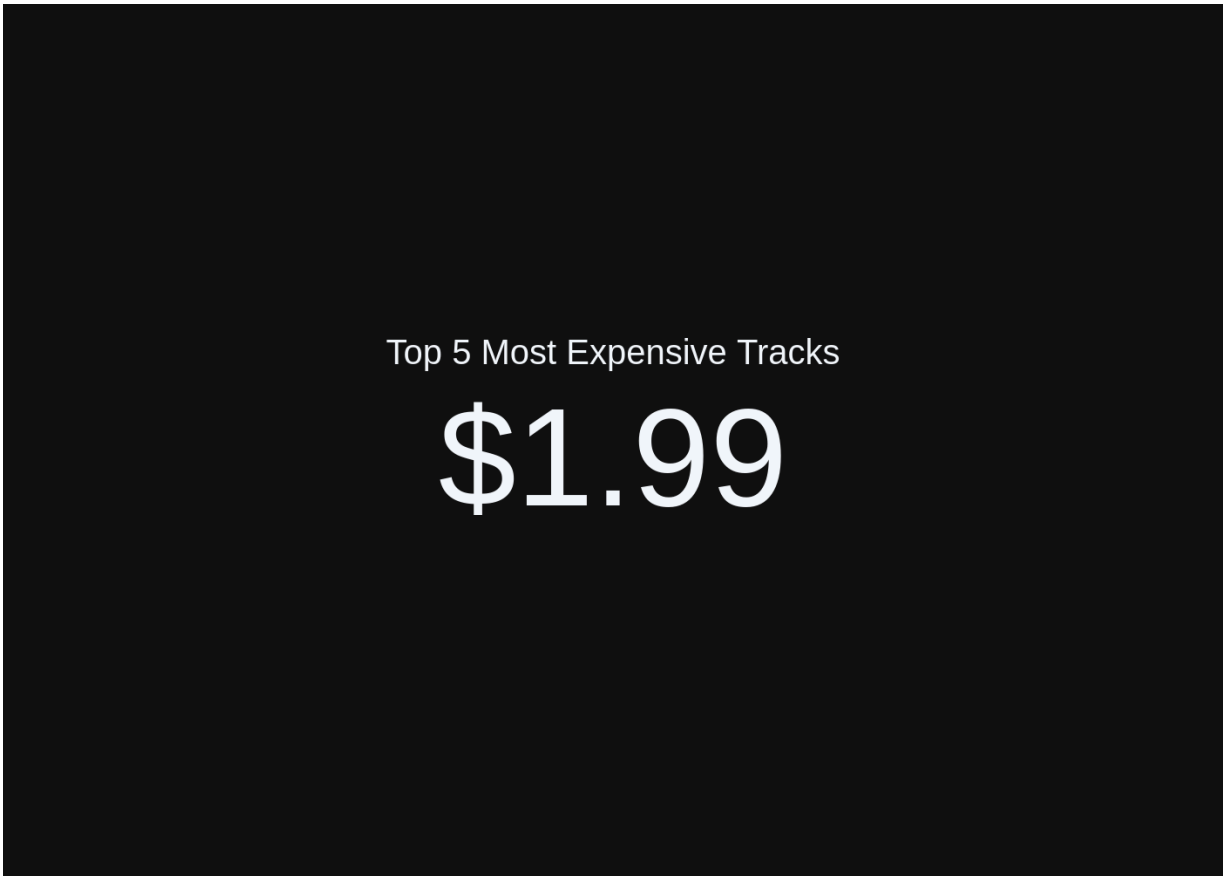
```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

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

FROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   Find the total number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry'}, {'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(*) AS NumberOfCustomers FROM customers'}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.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 e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE INDEX IFK_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 \"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_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\" (Artis

tId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": " \n   Find all tracks with a name containing \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELECT * \nFROM tracks \nWHERE LOWER(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"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n   Get the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId \nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"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(*) AS NumberOfCustomers FROM customers"}, {"role": "user", "content": " \n   Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.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 e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:20:37.036324167Z', 'message': {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5'}, 'done_reason': 'stop', 'done': True, 'total_duration': 38695223123, 'load_duration': 693876, 'prompt_eval_count': 1370, 'prompt_eval_duration': 36676827000, 'eval_count': 19, 'eval_duration': 1429292000}
LLM Response: SELECT Name, UnitPrice
FROM tracks
ORDER BY UnitPrice DESC
LIMIT 5
SELECT Name, UnitPrice
FROM tracks

```
ORDER BY UnitPrice DESC
LIMIT 5
                                     Name  UnitPrice
0  Battlestar Galactica: The Story So Far       1.99
1                  Occupation / Precipice       1.99
2                           Exodus, Pt. 1       1.99
3                           Exodus, Pt. 2       1.99
4                           Collaborators       1.99
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Find the top 5 most expensive tracks (based on unit price):\n'\n\nThe
DataFrame was produced using this query: SELECT Name, UnitPrice\nFROM tracks
\nORDER BY UnitPrice DESC\nLIMIT 5\n\nThe following is information about the
resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n Name
object\nUnitPrice    float64\ndtype: object"}, {"role": "user", "content":
"Can you generate the Python plotly code to chart the results of the datafra
me? Assume the data is in a pandas dataframe called 'df'. If there is only o
ne value in the dataframe, use an Indicator. Respond with only Python code.
Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:20:52.288
207764Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport numpy as np\n\nif df[\'UnitPrice\'].nunique
() == 1:\n    fig = go.Figure(go.Indicator(\n        mode="number",\n
value=df[\'UnitPrice\'].iloc[0],\n        title={"text": "Top 5 Most Expensi
ve Tracks"},\n        number={\'prefix\': "$"}\n    ))\nelse:\n    fig = go.
Figure()\n    fig.add_trace(go.Bar(x=df[\'Name\'], y=df[\'UnitPrice\'], text
=df[\'UnitPrice\']))\n    fig.update_layout(title="Top 5 Most Expensive Trac
ks", xaxis_tickangle=-45, yaxis_title="Unit Price")\n\nfig.show()\n```'}, 'd
one_reason': 'stop', 'done': True, 'total_duration': 15226288837, 'load_dura
tion': 42562317, 'prompt_eval_count': 174, 'prompt_eval_duration': 343293300
0, 'eval_count': 177, 'eval_duration': 11705649000}
```



Out[28]:
```
('SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5',
                                        Name  UnitPrice
 0  Battlestar Galactica: The Story So Far       1.99
 1                 Occupation / Precipice       1.99
 2                          Exodus, Pt. 1       1.99
 3                          Exodus, Pt. 2       1.99
 4                          Collaborators       1.99,
 Figure({
     'data': [{'mode': 'number',
               'number': {'prefix': '$'},
               'title': {'text': 'Top 5 Most Expensive Tracks'},
               'type': 'indicator',
               'value': 1.99}],
     'layout': {'template': '...'}
 }))
```

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

vn.ask(question=question)
```

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

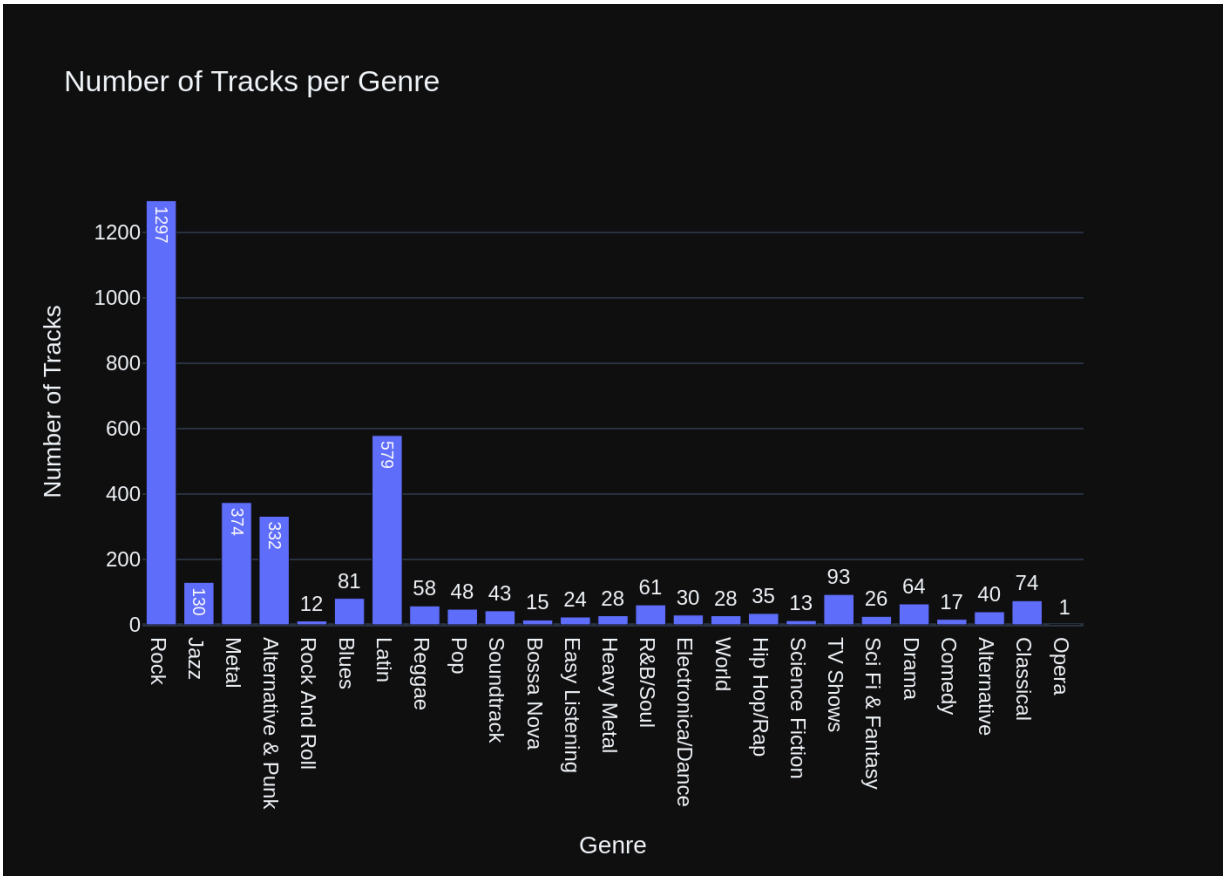
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER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

\nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   Get the total number of invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   Get the average invoice total for each customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   List all genres and the number of tracks in each genre:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION \r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (ArtistId)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer

exactly as it was given before. \n"}, {"role": "user", "content": " \n   F
ind the top 5 most expensive tracks (based on unit price):\n"}, {"role": "as
sistant", "content": "SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPric
e DESC\nLIMIT 5"}, {"role": "user", "content": " \n   List all albums and
their corresponding artist names \n"}, {"role": "assistant", "content": "SE
LECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artis
ts ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Fi
nd all tracks with a name containing \"What\" (case-insensitive)\n"}, {"rol
e": "assistant", "content": "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE
'%what%'"}, {"role": "user", "content": "How many customers are there"}, {"r
ole": "assistant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM cus
tomers"}, {"role": "user", "content": " \n   Find the total number of invo
ices per country:\n"}, {"role": "assistant", "content": "SELECT BillingCount
ry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"r
ole": "user", "content": "Can you list all tables in the SQLite database cat
alog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WH
ERE type='table'"}, {"role": "user", "content": " \n   List all invoices w
ith a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT *
\nFROM invoices \nWHERE Total > 10"}, {"role": "user", "content": " \n   F
ind all invoices since 2010 and the total amount invoiced:\n"}, {"role": "as
sistant", "content": "SELECT InvoiceDate, Total\nFROM invoices\nWHERE Invoic
eDate >= '2010-01-01'"}, {"role": "user", "content": " \n   Get the total
number of invoices for each customer\n"}, {"role": "assistant", "content":
"SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalIn
voices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerI
d\nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   Get the ave
rage invoice total for each customer:\n"}, {"role": "assistant", "content":
"SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoic
eTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerI
d\nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   List all ge
nres and the number of tracks in each genre:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:21:29.022
605843Z', 'message': {'role': 'assistant', 'content': 'SELECT g.Name AS Genr
e, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON
g.GenreId = t.GenreId\nGROUP BY g.GenreId'}, 'done_reason': 'stop', 'done':
True, 'total_duration': 36639595134, 'load_duration': 943088, 'prompt_eval_c
ount': 1239, 'prompt_eval_duration': 32511817000, 'eval_count': 46, 'eval_du
ration': 3525325000}
LLM Response: SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
GROUP BY g.GenreId
SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
GROUP BY g.GenreId

	Genre	NumberOfTracks
0	Rock	1297
1	Jazz	130
2	Metal	374
3	Alternative & Punk	332
4	Rock And Roll	12
5	Blues	81
6	Latin	579
7	Reggae	58

```
8                  Pop              48
9            Soundtrack              43
10           Bossa Nova              15
11      Easy Listening               24
12         Heavy Metal              28
13            R&B/Soul              61
14   Electronica/Dance              30
15                World              28
16         Hip Hop/Rap              35
17     Science Fiction              13
18            TV Shows              93
19    Sci Fi & Fantasy              26
20                Drama              64
21               Comedy              17
22         Alternative               40
23           Classical              74
24                Opera               1
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    List all genres and the number of tracks in each genre:\n'\n\nThe Data
Frame was produced using this query: SELECT g.Name AS Genre, COUNT(t.TrackI
d) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.Gen
reId\nGROUP BY g.GenreId\n\nThe following is information about the resulting
pandas DataFrame 'df': \nRunning df.dtypes gives:\n Genre             object
\nNumberOfTracks     int64\ndtype: object"}, {"role": "user", "content": "Ca
n 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 v
alue in the dataframe, use an Indicator. Respond with only Python code. Do n
ot answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:21:41.219
195974Z', 'message': {'role': 'assistant', 'content': " ```python\nimport pl
otly.express as px\n\nif len(df) == 1:\n    fig = px.indicator(value=df['Num
berOfTracks'].iloc[0], title='Number of Tracks per Genre')\nelse:\n    fig =
px.bar(df, x='Genre', y='NumberOfTracks', text='NumberOfTracks', title='Numb
er of Tracks per Genre')\nfig.update_xaxes(title_text='Genre')\nfig.update_y
axes(title_text='Number of Tracks')\nfig.show()\n```"}, 'done_reason': 'sto
p', 'done': True, 'total_duration': 12175899137, 'load_duration': 41168161,
'prompt_eval_count': 200, 'prompt_eval_duration': 3952918000, 'eval_count':
124, 'eval_duration': 8136918000}
```


Number of Tracks per Genre
Number of Tracks
0
200
400
600
800
1000
1200
1297
130
374
332
12
81
579
58
48
43
15
24
28
61
30
28
35
13
93
26
64
17
40
74
1
Rock
Jazz
Metal
Alternative & Punk
Rock And Roll
Blues
Latin
Reggae
Pop
Soundtrack
Bossa Nova
Easy Listening
Heavy Metal
R&B/Soul
Electronica/Dance
World
Hip Hop/Rap
Science Fiction
TV Shows
Sci Fi & Fantasy
Drama
Comedy
Alternative
Classical
Opera
Genre

Out[29]:
```
('SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g
\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId',
                  Genre  NumberOfTracks
 0                 Rock            1297
 1                 Jazz             130
 2                Metal             374
 3   Alternative & Punk             332
 4        Rock And Roll              12
 5                Blues              81
 6                Latin             579
 7               Reggae              58
 8                  Pop              48
 9           Soundtrack              43
 10          Bossa Nova              15
 11      Easy Listening              24
 12         Heavy Metal              28
 13            R&B/Soul              61
 14   Electronica/Dance              30
 15               World              28
 16         Hip Hop/Rap              35
 17     Science Fiction              13
 18            TV Shows              93
 19    Sci Fi & Fantasy              26
 20               Drama              64
 21              Comedy              17
 22         Alternative              40
 23           Classical              74
 24               Opera               1,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'Genre=%{x}<br>NumberOfTracks=%{text}<extra
></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'text': array([1.297e+03, 1.300e+02, 3.740e+02, 3.320e+02,
1.200e+01, 8.100e+01,
                              5.790e+02, 5.800e+01, 4.800e+01, 4.300e+01,
1.500e+01, 2.400e+01,
                              2.800e+01, 6.100e+01, 3.000e+01, 2.800e+01,
3.500e+01, 1.300e+01,
                              9.300e+01, 2.600e+01, 6.400e+01, 1.700e+01,
4.000e+01, 7.400e+01,
                              1.000e+00]),
               'textposition': 'auto',
               'type': 'bar',
               'x': array(['Rock', 'Jazz', 'Metal', 'Alternative & Punk',
'Rock And Roll', 'Blues',
                           'Latin', 'Reggae', 'Pop', 'Soundtrack', 'Bossa N
ova', 'Easy Listening',
                           'Heavy Metal', 'R&B/Soul', 'Electronica/Dance',
'World', 'Hip Hop/Rap',
                           'Science Fiction', 'TV Shows', 'Sci Fi & Fantas
```

```
y', 'Drama', 'Comedy',
                           'Alternative', 'Classical', 'Opera'], dtype=obje
ct),
               'xaxis': 'x',
               'y': array([1297,  130,  374,  332,   12,   81,  579,   58,
48,   43,   15,   24,
                           28,   61,   30,   28,   35,   13,   93,   26,
64,   17,   40,   74,
                            1]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Number of Tracks per Genre'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Genre'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Number of Tracks'}}}
 }))
```

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

vn.ask(question=question)
```

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

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON "playlist_track" (TrackId)\n\nCREATE INDEX IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "genres"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name 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_track"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_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\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   List all genres and the number of tracks in each genre:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId'}, {'role': 'user', 'content': ' \n   Find all tracks with a name containing "What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT a.Title AS AlbumTitle, 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': ' \n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'conten

t': " \n   List all employees and their reporting manager's name (if an
y):\n"}, {'role': 'assistant', 'content': "SELECT e1.FirstName || ' ' || e1.
LastName AS EmployeeName, \n   COALESCE(e2.FirstName || ' ' || e2.LastNa
me, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JOIN employees e2
ON e1.ReportsTo = e2.EmployeeId"}, {'role': 'user', 'content': ' \n   List
all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'conten
t': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'conten
t': 'How many customers are there'}, {'role': 'assistant', 'content': 'SELEC
T COUNT(*) AS NumberOfCustomers FROM customers'}, {'role': 'user', 'conten
t': ' \n   Find the total number of invoices per country:\n'}, {'role': 'a
ssistant', 'content': 'SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFRO
M invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'content': ' \n
Get all genres that do not have any tracks associated with them:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\n
CREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n
MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARC
HAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES
\"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n
FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO A
CTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"med
ia_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)
\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE
INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"genres\"\r
\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name N
VARCHAR(120)\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRI
MARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n
ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artis
ts\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREA
TE TABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n
TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY
(PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlist
s\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOR
EIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\"
(ArtistId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER PRIM
ARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Addit
ional Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can't be generated. \n4. Please use the most relevant table(s). \n5. If
the question has been asked and answered before, please repeat the answer ex

actly as it was given before. \n"}, {"role": "user", "content": " \n   Lis
t all genres and the number of tracks in each genre:\n"}, {"role": "assistan
t", "content": "SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\n
FROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.Genre
Id"}, {"role": "user", "content": " \n   Find all tracks with a name conta
ining \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SEL
ECT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "c
ontent": " \n   Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nFROM trac
ks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": " \n
List all albums and their corresponding artist names \n"}, {"role": "assist
ant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM
albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "c
ontent": "Can you list all tables in the SQLite database catalog?"}, {"rol
e": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {"role": "user", "content": " \n   Find all invoices since 2010 and
the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT Inv
oiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"rol
e": "user", "content": " \n   List all employees and their reporting manag
er's name (if any):\n"}, {"role": "assistant", "content": "SELECT e1.FirstNa
me || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName ||
' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JO
IN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {"role": "user", "conten
t": " \n   List all invoices with a total exceeding $10:\n"}, {"role": "as
sistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"}, {"rol
e": "user", "content": "How many customers are there"}, {"role": "assistan
t", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customers"}, {"rol
e": "user", "content": " \n   Find the total number of invoices per countr
y:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) AS
TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "c
ontent": " \n   Get all genres that do not have any tracks associated with
them:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:22:17.969
236637Z', 'message': {'role': 'assistant', 'content': 'SELECT g.Name \nFROM
genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nWHERE t.TrackId IS NU
LL'}, 'done_reason': 'stop', 'done': True, 'total_duration': 36605633223, 'l
oad_duration': 622788, 'prompt_eval_count': 1257, 'prompt_eval_duration': 33
454543000, 'eval_count': 33, 'eval_duration': 2557209000}
LLM Response: SELECT g.Name
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
WHERE t.TrackId IS NULL
SELECT g.Name
FROM genres g
LEFT JOIN tracks t ON g.GenreId = t.GenreId
WHERE t.TrackId IS NULL
Empty DataFrame
Columns: [Name]
Index: []
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con

tains the results of the query that answers the question the user asked: '
\n   Get all genres that do not have any tracks associated with them:\n'\n
\nThe DataFrame was produced using this query: SELECT g.Name \nFROM genres g
\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nWHERE t.TrackId IS NULL\n\nTh
e following is information about the resulting pandas DataFrame 'df': \nRunn
ing df.dtypes gives:\n Name   object\ndtype: object"}, {"role": "user", "co
ntent": "Can you generate the Python plotly code to chart the results of the
dataframe? Assume the data is in a pandas dataframe called 'df'. If there is
only one value in the dataframe, use an Indicator. Respond with only Python
code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:22:34.478
354465Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport numpy as np\n\nif df[\'Name\'].nunique() ==
1:\n   fig = go.Figure(go.Indicator(\n   mode="number",\n   value
=df[\'Name\'].iloc[0],\n   title={"text": "Genre"},\n   domain=
{\'x\': [0, 1], \'y\': [0, 1]}\n   ))\nelse:\n   fig = go.Figure(go.Bar(x=
df[\'Name\'], y=[1]*len(df), text=["Track" if np.isnan(t) else "No Track" fo
r t in df[\'TrackId\']], textposition=\'auto\'))\n\nfig.update_layout(title_
text="Genres with No Associated Tracks")\nfig.show()\n```'}, 'done_reason':
'stop', 'done': True, 'total_duration': 16506764584, 'load_duration': 409047
11, 'prompt_eval_count': 180, 'prompt_eval_duration': 3795556000, 'eval_coun
t': 183, 'eval_duration': 12625778000}

Out[30]:
```
('SELECT g.Name \nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreI
d\nWHERE t.TrackId IS NULL',
 Empty DataFrame
 Columns: [Name]
 Index: [],
 Figure({
     'data': [{'domain': {'x': [0.0, 1.0], 'y': [0.0, 1.0]},
               'hovertemplate': 'Name=%{label}<extra></extra>',
               'labels': array([], dtype=object),
               'legendgroup': '',
               'name': '',
               'showlegend': True,
               'type': 'pie'}],
     'layout': {'legend': {'tracegroupgap': 0}, 'margin': {'t': 60}, 'templ
ate': '...'}
 }))
```

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

vn.ask(question=question)
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOIN
CREMENT 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(40),\r\n    State NVARCHAR(40),\r\n    Country NV
ARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "inv
oice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NUL
L,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NU
LL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFE
RENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE "employees"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY A
UTOINCREMENT NOT NULL,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstN
ame 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 NVA
RCHAR(70),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Count
ry NVARCHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(2
4),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY
(ReportsTo) REFERENCES "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION
ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "playlist_track"\r\n(\r\n    Playli
stId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT
PK_PlaylistTrack PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (Pla
ylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION 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 "album
s"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Ti
tle NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREI
GN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_CustomerSupportRepId ON "cust
omers" (SupportRepId)\n\nCREATE TABLE "playlists"\r\n(\r\n    PlaylistId INT
EGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\nC
REATE TABLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT N
OT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n
MediaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARC
HAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n
UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES
"albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    F
OREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_t
ypes" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n
CREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\n\n===Additi
onal Context \n\nIn the chinook database invoice means order\n\n===Response
Guidelines \n1. If the provided context is sufficient, please generate a val
```

```
id SQL query without any explanations for the question. \n2. If the provided
context is almost sufficient but requires knowledge of a specific string in
a particular column, please generate an intermediate SQL query to find the d
istinct strings in that column. Prepend the query with a comment saying inte
rmediate_sql \n3. If the provided context is insufficient, please explain wh
y it can\'t be generated. \n4. Please use the most relevant table(s). \n5. I
f the question has been asked and answered before, please repeat the answer
exactly as it was given before. \n'}, {'role': 'user', 'content': '  \n    G
et the total number of invoices for each customer\n'}, {'role': 'assistant',
'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId)
AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId =
i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': 'How many
customers are there'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) AS
NumberOfCustomers FROM customers'}, {'role': 'user', 'content': '  \n    Get
the average invoice total for each customer:\n'}, {'role': 'assistant', 'con
tent': 'SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS Averag
eInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.Cu
stomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': '  \n    List
all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'conten
t': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'conten
t': '  \n    Find all invoices since 2010 and the total amount invoice
d:\n'}, {'role': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM in
voices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': "
\n    List all employees and their reporting manager's name (if any):\n"},
{'role': 'assistant', 'content': "SELECT e1.FirstName || ' ' || e1.LastName
AS EmployeeName, \n       COALESCE(e2.FirstName || ' ' || e2.LastName, 'No M
anager') AS ManagerName\nFROM employees e1\nLEFT JOIN employees e2 ON e1.Rep
ortsTo = e2.EmployeeId"}, {'role': 'user', 'content': '  \n    Find the tota
l number of invoices per country:\n'}, {'role': 'assistant', 'content': 'SEL
ECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY Billi
ngCountry'}, {'role': 'user', 'content': '  \n    List all albums and their
corresponding artist names  \n'}, {'role': 'assistant', 'content': 'SELECT
a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar
ON a.ArtistId = ar.ArtistId'}, {'role': 'user', 'content': '  \n    Find the
top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistan
t', 'content': 'SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC
\nLIMIT 5'}, {'role': 'user', 'content': '  \n    List all genres and the nu
mber of tracks in each genre:\n'}, {'role': 'assistant', 'content': 'SELECT
g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOI
N tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId'}, {'role': 'user',
'content': '  \n    List all customers who have not placed any orders:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n
InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    B
illingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
```

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

c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customers"}, {"role": "user", "content": " \n   Get the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": " \n   List all employees and their reporting manager's name (if any):\n"}, {"role": "assistant", "content": "SELECT e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {"role": "user", "content": " \n   Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": " \n   List all genres and the number of tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId"}, {"role": "user", "content": " \n   List all customers who have not placed any orders:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:23:34.139675275Z', 'message': {'role': 'assistant', 'content': 'SELECT c.*\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nWHERE i.InvoiceId IS NULL'}, 'done_reason': 'stop', 'done': True, 'total_duration': 59575050458, 'load_duration': 909297, 'prompt_eval_count': 1880, 'prompt_eval_duration': 56407542000, 'eval_count': 31, 'eval_duration': 2574578000}
LLM Response: SELECT c.*
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
WHERE i.InvoiceId IS NULL
SELECT c.*
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
WHERE i.InvoiceId IS NULL
Empty DataFrame
Columns: [CustomerId, FirstName, LastName, Company, Address, City, State, Country, PostalCode, Phone, Fax, Email, SupportRepId]
Index: []
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:

[{"role": "system", "content": "The following is a pandas DataFrame that contains the results of the query that answers the question the user asked: '\n   List all customers who have not placed any orders:\n'\n\nThe DataFrame was produced using this query: SELECT c.*\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nWHERE i.InvoiceId IS NULL\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId   object\nFirstName   object\nLastName   object\nCompany   object\nAddress   object\nCity   object\nState   object\nCountry   object\nPostalCode   object\nPhone   object\nFax   object\nEmail   object\nSupportRepId   object\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the results of the dataframe? Assume the data is in a pandas dataframe called 'df'. If there is only one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:23:47.59250323Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport plotly.express as px\nimport pandas as pd\n\n# Assuming df is already defined and contains the data from the query\nif len(df) == 1:\n   fig = px.indicator(value=len(df), title="Number of Customers with No Orders")\nelse:\n   fig = px.bar(df, x=\'CustomerId\', y=\'FirstName\', text_auto=\'.2s\', title="Customers without Orders")\n\nfig.update_layout(xaxis_title="Customer ID", yaxis_title="First Name")\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 13450133893, 'load_duration': 830203, 'prompt_eval_count': 229, 'prompt_eval_duration': 4584831000, 'eval_count': 131, 'eval_duration': 8730070000}

Out[31]:
```
('SELECT c.*\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.Cu
stomerId\nWHERE i.InvoiceId IS NULL',
 Empty DataFrame
 Columns: [CustomerId, FirstName, LastName, Company, Address, City, State,
Country, PostalCode, Phone, Fax, Email, SupportRepId]
 Index: [],
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>FirstName=%{y}<extra></
extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'textposition': 'auto',
               'texttemplate': '%{y:.2s}',
               'type': 'bar',
               'x': array([], dtype=object),
               'xaxis': 'x',
               'y': array([], dtype=object),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Customers without Orders'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Customer ID'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'First Name'}}}
 }))
```

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

vn.ask(question=question)
```

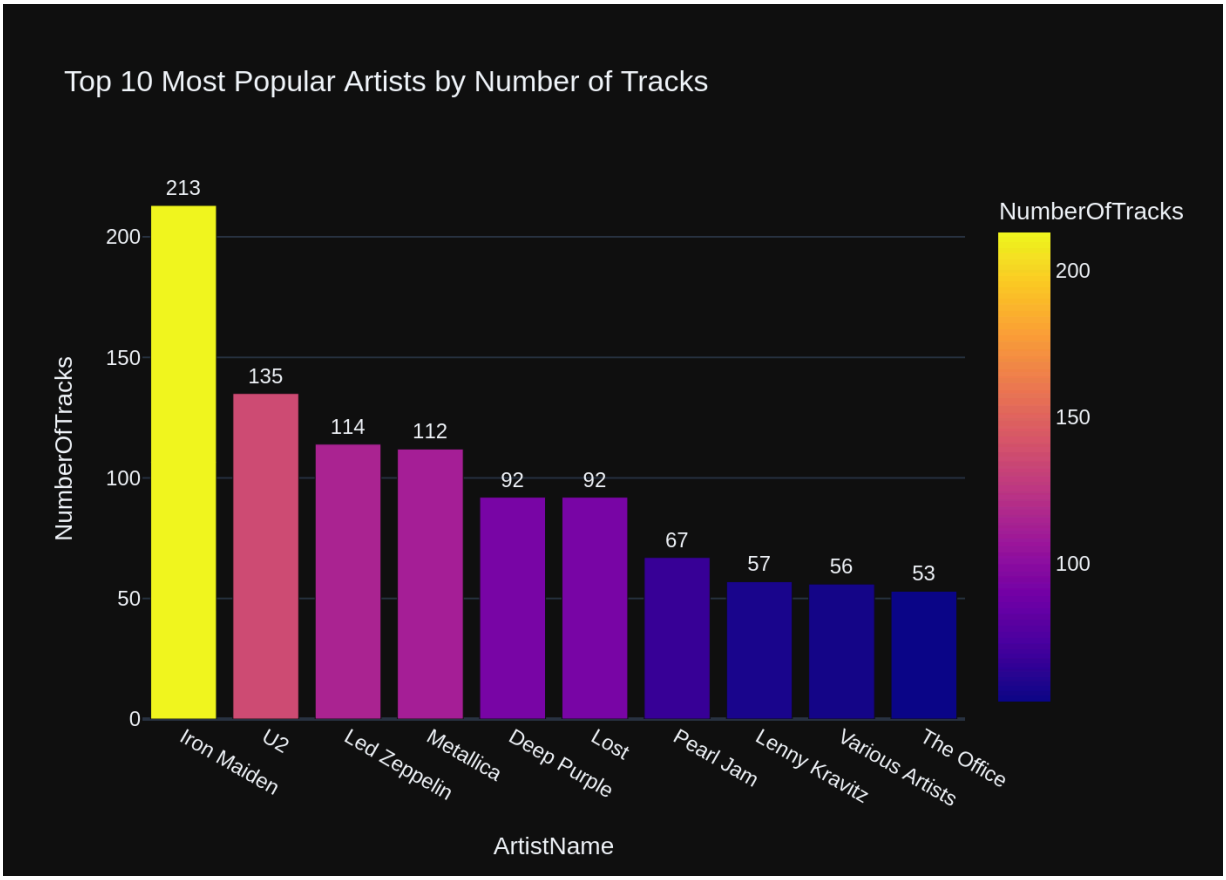
```
Number of requested results 10 is greater than number of elements in index
1, updating n_results = 1
```

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

OUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i O
N c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'co
ntent': ' \n   Get the average invoice total for each customer:\n'}, {'rol
e': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, A
VG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i O
N c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'co
ntent': 'How many customers are there'}, {'role': 'assistant', 'content': 'S
ELECT COUNT(*) AS NumberOfCustomers FROM customers'}, {'role': 'user', 'cont
ent': ' \n   Find the total number of invoices per country:\n'}, {'role':
'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS TotalInvoices\nF
ROM invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'content': ' \n
List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'co
ntent': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'co
ntent': ' \n   There are 3 tables: artists, albums and tracks, where albums
and artists are linked by ArtistId, albums and tracks are linked by AlbumI
d,\n   Can you find the top 10 most popular artists based on the number of
tracks\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEG
ER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NUL
L,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENC
ES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUT
OINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK_Albu
mArtistId ON \"albums\" (ArtistId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tra
cks\" (AlbumId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE
TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT N
ULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n
(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n
CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREI
GN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE
NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tra
cks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREA
TE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE INDEX IFK_Playli
stTrackTrackId ON \"playlist_track\" (TrackId)\n\n\n===Additional Context \n
\nIn the chinook database invoice means order\n\n===Response Guidelines \n1.
If the provided context is sufficient, please generate a valid SQL query wit
hout any explanations for the question. \n2. If the provided context is almo
st sufficient but requires knowledge of a specific string in a particular co

lumn, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n 3. If the provided context is insufficient, please explain why it can't be g enerated. \n4. Please use the most relevant table(s). \n5. If the question h as been asked and answered before, please repeat the answer exactly as it wa s given before. \n"}, {"role": "user", "content": " \n   List all genres a nd the number of tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\n LEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on u nit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nF ROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"rol e": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS Artist Name\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 tr acks \nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": "Can y ou list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "u ser", "content": " \n   Get the total number of invoices for each customer \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c. LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN i nvoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   Get the average invoice total for each custome r:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "How many customers are there"}, {"role": "assistant", "c ontent": "SELECT COUNT(*) AS NumberOfCustomers FROM customers"}, {"role": "u ser", "content": " \n   Find the total number of invoices per countr y:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "c ontent": " \n   List all invoices with a total exceeding $10:\n"}, {"rol e": "assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"}, {"role": "user", "content": " \n   There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks a re linked by AlbumId,\n   Can you find the top 10 most popular artists base d on the number of tracks\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:24:29.651 738503Z', 'message': {'role': 'assistant', 'content': 'SELECT ar.Name AS Art istName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albu ms al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.Alb umId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 10'}, 'done_ reason': 'stop', 'done': True, 'total_duration': 41974756392, 'load_duratio n': 722717, 'prompt_eval_count': 1312, 'prompt_eval_duration': 35605533000, 'eval_count': 73, 'eval_duration': 5696890000}
LLM Response: SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTrac ks
FROM artists ar
LEFT JOIN albums al ON ar.ArtistId = al.ArtistId
LEFT JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY ar.ArtistId
ORDER BY NumberOfTracks DESC
LIMIT 10

```
SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks
FROM artists ar
LEFT JOIN albums al ON ar.ArtistId = al.ArtistId
LEFT JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY ar.ArtistId
ORDER BY NumberOfTracks DESC
LIMIT 10
        ArtistName  NumberOfTracks
0      Iron Maiden             213
1               U2             135
2     Led Zeppelin             114
3        Metallica             112
4      Deep Purple              92
5             Lost              92
6        Pearl Jam              67
7    Lenny Kravitz              57
8  Various Artists              56
9       The Office              53
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    There are 3 tables: artists, albums and tracks, where albums and artis
ts are linked by ArtistId, albums and tracks are linked by AlbumId,\n    Can
you find the top 10 most popular artists based on the number of tracks\n'\n
\nThe DataFrame was produced using this query: SELECT ar.Name AS ArtistName,
COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON
ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGRO
UP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 10\n\nThe following i
s information about the resulting pandas DataFrame 'df': \nRunning df.dtypes
gives:\n ArtistName        object\nNumberOfTracks     int64\ndtype: objec
t"}, {"role": "user", "content": "Can you generate the Python plotly code to
chart the results of the dataframe? Assume the data is in a pandas dataframe
called 'df'. If there is only one value in the dataframe, use an Indicator.
Respond with only Python code. Do not answer with any explanations -- just t
he code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:24:44.343
458481Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.express as px\n\nif len(df) == 1:\n    fig = px.indicator(value=df[\'Nu
mberOfTracks\'].iloc[0], title="Top Artist Popularity", template="simple_whi
te")\nelse:\n    fig = px.bar(df, x=\'ArtistName\', y=\'NumberOfTracks\', te
xt=\'NumberOfTracks\', color=\'NumberOfTracks\', title="Top 10 Most Popular
Artists by Number of Tracks", template="plotly_dark")\n\nfig.update_traces(t
exttemplate=\'%{text}\', textposition=\'outside\')\nfig.show()\n```'}, 'done
_reason': 'stop', 'done': True, 'total_duration': 14663704906, 'load_duratio
n': 665606, 'prompt_eval_count': 265, 'prompt_eval_duration': 5396615000, 'e
val_count': 136, 'eval_duration': 9128547000}
```


Top 10 Most Popular Artists by Number of Tracks
NumberOfTracks
ArtistName
213
135
114
112
92
92
67
57
56
53
Iron Maiden
U2
Led Zeppelin
Metallica
Deep Purple
Lost
Pearl Jam
Lenny Kravitz
Various Artists
The Office
0
50
100
150
200

Out[32]:
```
('SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM ar
tists ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN track
s t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTrack
s DESC\nLIMIT 10',
         ArtistName  NumberOfTracks
 0      Iron Maiden             213
 1               U2             135
 2     Led Zeppelin             114
 3        Metallica             112
 4      Deep Purple              92
 5             Lost              92
 6        Pearl Jam              67
 7    Lenny Kravitz              57
 8  Various Artists              56
 9       The Office              53,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'ArtistName=%{x}<br>NumberOfTracks=%{marke
r.color}<extra></extra>',
               'legendgroup': '',
               'marker': {'color': array([213, 135, 114, 112,  92,  92,  6
7,  57,  56,  53]),
                          'coloraxis': 'coloraxis',
                          'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'text': array([213., 135., 114., 112.,  92.,  92.,  67.,  5
7.,  56.,  53.]),
               'textposition': 'outside',
               'texttemplate': '%{text}',
               'type': 'bar',
               'x': array(['Iron Maiden', 'U2', 'Led Zeppelin', 'Metallic
a', 'Deep Purple', 'Lost',
                           'Pearl Jam', 'Lenny Kravitz', 'Various Artists',
'The Office'],
                          dtype=object),
               'xaxis': 'x',
               'y': array([213, 135, 114, 112,  92,  92,  67,  57,  56,  5
3]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'coloraxis': {'colorbar': {'title': {'text': 'NumberOfTrack
s'}},
                              'colorscale': [[0.0, '#0d0887'], [0.111111111
1111111,
                                             '#46039f'], [0.222222222222222
2,
                                             '#7201a8'], [0.333333333333333
3,
                                             '#9c179e'], [0.444444444444444
4,
                                             '#bd3786'], [0.555555555555555
6,
                                             '#d8576b'], [0.666666666666666
```

```
6,
                                                  '#ed7953'], [0.777777777777777
8,
                                                  '#fb9f3a'], [0.888888888888888
8,
                                                  '#fdca26'], [1.0, '#f0f92
1']]},
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Top 10 Most Popular Artists by Number of
Tracks'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'ArtistName'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'NumberOfTracks'}}}
 }))
```

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

vn.ask(question=question)
```

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

```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_CustomerSupportRepId ON "cus
tomers" (SupportRepId)\n\nCREATE TABLE "customers"\r\n(\r\n    CustomerId IN
TEGER 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    Ph
one NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NU
LL,\r\n    SupportRepId INTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCE
S "employees" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUT
OINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate
DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity N
VARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCountry NVARCHA
R(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NO
T 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_Inv
oiceCustomerId ON "invoices" (CustomerId)\n\nCREATE TABLE "employees"\r\n(\r
\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    LastName
NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NOT NULL,\r\n    Titl
e NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDate DATETIME,\r\n
HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    City NVARCHAR(40),\r
\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n    PostalCode NVA
RCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(24),\r\n    Email
NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENCES "employees" (Employe
eId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "i
nvoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NO
T NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NUL
L,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT NU
LL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFE
RENCES "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_trac
k"\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) REFERENCES "playlists" (PlaylistId) \r\n\t\tO
N DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFEREN
CES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE
TABLE "albums"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Title NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NUL
L,\r\n    FOREIGN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn
the chinook database invoice means order\n\n===Response Guidelines \n1. If t
he provided context is sufficient, please generate a valid SQL query without
any explanations for the question. \n2. If the provided context is almost su
fficient but requires knowledge of a specific string in a particular column,
please generate an intermediate SQL query to find the distinct strings in th
at column. Prepend the query with a comment saying intermediate_sql \n3. If
the provided context is insufficient, please explain why it can\'t be genera
ted. \n4. Please use the most relevant table(s). \n5. If the question has be
en asked and answered before, please repeat the answer exactly as it was giv
en before. \n'}, {'role': 'user', 'content': '  \n    Get the total number o
f invoices for each customer\n'}, {'role': 'assistant', 'content': 'SELECT
```

c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\n
FROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP
BY c.CustomerId'}, {'role': 'user', 'content': 'How many customers are ther
e'}, {'role': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers
FROM customers'}, {'role': 'user', 'content': ' \n   Find the total number
of invoices per country:\n'}, {'role': 'assistant', 'content': 'SELECT Billi
ngCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountr
y'}, {'role': 'user', 'content': ' \n   Get the average invoice total for
each customer:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId,
c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers
c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerI
d'}, {'role': 'user', 'content': " \n   List all employees and their repor
ting manager's name (if any):\n"}, {'role': 'assistant', 'content': "SELECT
e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.Fir
stName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e
1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {'role': 'use
r', 'content': ' \n   List all invoices with a total exceeding $10:\n'},
{'role': 'assistant', 'content': 'SELECT * \nFROM invoices \nWHERE Total > 1
0'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 and t
he total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT Invo
iceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role':
'user', 'content': 'Can you list all tables in the SQLite database catalo
g?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHERE
type='table'"}, {'role': 'user', 'content': ' \n   Find the top 5 most exp
ensive tracks (based on unit price):\n'}, {'role': 'assistant', 'content':
'SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5'},
{'role': 'user', 'content': ' \n   List all albums and their corresponding
artist names \n'}, {'role': 'assistant', 'content': 'SELECT a.Title AS Albu
mTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId
= ar.ArtistId'}, {'role': 'user', 'content': ' \n   List all customers fr
om Canada and their email addresses:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_CustomerSupportRepId ON \"customers\" (S
upportRepId)\n\nCREATE TABLE \"customers\"\r\n(\r\n   CustomerId INTEGER PR
IMARY KEY AUTOINCREMENT NOT NULL,\r\n   FirstName NVARCHAR(40) NOT NULL,\r
\n   LastName NVARCHAR(20) NOT NULL,\r\n   Company NVARCHAR(80),\r\n   A
ddress NVARCHAR(70),\r\n   City NVARCHAR(40),\r\n   State NVARCHAR(40),\r
\n   Country NVARCHAR(40),\r\n   PostalCode NVARCHAR(10),\r\n   Phone NVA
RCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email NVARCHAR(60) NOT NULL,\r\n
SupportRepId INTEGER,\r\n   FOREIGN KEY (SupportRepId) REFERENCES \"employe
es\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCR
EATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY AUTOINCREM
ENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceDate DATETI
ME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCity NVARCHA
R(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVARCHAR(4
0),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2) NOT N
ULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (CustomerId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_Inv
oiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE TABLE \"employees\"\r

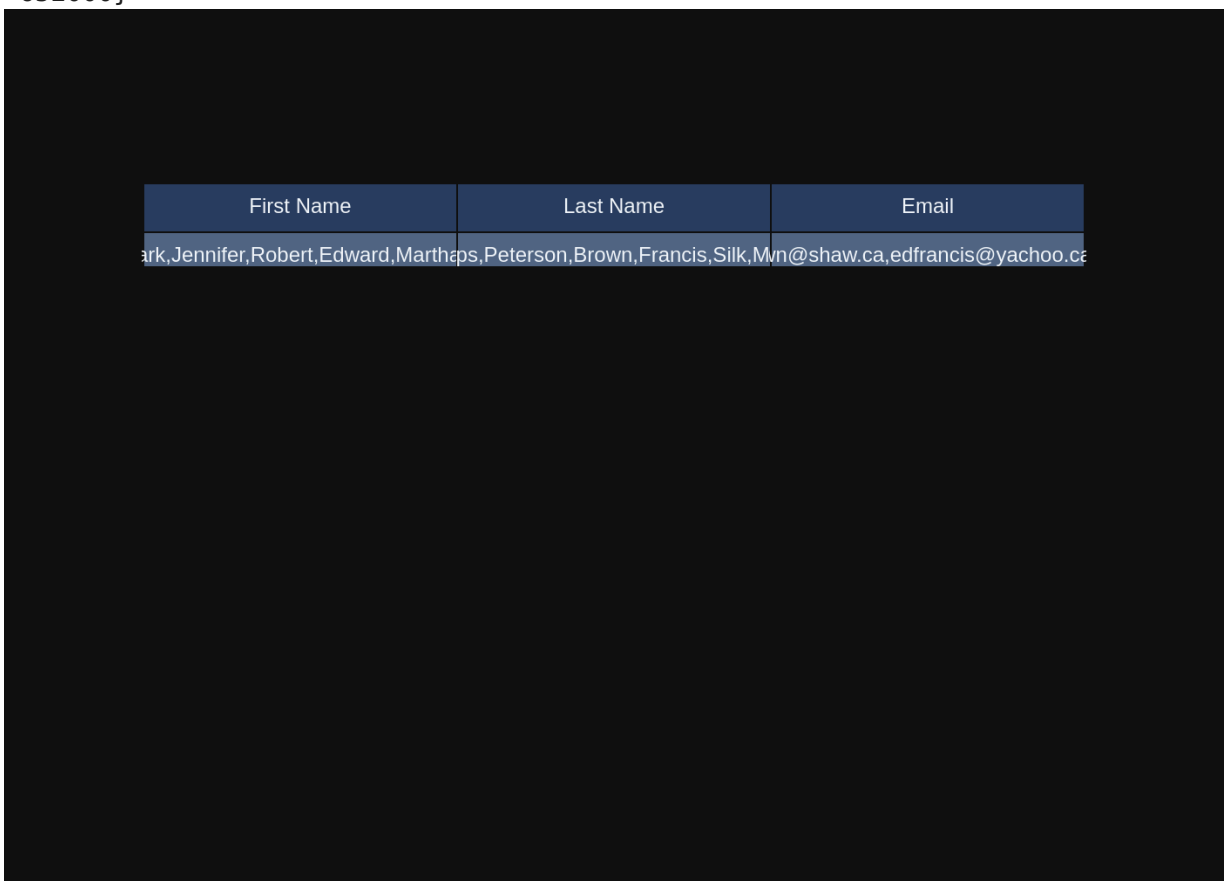
\n(\r\n   EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Las
tName NVARCHAR(20) NOT NULL,\r\n   FirstName NVARCHAR(20) NOT NULL,\r\n
Title NVARCHAR(30),\r\n   ReportsTo INTEGER,\r\n   BirthDate DATETIME,\r\n
HireDate DATETIME,\r\n   Address NVARCHAR(70),\r\n   City NVARCHAR(40),\r
\n   State NVARCHAR(40),\r\n   Country NVARCHAR(40),\r\n   PostalCode NVA
RCHAR(10),\r\n   Phone NVARCHAR(24),\r\n   Fax NVARCHAR(24),\r\n   Email
NVARCHAR(60),\r\n   FOREIGN KEY (ReportsTo) REFERENCES \"employees\" (Emplo
yeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE
\"invoice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT
NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT
NULL,\r\n   FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId)
REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION\r\n)\n\nCREATE TABLE sqlite_sequence(name,seq)\n\nCREATE TABLE \"playlis
t_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   TrackId INTEGER
NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, Track
Id),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\" (PlaylistId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON \"employees\" (ReportsT
o)\n\nCREATE TABLE \"albums\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n   Title NVARCHAR(160) NOT NULL,\r\n   ArtistId INTE
GER NOT NULL,\r\n   FOREIGN KEY (ArtistId) REFERENCES \"artists\" (ArtistI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional C
ontext \n\nIn the chinook database invoice means order\n\n===Response Guidel
ines \n1. If the provided context is sufficient, please generate a valid SQL
query without any explanations for the question. \n2. If the provided contex
t is almost sufficient but requires knowledge of a specific string in a part
icular column, please generate an intermediate SQL query to find the distinc
t strings in that column. Prepend the query with a comment saying intermedia
te_sql \n3. If the provided context is insufficient, please explain why it c
an't be generated. \n4. Please use the most relevant table(s). \n5. If the q
uestion has been asked and answered before, please repeat the answer exactly
as it was given before. \n"}, {"role": "user", "content": " \n   Get the t
otal number of invoices for each customer\n"}, {"role": "assistant", "conten
t": "SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS Tot
alInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.Custo
merId\nGROUP BY c.CustomerId"}, {"role": "user", "content": "How many custom
ers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) AS Number
OfCustomers FROM customers"}, {"role": "user", "content": " \n   Find the
total number of invoices per country:\n"}, {"role": "assistant", "content":
"SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY B
illingCountry"}, {"role": "user", "content": " \n   Get the average invoic
e total for each customer:\n"}, {"role": "assistant", "content": "SELECT c.C
ustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFRO
M customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY
c.CustomerId"}, {"role": "user", "content": " \n   List all employees and
their reporting manager's name (if any):\n"}, {"role": "assistant", "conten
t": "SELECT e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COA
LESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM
employees e1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {"ro
le": "user", "content": " \n   List all invoices with a total exceeding $1
0:\n"}, {"role": "assistant", "content": "SELECT * \nFROM invoices \nWHERE T
otal > 10"}, {"role": "user", "content": " \n   Find all invoices since 20
10 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SEL

ECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   List all customers from Canada and their email addresses:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:25:33.112756944Z', 'message': {'role': 'assistant', 'content': "SELECT FirstName, LastName, Email\nFROM customers\nWHERE Country = 'Canada'"}, 'done_reason': 'stop', 'done': True, 'total_duration': 48654659937, 'load_duration': 734659, 'prompt_eval_count': 1630, 'prompt_eval_duration': 46519909000, 'eval_count': 19, 'eval_duration': 1455428000}
LLM Response: SELECT FirstName, LastName, Email
FROM customers
WHERE Country = 'Canada'
SELECT FirstName, LastName, Email
FROM customers
WHERE Country = 'Canada'

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

Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that contains the results of the query that answers the question the user asked: '\n   List all customers from Canada and their email addresses:\n'\n\nThe DataFrame was produced using this query: SELECT FirstName, LastName, Email\nFROM customers\nWHERE Country = 'Canada'\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n FirstName   object\nLastName   object\nEmail   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."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:25:46.514483677Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport plotly.graph_objects as go\nimport numpy as np\n\nif df.shape[0] == 1:\n   fig = go.Figure(go.Indicator(\n   mode="number",\n   value=df[\'FirstName\'].iloc[0],\n   title={"text": "Customer"},\n   number={\'pr

efix\': ""}\n   ))\nelse:\n   fig = go.Figure(data=[go.Table(header=dict(v
alues=[\'First Name\', \'Last Name\', \'Email\']),\n
cells=dict(values=[[df[\'FirstName\']], [df[\'LastName\']], [df[\'Email
\']])])])\n\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_
duration': 13374178186, 'load_duration': 658562, 'prompt_eval_count': 172,
'prompt_eval_duration': 3347317000, 'eval_count': 150, 'eval_duration': 9893
652000}



Out[33]:
```
("SELECT FirstName, LastName, Email\nFROM customers\nWHERE Country = 'Canad
a'",
   FirstName  LastName                  Email
 0  François  Tremblay     ftremblay@gmail.com
 1      Mark   Philips      mphilips12@shaw.ca
 2  Jennifer  Peterson     jenniferp@rogers.ca
 3    Robert     Brown        robbrown@shaw.ca
 4    Edward   Francis     edfrancis@yachoo.ca
 5    Martha      Silk    marthasilk@gmail.com
 6     Aaron  Mitchell  aaronmitchell@yahoo.ca
 7     Ellie  Sullivan  ellie.sullivan@shaw.ca,
 Figure({
     'data': [{'cells': {'values': [[['François', 'Mark', 'Jennifer', 'Robe
rt',
                                      'Edward', 'Martha', 'Aaron', 'Ellie']],
                                      [['Tremblay', 'Philips', 'Peterson', 'B
rown',
                                      'Francis', 'Silk', 'Mitchell', 'Sulliva
n']],
                                      [['ftremblay@gmail.com', 'mphilips12@sh
aw.ca',
                                      'jenniferp@rogers.ca', 'robbrown@shaw.c
a',
                                      'edfrancis@yachoo.ca', 'marthasilk@gmai
l.com',
                                      'aaronmitchell@yahoo.ca',
                                      'ellie.sullivan@shaw.ca']]]},
                'header': {'values': ['First Name', 'Last Name', 'Email']},
                'type': 'table'}],
     'layout': {'template': '...'}
 }))
```

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

vn.ask(question=question)
```

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

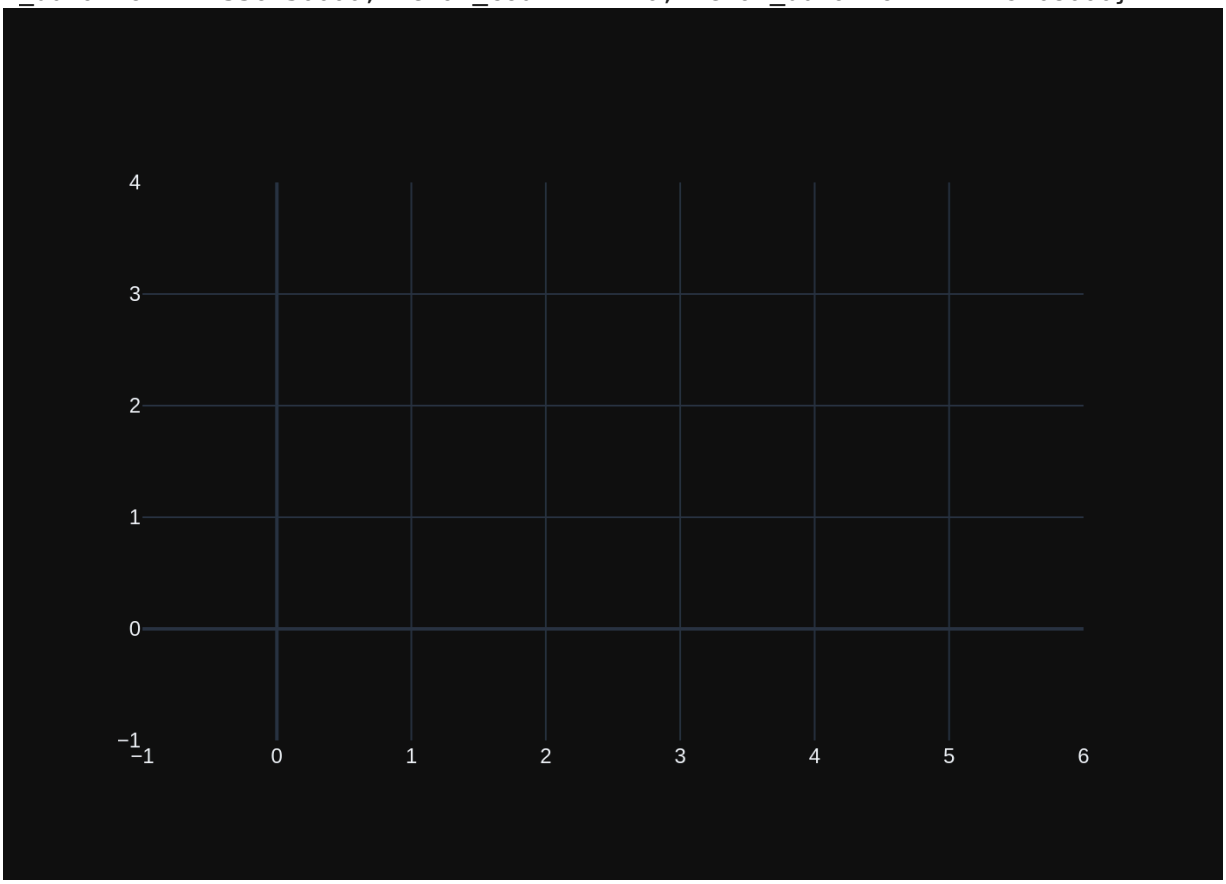
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)\n\nCREATE IN
DEX IFK_InvoiceLineInvoiceId ON "invoice_items" (InvoiceId)\n\nCREATE TABLE
"invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT
NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER  NOT NU
LL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER  NOT N
ULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (InvoiceId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackId) REFE
RENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nC
REATE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCRE
MENT 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(7
0),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVAR
CHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_
CustomerSupportRepId ON "customers" (SupportRepId)\n\nCREATE TABLE "employee
s"\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 DATETIM
E,\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    Posta
lCode 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 IN
DEX IFK_EmployeeReportsTo ON "employees" (ReportsTo)\n\nCREATE TABLE "track
s"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Na
me NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEG
ER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n
Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumI
d) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (Genr
eId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaType
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional
Context \n\nIn the chinook database invoice means order\n\n===Response Guide
lines \n1. If the provided context is sufficient, please generate a valid SQ
L query without any explanations for the question. \n2. If the provided cont
ext is almost sufficient but requires knowledge of a specific string in a pa
rticular column, please generate an intermediate SQL query to find the disti
nct strings in that column. Prepend the query with a comment saying intermed
iate_sql \n3. If the provided context is insufficient, please explain why it
can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the
question has been asked and answered before, please repeat the answer exactl
y as it was given before. \n'}, {'role': 'user', 'content': '  \n    Get the
```

total number of invoices for each customer\n'}, {'role': 'assistant', 'conte
nt': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS To
talInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.Cust
omerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n   List a
ll invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'conten
t': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'conten
t': ' \n   Get the average invoice total for each customer:\n'}, {'role':
'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, AVG
(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON
c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'cont
ent': ' \n   Find all invoices since 2010 and the total amount invoice
d:\n'}, {'role': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM in
voices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': '
\n   Find the total number of invoices per country:\n'}, {'role': 'assistan
t', 'content': 'SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoi
ces\nGROUP BY BillingCountry'}, {'role': 'user', 'content': ' \n   Find th
e top 5 most expensive tracks (based on unit price):\n'}, {'role': 'assistan
t', 'content': 'SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC
\nLIMIT 5'}, {'role': 'user', 'content': 'How many customers are there'},
{'role': 'assistant', 'content': 'SELECT COUNT(*) AS NumberOfCustomers FROM
customers'}, {'role': 'user', 'content': ' \n   List all customers from C
anada and their email addresses:\n'}, {'role': 'assistant', 'content': "SELE
CT FirstName, LastName, Email\nFROM customers\nWHERE Country = 'Canada'"},
{'role': 'user', 'content': ' \n   There are 3 tables: artists, albums and
tracks, where albums and artists are linked by ArtistId, albums and tracks a
re linked by AlbumId,\n   Can you find the top 10 most popular artists base
d on the number of tracks\n'}, {'role': 'assistant', 'content': 'SELECT ar.N
ame AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT
JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumI
d = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 1
0'}, {'role': 'user', 'content': " \n   List all employees and their repor
ting manager's name (if any):\n"}, {'role': 'assistant', 'content': "SELECT
e1.FirstName || ' ' || e1.LastName AS EmployeeName, \n   COALESCE(e2.Fir
stName || ' ' || e2.LastName, 'No Manager') AS ManagerName\nFROM employees e
1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.EmployeeId"}, {'role': 'use
r', 'content': ' \n   Find the customer with the most invoices \n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n
InvoiceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   B
illingCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCou
ntry NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMER
IC(10,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK
_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE TABLE \"inv
oice_items\"\r\n(\r\n   InvoiceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT
NULL,\r\n   InvoiceId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NUL
L,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   Quantity INTEGER NOT NU

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

voices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "conten
t": " \n   Find the top 5 most expensive tracks (based on unit pric
e):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nFROM trac
ks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "content": "How man
y customers are there"}, {"role": "assistant", "content": "SELECT COUNT(*) A
S NumberOfCustomers FROM customers"}, {"role": "user", "content": " \n
List all customers from Canada and their email addresses:\n"}, {"role": "ass
istant", "content": "SELECT FirstName, LastName, Email\nFROM customers\nWHER
E Country = 'Canada'"}, {"role": "user", "content": " \n   There are 3 tabl
es: artists, albums and tracks, where albums and artists are linked by Artis
tId, albums and tracks are linked by AlbumId,\n   Can you find the top 10 m
ost popular artists based on the number of tracks\n"}, {"role": "assistant",
"content": "SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks
\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JO
IN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY Number
OfTracks DESC\nLIMIT 10"}, {"role": "user", "content": " \n   List all emp
loyees and their reporting manager's name (if any):\n"}, {"role": "assistan
t", "content": "SELECT e1.FirstName || ' ' || e1.LastName AS EmployeeName,
\n   COALESCE(e2.FirstName || ' ' || e2.LastName, 'No Manager') AS Manag
erName\nFROM employees e1\nLEFT JOIN employees e2 ON e1.ReportsTo = e2.Emplo
yeeId"}, {"role": "user", "content": " \n   Find the customer with the mo
st invoices \n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:26:44.249
392415Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId,
c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers
c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerI
d\nORDER BY TotalInvoices DESC\nLIMIT 1'}, 'done_reason': 'stop', 'done': Tr
ue, 'total_duration': 57661851688, 'load_duration': 760047, 'prompt_eval_cou
nt': 1755, 'prompt_eval_duration': 51481580000, 'eval_count': 63, 'eval_dura
tion': 5490453000}
LLM Response: SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceI
d) AS TotalInvoices
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalInvoices DESC
LIMIT 1
SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInv
oices
FROM customers c
LEFT JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalInvoices DESC
LIMIT 1
   CustomerId FirstName   LastName TotalInvoices
0   1   Luís Gonçalves   7
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n   Find the customer with the most invoices \n'\n\nThe DataFrame was pro
duced using this query: SELECT c.CustomerId, c.FirstName, c.LastName, COUNT

(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1\n\nThe following is information about the resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId   int64\nFirstName   object\nLastName   object\nTotalInvoices   int64\ndtype: object"}, {"role": "user", "content": "Can you generate the Python plotly code to chart the results of the dataframe? Assume the data is in a pandas dataframe called 'df'. If there is only one value in the dataframe, use an Indicator. Respond with only Python code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:26:56.387231781Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport plotly.graph_objects as go\nimport numpy as np\n\nif df[\'TotalInvoices\'].iloc[0] == 1:\n   fig = go.Figure(go.Indicator(\n   mode="number",\n   value=df[\'TotalInvoices\'].iloc[0],\n   title={"text": "Total Invoices"},\n   number={"prefix": ""}\n   ))\nelse:\n   fig = go.Figure()\n\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 12115798907, 'load_duration': 686230, 'prompt_eval_count': 224, 'prompt_eval_duration': 4556130000, 'eval_count': 110, 'eval_duration': 7429708000}



Out[34]: ('SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1',
    CustomerId FirstName   LastName TotalInvoices
 0   1   Luís Gonçalves   7,
 Figure({
     'data': [], 'layout': {'template': '...'}
 }))

In [ ]:

Advanced SQL questions

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

vn.ask(question=question)
```

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

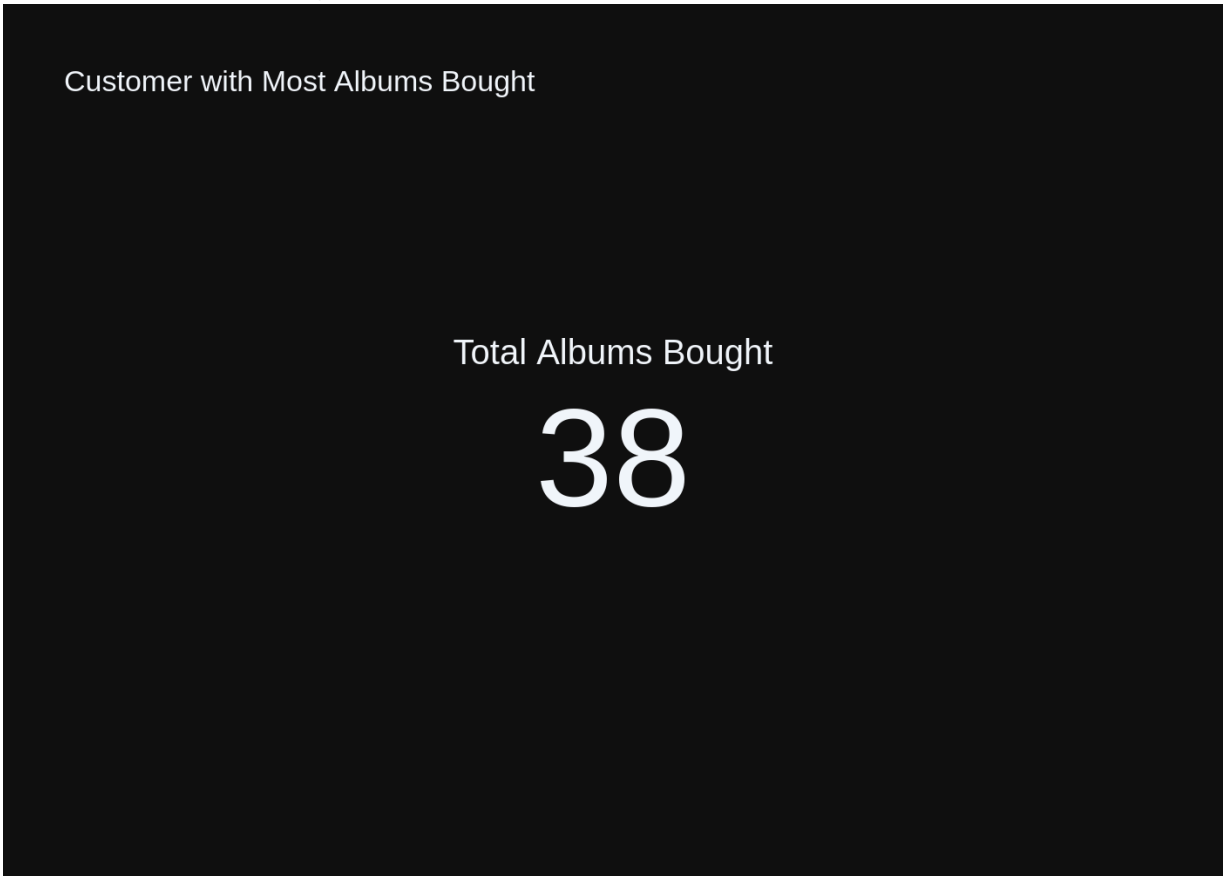
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER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

e': 'user', 'content': ' \n   Get the total number of invoices for each cu
stomer\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstNa
me, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT
JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'r
ole': 'user', 'content': ' \n   Find all invoices since 2010 and the total
amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT InvoiceDate,
Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user',
'content': ' \n   Find the top 5 most expensive tracks (based on unit pric
e):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\nFROM trac
ks\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': ' \n
List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'co
ntent': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'co
ntent': ' \n   Get the average invoice total for each customer:\n'}, {'rol
e': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, A
VG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i O
N c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'co
ntent': ' \n   Find the total number of invoices per country:\n'}, {'rol
e': 'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS TotalInvoice
s\nFROM invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'content': '
\n   List all albums and their corresponding artist names \n'}, {'role':
'assistant', 'content': 'SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName
\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId'}, {'role': 'us
er', 'content': ' \n   List all genres and the number of tracks in each ge
nre:\n'}, {'role': 'assistant', 'content': 'SELECT g.Name AS Genre, COUNT(t.
TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId =
t.GenreId\nGROUP BY g.GenreId'}, {'role': 'user', 'content': ' \n   Find
the customer who bought the most albums in total quantity (across all invoic
es): \n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"invoice_items\"\r\n(\r\n   Invoic
eLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER
NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2)
NOT NULL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId)
REFERENCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"albums\"\r
\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title
NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN K
EY (ArtistId) REFERENCES \"artists\" (ArtistId) \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 \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY K
EY AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   Invoi
ceDate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   Billin
gCity NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry
NVARCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(1
0,2) NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (Cu
stomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE IND
EX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCREATE INDEX IFK
_InvoiceLineInvoiceId ON \"invoice_items\" (InvoiceId)\n\nCREATE INDEX IFK_I
nvoiceCustomerId ON \"invoices\" (CustomerId)\n\nCREATE INDEX IFK_TrackAlbum
Id ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n
\n\n===Additional Context \n\nIn the chinook database invoice means order\n
\n===Response Guidelines \n1. If the provided context is sufficient, please
generate a valid SQL query without any explanations for the question. \n2. I
f the provided context is almost sufficient but requires knowledge of a spec
ific string in a particular column, please generate an intermediate SQL quer
y to find the distinct strings in that column. Prepend the query with a comm
ent saying intermediate_sql \n3. If the provided context is insufficient, pl
ease explain why it can't be generated. \n4. Please use the most relevant ta
ble(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", "conten
t": " \n   Find the customer with the most invoices \n"}, {"role": "assis
tant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.Inv
oiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.Custom
erId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIM
IT 1"}, {"role": "user", "content": " \n   There are 3 tables: artists, alb
ums 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 arti
sts based on the number of tracks\n"}, {"role": "assistant", "content": "SEL
ECT ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists
ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON
al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\n
LIMIT 10"}, {"role": "user", "content": " \n   Get the total number of inv
oices for each customer\n"}, {"role": "assistant", "content": "SELECT c.Cust
omerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM c
ustomers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.
CustomerId"}, {"role": "user", "content": " \n   Find all invoices since 2
010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SE
LECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"},
{"role": "user", "content": " \n   Find the top 5 most expensive tracks (b
ased on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, Uni
tPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "c
ontent": " \n   List all invoices with a total exceeding $10:\n"}, {"rol
e": "assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"},
{"role": "user", "content": " \n   Get the average invoice total for each
customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.Firs
tName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLE
FT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"},
{"role": "user", "content": " \n   Find the total number of invoices per c
ountry:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT
(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "us
er", "content": " \n   List all albums and their corresponding artist name
s \n"}, {"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.
Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.Artist
Id"}, {"role": "user", "content": " \n   List all genres and the number of

```
tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name A
S Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks
t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId"}, {"role": "user", "conten
t": "  \n     Find the customer who bought the most albums in total quantity
(across all invoices): \n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:27:48.884
385509Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId,
c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM custome
rs c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii
ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBo
ught DESC\nLIMIT 1'}, 'done_reason': 'stop', 'done': True, 'total_duration':
52417087982, 'load_duration': 728676, 'prompt_eval_count': 1563, 'prompt_eva
l_duration': 45113627000, 'eval_count': 81, 'eval_duration': 6609140000}
LLM Response: SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity)
AS TotalAlbumsBought
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
JOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsBought DESC
LIMIT 1
SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbum
sBought
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
JOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumsBought DESC
LIMIT 1
   CustomerId FirstName   LastName  TotalAlbumsBought
0           1      Luís  Gonçalves                 38
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n     Find the customer who bought the most albums in total quantity (acros
s all invoices): \n'\n\nThe DataFrame was produced using this query: SELECT
c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought
\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN inv
oice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY
TotalAlbumsBought DESC\nLIMIT 1\n\nThe following is information about the re
sulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId
int64\nFirstName            object\nLastName             object\nTotalAlbums
Bought     int64\ndtype: object"}, {"role": "user", "content": "Can you gene
rate the Python plotly code to chart the results of the dataframe? Assume th
e data is in a pandas dataframe called 'df'. If there is only one value in t
he dataframe, use an Indicator. Respond with only Python code. Do not answer
with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:28:05.788
315324Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\n\nif df[\'TotalAlbumsBought\'].nunique() == 1:\n
```

fig = go.Figure(go.Indicator(mode="number", value=df[\'TotalAlbumsBought\'].iloc[0], title={"text": "Total Albums Bought"}))\nelse:\n   fig = go.Figure([go.Bar(x=[f"{row[\'FirstName\']} {row[\'LastName\']}"], y=[row[\'TotalAlbumsBought\']]) for index, row in df.iterrows()])\n\nfig.update_layout(title="Customer with Most Albums Bought", xaxis_title="Customer Name", yaxis_title="Total Albums Bought")\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 16875193175, 'load_duration': 634377, 'prompt_eval_count': 253, 'prompt_eval_duration': 5063176000, 'eval_count': 171, 'eval_duration': 11679610000}



Out[35]:
```
('SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMIT 1',
    CustomerId FirstName   LastName  TotalAlbumsBought
 0           1      Luís  Gonçalves                 38,
 Figure({
     'data': [{'mode': 'number', 'title': {'text': 'Total Albums Bought'}, 'type': 'indicator', 'value': 38}],
     'layout': {'template': '...',
                'title': {'text': 'Customer with Most Albums Bought'},
                'xaxis': {'title': {'text': 'Customer Name'}},
                'yaxis': {'title': {'text': 'Total Albums Bought'}}}
 }))
```

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

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

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

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

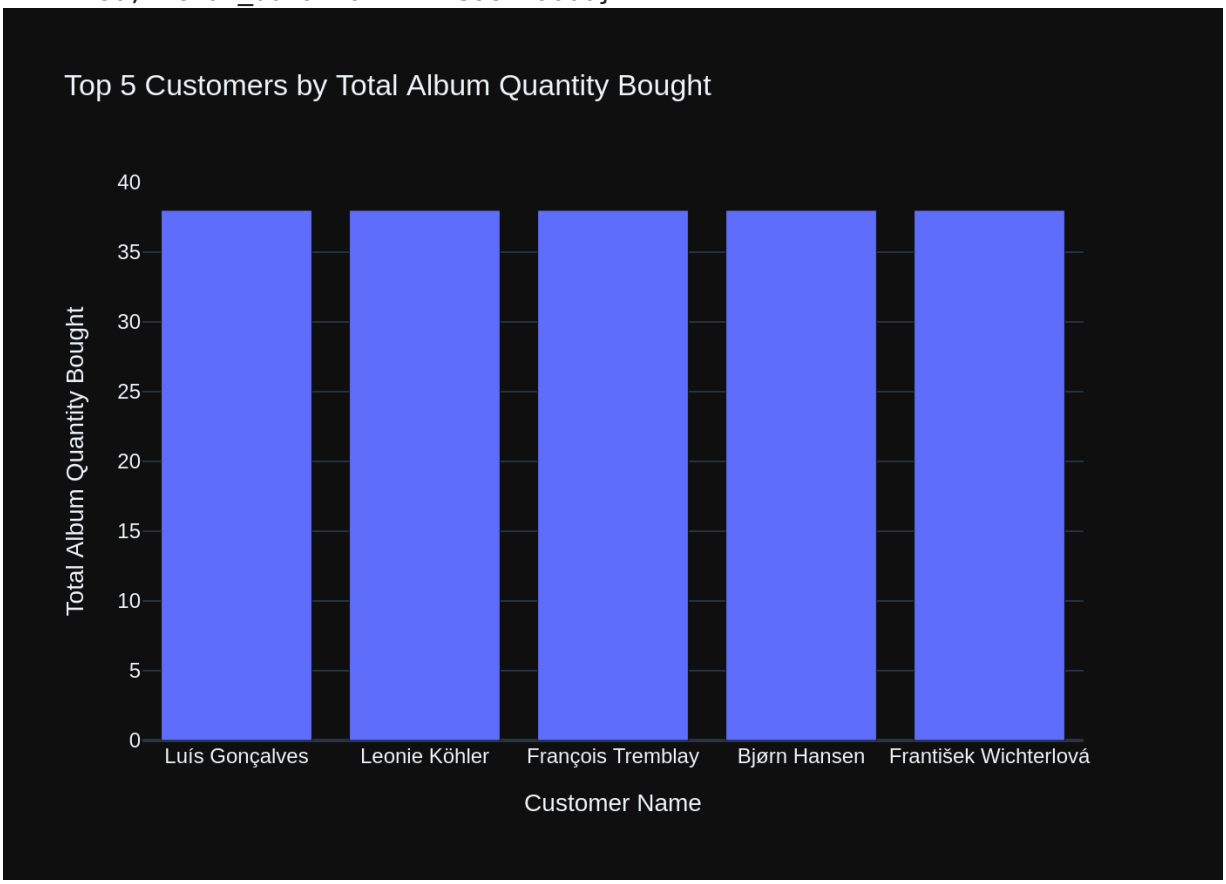
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoice_items"\r\n(\r\n    Invo
iceLineId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEG
ER  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,
2)  NOT NULL,\r\n    Quantity INTEGER  NOT NULL,\r\n    FOREIGN KEY (Invoice
Id) REFERENCES "invoices" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE
NO ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n
(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVA
RCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NO
T NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Millis
econds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,
2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId)
REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "album
s"\r\n(\r\n    AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Ti
tle NVARCHAR(160)  NOT NULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREI
GN KEY (ArtistId) REFERENCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (Ar
tistId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items" (Invoice
Id)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nC
REATE TABLE "invoices"\r\n(\r\n    InvoiceId INTEGER PRIMARY KEY AUTOINCREME
NT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n    InvoiceDate DATETIM
E  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    BillingCity NVARCHAR
(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCountry NVARCHAR(4
0),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMERIC(10,2)  NOT N
ULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "customers" (CustomerId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_Invoi
ceCustomerId ON "invoices" (CustomerId)\n\nCREATE INDEX IFK_TrackAlbumId ON
"tracks" (AlbumId)\n\nCREATE TABLE "artists"\r\n(\r\n    ArtistId INTEGER PR
IMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Add
itional Context \n\nIn the chinook database invoice means order\n\n===Respon
se Guidelines \n1. If the provided context is sufficient, please generate a
valid SQL query without any explanations for the question. \n2. If the provi
ded context is almost sufficient but requires knowledge of a specific string
in a particular column, please generate an intermediate SQL query to find th
e distinct strings in that column. Prepend the query with a comment saying i
ntermediate_sql \n3. If the provided context is insufficient, please explain
why it can\'t be generated. \n4. Please use the most relevant table(s). \n5.
If the question has been asked and answered before, please repeat the answer
exactly as it was given before. \n'}, {'role': 'user', 'content': '  \n
Find the customer who bought the most albums in total quantity (across all i
nvoices): \n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.Fir
stName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c
\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON
i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBough
t DESC\nLIMIT 1'}, {'role': 'user', 'content': '  \n     Find the customer w
ith the most invoices \n'}, {'role': 'assistant', 'content': 'SELECT c.Custo
merId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM cu
stomers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.C
ustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1'}, {'role': 'user', 'conten
t': ' \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 ar.Name AS ArtistName, COUNT
(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.Ar
tistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY
ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 10'}, {'role': 'user', 'con
tent': ' \n   Find the top 5 most expensive tracks (based on unit pric
e):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\nFROM trac
ks\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content': ' \n
Get the total number of invoices for each customer\n'}, {'role': 'assistan
t', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.Invoic
eId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerI
d = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'content': ' \n
List all invoices with a total exceeding $10:\n'}, {'role': 'assistant', 'co
ntent': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'user', 'co
ntent': ' \n   Get the average invoice total for each customer:\n'}, {'rol
e': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, A
VG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i O
N c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role': 'user', 'co
ntent': ' \n   Find the total number of invoices per country:\n'}, {'rol
e': 'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS TotalInvoice
s\nFROM invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'content': '
\n   Find all invoices since 2010 and the total amount invoiced:\n'}, {'rol
e': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM invoices\nWHERE
InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': ' \n   List all
albums and their corresponding artist names \n'}, {'role': 'assistant', 'co
ntent': 'SELECT a.Title AS AlbumTitle, 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 th
e top 5 customers who bought the most albums in total quantity (across all i
nvoices):\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoice_items\"\r\n(\r\n   InvoiceLineId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   InvoiceId INTEGER NOT NU
LL,\r\n   TrackId INTEGER NOT NULL,\r\n   UnitPrice NUMERIC(10,2) NOT NU
LL,\r\n   Quantity INTEGER NOT NULL,\r\n   FOREIGN KEY (InvoiceId) REFERE
NCES \"invoices\" (InvoiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON D
ELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n
TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(20
0) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NUL
L,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Millisecond
s INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) N
OT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFE
RENCES \"genres\" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTIO
N,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES \"media_types\" (MediaTypeId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"album
s\"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   T

itle NVARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FORE
IGN KEY (ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO AC
TION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums
\" (ArtistId)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\"
(InvoiceId)\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (Tra
ckId)\n\nCREATE TABLE \"invoices\"\r\n(\r\n   InvoiceId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n   CustomerId INTEGER NOT NULL,\r\n   InvoiceD
ate DATETIME NOT NULL,\r\n   BillingAddress NVARCHAR(70),\r\n   BillingCi
ty NVARCHAR(40),\r\n   BillingState NVARCHAR(40),\r\n   BillingCountry NVA
RCHAR(40),\r\n   BillingPostalCode NVARCHAR(10),\r\n   Total NUMERIC(10,2)
NOT NULL,\r\n   FOREIGN KEY (CustomerId) REFERENCES \"customers\" (Customer
Id) \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_TrackAlb
umId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"artists\"\r\n(\r\n   ArtistI
d INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)
\n\n\n===Additional Context \n\nIn the chinook database invoice means order
\n\n===Response Guidelines \n1. If the provided context is sufficient, pleas
e generate a valid SQL query without any explanations for the question. \n2.
If the provided context is almost sufficient but requires knowledge of a spe
cific string in a particular column, please generate an intermediate SQL que
ry 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, p
lease explain why it can't be generated. \n4. Please use the most relevant t
able(s). \n5. If the question has been asked and answered before, please rep
eat the answer exactly as it was given before. \n"}, {"role": "user", "conte
nt": " \n   Find the customer who bought the most albums in total quantit
y (across all invoices): \n"}, {"role": "assistant", "content": "SELECT c.Cu
stomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFR
OM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice
_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY Tot
alAlbumsBought DESC\nLIMIT 1"}, {"role": "user", "content": " \n   Find t
he customer with the most invoices \n"}, {"role": "assistant", "content": "S
ELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvo
ices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId
\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1"}, {"role": "u
ser", "content": " \n   There are 3 tables: artists, albums and tracks, whe
re albums and artists are linked by ArtistId, albums and tracks are linked b
y AlbumId,\n   Can you find the top 10 most popular artists based on the nu
mber of tracks\n"}, {"role": "assistant", "content": "SELECT ar.Name AS Arti
stName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN album
s al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.Albu
mId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 10"}, {"rol
e": "user", "content": " \n   Find the top 5 most expensive tracks (based
on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPric
e\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5"}, {"role": "user", "conten
t": " \n   Get the total number of invoices for each customer\n"}, {"rol
e": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, C
OUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i O
N c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "co
ntent": " \n   List all invoices with a total exceeding $10:\n"}, {"role":
"assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"}, {"ro
le": "user", "content": " \n   Get the average invoice total for each cust
omer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstNam
e, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT J
OIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"ro
le": "user", "content": " \n   Find the total number of invoices per count

```
ry:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) A
S TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user",
"content": "  \n    Find all invoices since 2010 and the total amount invoic
ed:\n"}, {"role": "assistant", "content": "SELECT InvoiceDate, Total\nFROM i
nvoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": "
\n    List all albums and their corresponding artist names  \n"}, {"role":
"assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName
\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "us
er", "content": "  \n    Hint: album quantity is found in invoice_items, \n
\n    Find the top 5 customers who bought the most albums in total quantity
(across all invoices):\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:28:59.853
520562Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId,
c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought\nFROM
customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_it
ems ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalA
lbumQuantityBought DESC\nLIMIT 5'}, 'done_reason': 'stop', 'done': True, 'to
tal_duration': 53992578705, 'load_duration': 776606, 'prompt_eval_count': 16
21, 'prompt_eval_duration': 46652240000, 'eval_count': 81, 'eval_duration':
6652234000}
LLM Response: SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity)
AS TotalAlbumQuantityBought
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
JOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumQuantityBought DESC
LIMIT 5
SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbum
QuantityBought
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
JOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId
GROUP BY c.CustomerId
ORDER BY TotalAlbumQuantityBought DESC
LIMIT 5
   CustomerId  FirstName     LastName  TotalAlbumQuantityBought
0           1       Luís    Gonçalves                        38
1           2     Leonie       Köhler                        38
2           3   François     Tremblay                        38
3           4      Bjørn       Hansen                        38
4           5  František  Wichterlová                        38
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n    Hint: album quantity is found in invoice_items, \n    \n    Find the t
op 5 customers who bought the most albums in total quantity (across all invo
ices):\n'\n\nThe DataFrame was produced using this query: SELECT c.CustomerI
d, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought\nFR
OM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice
_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY Tot
```

alAlbumQuantityBought DESC\nLIMIT 5\n\nThe following is information about th
e resulting pandas DataFrame 'df': \nRunning df.dtypes gives:\n CustomerId
int64\nFirstName object\nLastName objec
t\nTotalAlbumQuantityBought int64\ndtype: object"}, {"role": "user", "co
ntent": "Can you generate the Python plotly code to chart the results of the
dataframe? Assume the data is in a pandas dataframe called 'df'. If there is
only one value in the dataframe, use an Indicator. Respond with only Python
code. Do not answer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:29:18.083
088372Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport numpy as np\n\nif len(df) == 1:\n fig =
go.Figure(go.Indicator(\n mode="number",\n value=df[\'TotalAlb
umQuantityBought\'].iloc[0],\n title={"text": "Top Customer\'s Total
Album Quantity Bought"},\n number={\'prefix\': ""}\n))\nelse:\n
fig = go.Figure([go.Bar(x=df[\'FirstName\'] + \' \' + df[\'LastName\'], y=df
[\'TotalAlbumQuantityBought\'])])\n fig.update_layout(title=\'Top 5 Custo
mers by Total Album Quantity Bought\', xaxis_title=\'Customer Name\', yaxis_
title=\'Total Album Quantity Bought\')\n\nfig.show()\n```'}, 'done_reason':
'stop', 'done': True, 'total_duration': 18201390921, 'load_duration': 73342
6, 'prompt_eval_count': 271, 'prompt_eval_duration': 5671378000, 'eval_coun
t': 180, 'eval_duration': 12398249000}



Out[36]:
```
('SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAl
bumQuantityBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.Cu
stomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.C
ustomerId\nORDER BY TotalAlbumQuantityBought DESC\nLIMIT 5',
    CustomerId  FirstName     LastName  TotalAlbumQuantityBought
 0           1       Luís    Gonçalves                        38
 1           2     Leonie       Köhler                        38
 2           3   François     Tremblay                        38
 3           4      Bjørn       Hansen                        38
 4           5  František  Wichterlová                        38,
 Figure({
     'data': [{'type': 'bar',
               'x': array(['Luís Gonçalves', 'Leonie Köhler', 'François Tre
mblay', 'Bjørn Hansen',
                           'František Wichterlová'], dtype=object),
               'y': array([38, 38, 38, 38, 38])}],
     'layout': {'template': '...',
                'title': {'text': 'Top 5 Customers by Total Album Quantity
Bought'},
                'xaxis': {'title': {'text': 'Customer Name'}},
                'yaxis': {'title': {'text': 'Total Album Quantity Bough
t'}}}
 }))
```

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

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

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

vn.ask(question=question)
```

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

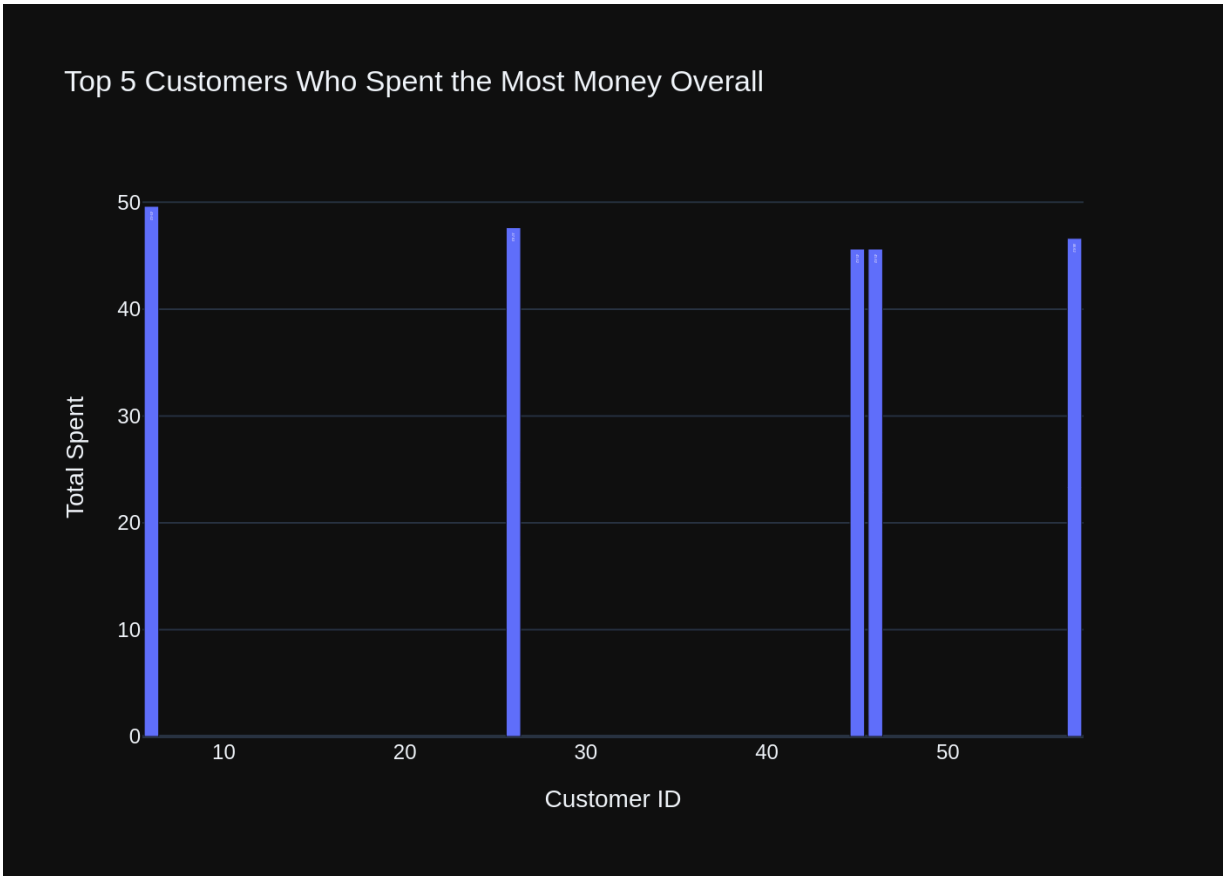
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE TABLE "invoices"\r\n(\r\n    InvoiceId
INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT N
ULL,\r\n    InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(7
0),\r\n    BillingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n
BillingCountry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    T
otal NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES "cu
stomers" (CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n
\nCREATE TABLE "invoice_items"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY
AUTOINCREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId I
NTEGER  NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity
INTEGER  NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES "invoices" (Inv
oiceId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY
(TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDAT
E NO ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON "invoice_items"
(InvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON "invoices" (CustomerId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON "invoice_items" (TrackId)\n\nCREA
TE TABLE "customers"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCREMEN
T NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVARCHA
R(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(70),\r
\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR
(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax
NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId INTEG
ER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES "employees" (EmployeeId) \r
\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "employee
s"\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 DATETIM
E,\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    Posta
lCode 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 TA
BLE "tracks"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGER,\r\n    Media
TypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Composer NVARCHAR(2
20),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTEGER,\r\n    Uni
tPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId) REFERENCES "alb
ums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREI
GN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFERENCES "media_type
s" (MediaTypeId) \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) REFERENCES "playlist
s" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FORE
IGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION
ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeReportsTo ON "employee
s" (ReportsTo)\n\n\n===Additional Context \n\nIn the chinook database invoic
e means order\n\n===Response Guidelines \n1. If the provided context is suff
icient, please generate a valid SQL query without any explanations for the q
uestion. \n2. If the provided context is almost sufficient but requires know
ledge of a specific string in a particular column, please generate an interm
ediate SQL query to find the distinct strings in that column. Prepend the qu
```

ery with a comment saying intermediate_sql \n3. If the provided context is i
nsufficient, please explain why it can\'t be generated. \n4. Please use the
most relevant table(s). \n5. If the question has been asked and answered bef
ore, please repeat the answer exactly as it was given before. \n'}, {'role':
'user', 'content': ' \n   Hint: album quantity is found in invoice_items,
\n   \n   Find the top 5 customers who bought the most albums in total qua
ntity (across all invoices):\n'}, {'role': 'assistant', 'content': 'SELECT
c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantit
yBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJ
OIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nO
RDER BY TotalAlbumQuantityBought DESC\nLIMIT 5'}, {'role': 'user', 'conten
t': ' \n   Find the customer with the most invoices \n'}, {'role': 'assis
tant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.Inv
oiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.Custom
erId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIM
IT 1'}, {'role': 'user', 'content': ' \n   Find the customer who bought t
he most albums in total quantity (across all invoices): \n'}, {'role': 'assi
stant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Qua
ntity) AS TotalAlbumsBought\nFROM customers c\nJOIN invoices i ON c.Customer
Id = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROU
P BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMIT 1'}, {'role': 'use
r', 'content': ' \n   Get the average invoice total for each custome
r:\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName,
c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN
invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role':
'user', 'content': ' \n   Get the total number of invoices for each custom
er\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName,
c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN
invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId'}, {'role':
'user', 'content': ' \n   Find the top 5 most expensive tracks (based on u
nit price):\n'}, {'role': 'assistant', 'content': 'SELECT Name, UnitPrice\nF
ROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'user', 'content':
' \n   List all invoices with a total exceeding $10:\n'}, {'role': 'assist
ant', 'content': 'SELECT * \nFROM invoices \nWHERE Total > 10'}, {'role': 'u
ser', 'content': ' \n   Find the total number of invoices per countr
y:\n'}, {'role': 'assistant', 'content': 'SELECT BillingCountry, COUNT(*) AS
TotalInvoices\nFROM invoices\nGROUP BY BillingCountry'}, {'role': 'user', 'c
ontent': ' \n   Find all invoices since 2010 and the total amount invoice
d:\n'}, {'role': 'assistant', 'content': "SELECT InvoiceDate, Total\nFROM in
voices\nWHERE InvoiceDate >= '2010-01-01'"}, {'role': 'user', 'content': '
\n   There are 3 tables: artists, albums and tracks, where albums and artis
ts are linked by ArtistId, albums and tracks are linked by AlbumId,\n   Can
you find the top 10 most popular artists based on the number of tracks\n'},
{'role': 'assistant', 'content': 'SELECT ar.Name AS ArtistName, COUNT(t.Trac
kId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId
= al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.Art
istId\nORDER BY NumberOfTracks 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 i
nvoice_items detail table is unnecessary \n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene

```
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"invoices\"\r\n(\r\n    InvoiceId INTEGER P
RIMARY KEY AUTOINCREMENT NOT NULL,\r\n    CustomerId INTEGER  NOT NULL,\r\n
InvoiceDate DATETIME  NOT NULL,\r\n    BillingAddress NVARCHAR(70),\r\n    B
illingCity NVARCHAR(40),\r\n    BillingState NVARCHAR(40),\r\n    BillingCou
ntry NVARCHAR(40),\r\n    BillingPostalCode NVARCHAR(10),\r\n    Total NUMER
IC(10,2)  NOT NULL,\r\n    FOREIGN KEY (CustomerId) REFERENCES \"customers\"
(CustomerId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE
TABLE \"invoice_items\"\r\n(\r\n    InvoiceLineId INTEGER PRIMARY KEY AUTOIN
CREMENT NOT NULL,\r\n    InvoiceId INTEGER  NOT NULL,\r\n    TrackId INTEGER
NOT NULL,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    Quantity INTEGER
NOT NULL,\r\n    FOREIGN KEY (InvoiceId) REFERENCES \"invoices\" (InvoiceId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (TrackI
d) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE INDEX IFK_InvoiceLineInvoiceId ON \"invoice_items\" (I
nvoiceId)\n\nCREATE INDEX IFK_InvoiceCustomerId ON \"invoices\" (CustomerId)
\n\nCREATE INDEX IFK_InvoiceLineTrackId ON \"invoice_items\" (TrackId)\n\nCR
EATE TABLE \"customers\"\r\n(\r\n    CustomerId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    FirstName NVARCHAR(40)  NOT NULL,\r\n    LastName NVA
RCHAR(20)  NOT NULL,\r\n    Company NVARCHAR(80),\r\n    Address NVARCHAR(7
0),\r\n    City NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVAR
CHAR(40),\r\n    PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n
Fax NVARCHAR(24),\r\n    Email NVARCHAR(60)  NOT NULL,\r\n    SupportRepId I
NTEGER,\r\n    FOREIGN KEY (SupportRepId) REFERENCES \"employees\" (Employee
Id) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"e
mployees\"\r\n(\r\n    EmployeeId INTEGER PRIMARY KEY AUTOINCREMENT NOT NUL
L,\r\n    LastName NVARCHAR(20)  NOT NULL,\r\n    FirstName NVARCHAR(20)  NO
T NULL,\r\n    Title NVARCHAR(30),\r\n    ReportsTo INTEGER,\r\n    BirthDat
e DATETIME,\r\n    HireDate DATETIME,\r\n    Address NVARCHAR(70),\r\n    Ci
ty NVARCHAR(40),\r\n    State NVARCHAR(40),\r\n    Country NVARCHAR(40),\r\n
PostalCode NVARCHAR(10),\r\n    Phone NVARCHAR(24),\r\n    Fax NVARCHAR(2
4),\r\n    Email NVARCHAR(60),\r\n    FOREIGN KEY (ReportsTo) REFERENCES \"e
mployees\" (EmployeeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)
\n\nCREATE TABLE \"tracks\"\r\n(\r\n    TrackId INTEGER PRIMARY KEY AUTOINCR
EMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r\n    AlbumId INTEGE
R,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    GenreId INTEGER,\r\n    Comp
oser NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NULL,\r\n    Bytes INTE
GER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FOREIGN KEY (AlbumId)
REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACT
ION,\r\n    FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId) \r\n\t\tON
DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (MediaTypeId) REFER
ENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n    PlaylistId INTEGE
R  NOT NULL,\r\n    TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_Playlist
Track PRIMARY KEY  (PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) RE
FERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO
ACTION,\r\n    FOREIGN KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t
\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_EmployeeR
eportsTo ON \"employees\" (ReportsTo)\n\n\n===Additional Context \n\nIn the
chinook database invoice means order\n\n===Response Guidelines \n1. If the p
rovided context is sufficient, please generate a valid SQL query without any
explanations for the question. \n2. If the provided context is almost suffic
ient but requires knowledge of a specific string in a particular column, ple
ase generate an intermediate SQL query to find the distinct strings in that
column. Prepend the query with a comment saying intermediate_sql \n3. If the
```

provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been 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 the most albums in total quantity (across all invoices):\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumQuantityBought DESC\nLIMIT 5"}, {"role": "user", "content": " \n   Find the customer with the most invoices \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalInvoices DESC\nLIMIT 1"}, {"role": "user", "content": " \n   Find the customer who bought the most albums in total quantity (across all invoices): \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMIT 1"}, {"role": "user", "content": " \n   Get the average invoice total for each customer:\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, AVG(i.Total) AS AverageInvoiceTotal\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   Get the total number of invoices for each customer\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, COUNT(i.InvoiceId) AS TotalInvoices\nFROM customers c\nLEFT JOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nFROM tracks\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"}, {"role": "user", "content": " \n   Find the total number of invoices per country:\n"}, {"role": "assistant", "content": "SELECT BillingCountry, COUNT(*) AS TotalInvoices\nFROM invoices\nGROUP BY BillingCountry"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"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 ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks 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"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:29:50.074871257Z', 'message': {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.Total) AS TotalSpent\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP BY c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5'}, 'done_reason': 'stop', 'done': True, 'total_duration': 31911790138, 'load_duration': 1193894, 'prompt_eval_count': 1025, 'pro

```
mpt_eval_duration': 26472699000, 'eval_count': 61, 'eval_duration': 47446550
00}
LLM Response: SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.Total) AS
TotalSpent
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalSpent DESC
LIMIT 5
SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.Total) AS TotalSpent
FROM customers c
JOIN invoices i ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalSpent DESC
LIMIT 5
   CustomerId FirstName    LastName  TotalSpent
0           6    Helena        Holý       49.62
1          26   Richard  Cunningham       47.62
2          57      Luis       Rojas       46.62
3          45  Ladislav      Kovács       45.62
4          46      Hugh    O'Reilly       45.62
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n     Find the top 5 customers who spent the most money overall, \n     \n
Hint: order total can be found on invoices table, calculation using invoice_
items detail table is unnecessary \n'\n\nThe DataFrame was produced using th
is query: SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.Total) AS Tota
lSpent\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nGR
OUP BY c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5\n\nThe following is i
nformation about the resulting pandas DataFrame 'df': \nRunning df.dtypes gi
ves:\n CustomerId      int64\nFirstName      object\nLastName       object\n
TotalSpent    float64\ndtype: object"}, {"role": "user", "content": "Can you
generate the Python plotly code to chart the results of the dataframe? Assum
e 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 an
swer with any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:30:05.169
691676Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.express as px\n\nif df[\'TotalSpent\'].nunique() == 1:\n    fig = px.in
dicator(title="Top Customers by Total Spent", value=df[\'TotalSpent\'].iloc
[0], label="Total Spent")\nelse:\n    fig = px.bar(df, x=\'CustomerId\', y=
\'TotalSpent\', text=\'TotalSpent\', title=\'Top 5 Customers Who Spent the M
ost Money Overall\')\n\nfig.update_layout(xaxis_title=\'Customer ID\', yaxis
_title=\'Total Spent\')\nfig.show()\n```'}, 'done_reason': 'stop', 'done': T
rue, 'total_duration': 15066615681, 'load_duration': 707241, 'prompt_eval_co
unt': 253, 'prompt_eval_duration': 5212729000, 'eval_count': 139, 'eval_dura
tion': 9723257000}
```


Top 5 Customers Who Spent the Most Money Overall
Total Spent
0
10
20
30
40
50
10
20
30
40
50
Customer ID

Out[37]:
```
('SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.Total) AS TotalSpent
\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nGROUP B
Y c.CustomerId\nORDER BY TotalSpent DESC\nLIMIT 5',
    CustomerId FirstName    LastName  TotalSpent
 0           6    Helena        Holý       49.62
 1          26   Richard  Cunningham       47.62
 2          57      Luis       Rojas       46.62
 3          45  Ladislav      Kovács       45.62
 4          46      Hugh    O'Reilly       45.62,
 Figure({
     'data': [{'alignmentgroup': 'True',
               'hovertemplate': 'CustomerId=%{x}<br>TotalSpent=%{text}<extr
a></extra>',
               'legendgroup': '',
               'marker': {'color': '#636efa', 'pattern': {'shape': ''}},
               'name': '',
               'offsetgroup': '',
               'orientation': 'v',
               'showlegend': False,
               'text': array([49.62, 47.62, 46.62, 45.62, 45.62]),
               'textposition': 'auto',
               'type': 'bar',
               'x': array([ 6, 26, 57, 45, 46]),
               'xaxis': 'x',
               'y': array([49.62, 47.62, 46.62, 45.62, 45.62]),
               'yaxis': 'y'}],
     'layout': {'barmode': 'relative',
                'legend': {'tracegroupgap': 0},
                'template': '...',
                'title': {'text': 'Top 5 Customers Who Spent the Most Money
Overall'},
                'xaxis': {'anchor': 'y', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Customer ID'}},
                'yaxis': {'anchor': 'x', 'domain': [0.0, 1.0], 'title': {'t
ext': 'Total Spent'}}}
 }))
```

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

vn.ask(question=question)
```

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

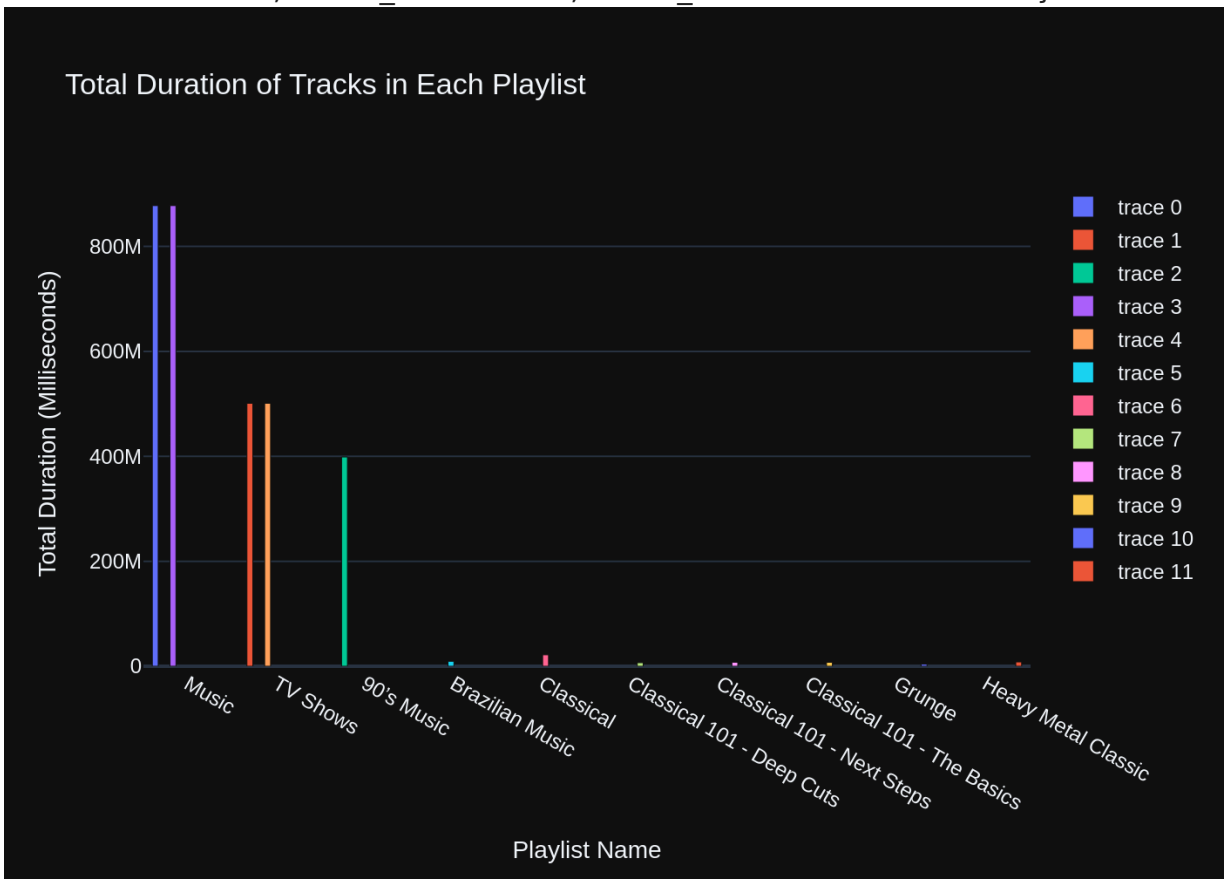
```
SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please
help to generate a SQL query to answer the question. Your response should ON
LY be based on the given context and follow the response guidelines and form
at instructions. \n===Tables \nCREATE INDEX IFK_PlaylistTrackTrackId ON "pla
ylist_track" (TrackId)\n\nCREATE TABLE "playlists"\r\n(\r\n    PlaylistId IN
TEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n
CREATE TABLE "playlist_track"\r\n(\r\n    PlaylistId INTEGER  NOT NULL,\r\n
TrackId INTEGER  NOT NULL,\r\n    CONSTRAINT PK_PlaylistTrack PRIMARY KEY
(PlaylistId, TrackId),\r\n    FOREIGN KEY (PlaylistId) REFERENCES "playlist
s" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FORE
IGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION
ON UPDATE NO ACTION\r\n)\n\nCREATE TABLE "tracks"\r\n(\r\n    TrackId INTEGE
R PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Name NVARCHAR(200)  NOT NULL,\r
\n    AlbumId INTEGER,\r\n    MediaTypeId INTEGER  NOT NULL,\r\n    GenreId
INTEGER,\r\n    Composer NVARCHAR(220),\r\n    Milliseconds INTEGER  NOT NUL
L,\r\n    Bytes INTEGER,\r\n    UnitPrice NUMERIC(10,2)  NOT NULL,\r\n    FO
REIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTIO
N ON UPDATE NO ACTION,\r\n    FOREIGN KEY (GenreId) REFERENCES "genres" (Gen
reId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n    FOREIGN KEY (M
ediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTI
ON ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (Ge
nreId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE INDE
X IFK_TrackMediaTypeId ON "tracks" (MediaTypeId)\n\nCREATE INDEX IFK_AlbumAr
tistId ON "albums" (ArtistId)\n\nCREATE TABLE "albums"\r\n(\r\n    AlbumId I
NTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n    Title NVARCHAR(160)  NOT N
ULL,\r\n    ArtistId INTEGER  NOT NULL,\r\n    FOREIGN KEY (ArtistId) REFERE
NCES "artists" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r
\n)\n\nCREATE TABLE "genres"\r\n(\r\n    GenreId INTEGER PRIMARY KEY AUTOINC
REMENT NOT NULL,\r\n    Name NVARCHAR(120)\r\n)\n\n\n===Additional Context
\n\nIn the chinook database invoice means order\n\n===Response Guidelines \n
1. If the provided context is sufficient, please generate a valid SQL query
without any explanations for the question. \n2. If the provided context is a
lmost sufficient but requires knowledge of a specific string in a particular
column, please generate an intermediate SQL query to find the distinct strin
gs in that column. Prepend the query with a comment saying intermediate_sql
\n3. If the provided context is insufficient, please explain why it can\'t b
e generated. \n4. Please use the most relevant table(s). \n5. If the questio
n has been asked and answered before, please repeat the answer exactly as it
was given before. \n'}, {'role': 'user', 'content': '  \n    List all genres
and the number of tracks in each genre:\n'}, {'role': 'assistant', 'conten
t': 'SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres
g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId'}, {'rol
e': 'user', 'content': ' \n    There are 3 tables: artists, albums and track
s, where albums and artists are linked by ArtistId, albums and tracks are li
nked by AlbumId,\n    Can you find the top 10 most popular artists based on
the number of tracks\n'}, {'role': 'assistant', 'content': 'SELECT ar.Name A
S ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN
albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId =
t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 10'},
{'role': 'user', 'content': '  \n    Find all tracks with a name containing
"What" (case-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \n
FROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content':
'  \n    Find the top 5 most expensive tracks (based on unit price):\n'},
{'role': 'assistant', 'content': 'SELECT Name, UnitPrice\nFROM tracks\nORDER
BY UnitPrice 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.CustomerId, c.FirstName, c.
LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c\nJOIN invo
ices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId
= ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMI
T 1'}, {'role': 'user', 'content': ' \n   Hint: album quantity is found in
invoice_items, \n   \n   Find the top 5 customers who bought the most albu
ms in total quantity (across all invoices):\n'}, {'role': 'assistant', 'cont
ent': 'SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS Tot
alAlbumQuantityBought\nFROM customers c\nJOIN invoices i ON c.CustomerId =
i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY
c.CustomerId\nORDER BY TotalAlbumQuantityBought DESC\nLIMIT 5'}, {'role': 'u
ser', 'content': ' \n   List all albums and their corresponding artist nam
es \n'}, {'role': 'assistant', 'content': 'SELECT a.Title AS AlbumTitle, a
r.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.Arti
stId'}, {'role': 'user', 'content': ' \n   Find all invoices since 2010 an
d the total amount invoiced:\n'}, {'role': 'assistant', 'content': "SELECT I
nvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {'rol
e': 'user', 'content': 'Can you list all tables in the SQLite database catal
og?'}, {'role': 'assistant', 'content': "SELECT name FROM sqlite_master WHER
E type='table'"}, {'role': 'user', 'content': ' \n   List all invoices wit
h a total exceeding $10:\n'}, {'role': 'assistant', 'content': 'SELECT * \nF
ROM invoices \nWHERE Total > 10'}, {'role': 'user', 'content': ' \n   Get
all playlists containing at least 10 tracks and the total duration of those
tracks:\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE INDEX IFK_PlaylistTrackTrackId ON \"playlist_track
\" (TrackId)\n\nCREATE TABLE \"playlists\"\r\n(\r\n   PlaylistId INTEGER PR
IMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE T
ABLE \"playlist_track\"\r\n(\r\n   PlaylistId INTEGER NOT NULL,\r\n   Tra
ckId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (Pla
ylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES \"playlists\"
(PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN
KEY (TrackId) REFERENCES \"tracks\" (TrackId) \r\n\t\tON DELETE NO ACTION ON
UPDATE NO ACTION\r\n)\n\nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER
PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n
AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (Genr
eId)\n\nCREATE INDEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE INDE
X IFK_TrackMediaTypeId ON \"tracks\" (MediaTypeId)\n\nCREATE INDEX IFK_Album
ArtistId ON \"albums\" (ArtistId)\n\nCREATE TABLE \"albums\"\r\n(\r\n   Alb
umId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NVARCHAR(160)
NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY (ArtistId) R
EFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO AC

TION\r\n)\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can't be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n"}, {"role": "user", "content": " \n   List all genres and the number of tracks in each genre:\n"}, {"role": "assistant", "content": "SELECT g.Name AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId"}, {"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 ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks 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 LOWER(Name) LIKE '%what%'"}, {"role": "user", "content": " \n   Find the top 5 most expensive tracks (based on unit price):\n"}, {"role": "assistant", "content": "SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice 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.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMIT 1"}, {"role": "user", "content": " \n   Hint: album quantity is found in invoice_items, \n   \n   Find the top 5 customers who bought the most albums in total quantity (across all invoices):\n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumQuantityBought DESC\nLIMIT 5"}, {"role": "user", "content": " \n   List all albums and their corresponding artist names \n"}, {"role": "assistant", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM albums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "content": " \n   Find all invoices since 2010 and the total amount invoiced:\n"}, {"role": "assistant", "content": "SELECT InvoiceDate, Total\nFROM invoices\nWHERE InvoiceDate >= '2010-01-01'"}, {"role": "user", "content": "Can you list all tables in the SQLite database catalog?"}, {"role": "assistant", "content": "SELECT name FROM sqlite_master WHERE type='table'"}, {"role": "user", "content": " \n   List all invoices with a total exceeding $10:\n"}, {"role": "assistant", "content": "SELECT * \nFROM invoices \nWHERE Total > 10"}, {"role": "user", "content": " \n   Get all playlists containing at least 10 tracks and the total duration of those tracks:\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:30:53.650867251Z', 'message': {'role': 'assistant', 'content': 'SELECT p.Name AS Play

```
listName, SUM(t.Milliseconds) AS TotalDuration\nFROM playlists p\nLEFT JOIN
playlist_track pt ON p.PlaylistId = pt.PlaylistId\nLEFT JOIN tracks t ON pt.
TrackId = t.TrackId\nGROUP BY p.PlaylistId\nHAVING COUNT(pt.TrackId) >= 1
0'}, 'done_reason': 'stop', 'done': True, 'total_duration': 48336966869, 'lo
ad_duration': 742547, 'prompt_eval_count': 1480, 'prompt_eval_duration': 413
95813000, 'eval_count': 76, 'eval_duration': 6253241000}
LLM Response: SELECT p.Name AS PlaylistName, SUM(t.Milliseconds) AS TotalDur
ation
FROM playlists p
LEFT JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId
LEFT JOIN tracks t ON pt.TrackId = t.TrackId
GROUP BY p.PlaylistId
HAVING COUNT(pt.TrackId) >= 10
SELECT p.Name AS PlaylistName, SUM(t.Milliseconds) AS TotalDuration
FROM playlists p
LEFT JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId
LEFT JOIN tracks t ON pt.TrackId = t.TrackId
GROUP BY p.PlaylistId
HAVING COUNT(pt.TrackId) >= 10
                  PlaylistName  TotalDuration
0                        Music      877683083
1                    TV Shows      501094957
2                  90’s Music      398705153
3                        Music      877683083
4                    TV Shows      501094957
5              Brazilian Music        9486559
6                    Classical       21770592
7    Classical 101 - Deep Cuts        6755730
8   Classical 101 - Next Steps        7575051
9   Classical 101 - The Basics        7439811
10                      Grunge        4122018
11         Heavy Metal Classic        8206312
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "The following is a pandas DataFrame that con
tains the results of the query that answers the question the user asked: '
\n     Get all playlists containing at least 10 tracks and the total duratio
n of those tracks:\n'\n\nThe DataFrame was produced using this query: SELECT
p.Name AS PlaylistName, SUM(t.Milliseconds) AS TotalDuration\nFROM playlists
p\nLEFT JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId\nLEFT JOIN tr
acks t ON pt.TrackId = t.TrackId\nGROUP BY p.PlaylistId\nHAVING COUNT(pt.Tra
ckId) >= 10\n\nThe following is information about the resulting pandas DataF
rame 'df': \nRunning df.dtypes gives:\n PlaylistName     object\nTotalDurati
on     int64\ndtype: object"}, {"role": "user", "content": "Can you generate
the Python plotly code to chart the results of the dataframe? Assume the dat
a is in a pandas dataframe called 'df'. If there is only one value in the da
taframe, use an Indicator. Respond with only Python code. Do not answer with
any explanations -- just the code."}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:31:11.866
772841Z', 'message': {'role': 'assistant', 'content': ' ```python\nimport pl
otly.graph_objects as go\nimport pandas as pd\n\n# Assuming df is your DataF
rame\nif len(df) == 1:\n    fig = go.Figure(go.Indicator(\n        mode="num
```

ber",\n value=df[\'TotalDuration\'].iloc[0],\n title={"text": f"Total Duration of Tracks in Playlist: {df[\'PlaylistName\'].iloc[0]}"}\n))\nelse:\n fig = go.Figure()\n for index, row in df.iterrows():\n fig.add_trace(go.Bar(x=[row[\'PlaylistName\']], y=[row[\'TotalDuration\']]))\n\nfig.update_layout(title="Total Duration of Tracks in Each Playlist", xaxis_title="Playlist Name", yaxis_title="Total Duration (Milliseconds)")\nfig.show()\n```'}, 'done_reason': 'stop', 'done': True, 'total_duration': 18189503595, 'load_duration': 665710, 'prompt_eval_count': 237, 'prompt_eval_duration': 4908230000, 'eval_count': 194, 'eval_duration': 13148972000}



Out[38]:
```
('SELECT p.Name AS PlaylistName, SUM(t.Milliseconds) AS TotalDuration\nFROM
playlists p\nLEFT JOIN playlist_track pt ON p.PlaylistId = pt.PlaylistId\nL
EFT JOIN tracks t ON pt.TrackId = t.TrackId\nGROUP BY p.PlaylistId\nHAVING
COUNT(pt.TrackId) >= 10',
                   PlaylistName  TotalDuration
 0                         Music      877683083
 1                     TV Shows      501094957
 2                   90’s Music      398705153
 3                         Music      877683083
 4                     TV Shows      501094957
 5               Brazilian Music        9486559
 6                     Classical       21770592
 7    Classical 101 - Deep Cuts        6755730
 8   Classical 101 - Next Steps        7575051
 9   Classical 101 - The Basics        7439811
 10                       Grunge        4122018
 11        Heavy Metal Classic        8206312,
 Figure({
     'data': [{'type': 'bar', 'x': ['Music'], 'y': [877683083]},
              {'type': 'bar', 'x': ['TV Shows'], 'y': [501094957]},
              {'type': 'bar', 'x': ['90’s Music'], 'y': [398705153]},
              {'type': 'bar', 'x': ['Music'], 'y': [877683083]},
              {'type': 'bar', 'x': ['TV Shows'], 'y': [501094957]},
              {'type': 'bar', 'x': ['Brazilian Music'], 'y': [9486559]},
              {'type': 'bar', 'x': ['Classical'], 'y': [21770592]},
              {'type': 'bar', 'x': ['Classical 101 - Deep Cuts'], 'y': [675
5730]},
              {'type': 'bar', 'x': ['Classical 101 - Next Steps'], 'y': [75
75051]},
              {'type': 'bar', 'x': ['Classical 101 - The Basics'], 'y': [74
39811]},
              {'type': 'bar', 'x': ['Grunge'], 'y': [4122018]},
              {'type': 'bar', 'x': ['Heavy Metal Classic'], 'y': [820631
2]}],
     'layout': {'template': '...',
                'title': {'text': 'Total Duration of Tracks in Each Playlis
t'},
                'xaxis': {'title': {'text': 'Playlist Name'}},
                'yaxis': {'title': {'text': 'Total Duration (Millisecond
s)'}}}
 }))
```

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


"""

vn.ask(question=question)
```

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

SQL Prompt: [{'role': 'system', 'content': 'You are a SQLite expert. Please help to generate a SQL query to answer the question. Your response should ONLY be based on the given context and follow the response guidelines and format instructions. \n===Tables \nCREATE TABLE "tracks"\r\n(\r\n   TrackId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   AlbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGER,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n   Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY (AlbumId) REFERENCES "albums" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES "genres" (GenreId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTypeId) REFERENCES "media_types" (MediaTypeId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON "albums" (ArtistId)\n\nCREATE INDEX IFK_TrackGenreId ON "tracks" (GenreId)\n\nCREATE INDEX IFK_TrackAlbumId ON "tracks" (AlbumId)\n\nCREATE TABLE "albums"\r\n(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title 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_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 "playlist_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 NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (PlaylistId) REFERENCES "playlists" (PlaylistId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES "tracks" (TrackId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional Context \n\nIn the chinook database invoice means order\n\n===Response Guidelines \n1. If the provided context is sufficient, please generate a valid SQL query without any explanations for the question. \n2. If the provided context is almost sufficient but requires knowledge of a specific string in a particular column, please generate an intermediate SQL query to find the distinct strings in that column. Prepend the query with a comment saying intermediate_sql \n3. If the provided context is insufficient, please explain why it can\'t be generated. \n4. Please use the most relevant table(s). \n5. If the question has been asked and answered before, please repeat the answer exactly as it was given before. \n'}, {'role': 'user', 'content': ' \n   There are 3 tables: artists, albums and tracks, where albums and artists are linked by ArtistId, albums and tracks are linked by AlbumId,\n   Can you find the top 10 most popular artists based on the number of tracks\n'}, {'role': 'assistant', 'content': 'SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS NumberOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER BY NumberOfTracks DESC\nLIMIT 10'}, {'role': 'user', 'content': ' \n   List all albums and their corresponding artist names \n'}, {'role': 'assistant', 'content': 'SELECT a.Title AS AlbumTitle, 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 AS Genre, COUNT(t.TrackId) AS NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId\nGROUP BY g.GenreId'}, {'role': 'user', 'content': ' \n   Get all playlists containing at least 10 tracks and the total duration of those tracks:\n'}, {'role': 'assistant', 'content': 'SELECT p.Name AS PlaylistName, SUM(t.Milliseconds) AS TotalDuration\nFROM playlists p\nLEFT JOIN playlist_track pt ON p.PlaylistId

= pt.PlaylistId\nLEFT JOIN tracks t ON pt.TrackId = t.TrackId\nGROUP BY p.Pl
aylistId\nHAVING COUNT(pt.TrackId) >= 10'}, {'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 invo
ices):\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstNa
me, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought\nFROM customers
c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON
i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumQuanti
tyBought DESC\nLIMIT 5'}, {'role': 'user', 'content': ' \n   Find the cus
tomer who bought the most albums in total quantity (across all invoices):
\n'}, {'role': 'assistant', 'content': 'SELECT c.CustomerId, c.FirstName, c.
LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c\nJOIN invo
ices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON i.InvoiceId
= ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBought DESC\nLIMI
T 1'}, {'role': 'user', 'content': ' \n   Find the top 5 most expensive tr
acks (based on unit price):\n'}, {'role': 'assistant', 'content': 'SELECT Na
me, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT 5'}, {'role': 'us
er', 'content': ' \n   Find all tracks with a name containing "What" (case
-insensitive)\n'}, {'role': 'assistant', 'content': "SELECT * \nFROM tracks
\nWHERE LOWER(Name) LIKE '%what%'"}, {'role': 'user', 'content': 'Can you li
st all tables in the SQLite database catalog?'}, {'role': 'assistant', 'cont
ent': "SELECT name FROM sqlite_master WHERE type='table'"}, {'role': 'user',
'content': 'How many customers are there'}, {'role': 'assistant', 'content':
'SELECT COUNT(*) AS NumberOfCustomers FROM customers'}, {'role': 'user', 'co
ntent': ' \n   Identify artists who have albums with tracks appearing in
multiple genres:\n\n\n'}]
Info: Ollama parameters:
model=deepseek-coder-v2:latest,
options={},
keep_alive=None
Info: Prompt Content:
[{"role": "system", "content": "You are a SQLite expert. Please help to gene
rate a SQL query to answer the question. Your response should ONLY be based
on the given context and follow the response guidelines and format instructi
ons. \n===Tables \nCREATE TABLE \"tracks\"\r\n(\r\n   TrackId INTEGER PRIMA
RY KEY AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(200) NOT NULL,\r\n   A
lbumId INTEGER,\r\n   MediaTypeId INTEGER NOT NULL,\r\n   GenreId INTEGE
R,\r\n   Composer NVARCHAR(220),\r\n   Milliseconds INTEGER NOT NULL,\r\n
Bytes INTEGER,\r\n   UnitPrice NUMERIC(10,2) NOT NULL,\r\n   FOREIGN KEY
(AlbumId) REFERENCES \"albums\" (AlbumId) \r\n\t\tON DELETE NO ACTION ON UPD
ATE NO ACTION,\r\n   FOREIGN KEY (GenreId) REFERENCES \"genres\" (GenreId)
\r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION,\r\n   FOREIGN KEY (MediaTy
peId) REFERENCES \"media_types\" (MediaTypeId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_AlbumArtistId ON \"albums\" (Art
istId)\n\nCREATE INDEX IFK_TrackGenreId ON \"tracks\" (GenreId)\n\nCREATE IN
DEX IFK_TrackAlbumId ON \"tracks\" (AlbumId)\n\nCREATE TABLE \"albums\"\r\n
(\r\n   AlbumId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n   Title NV
ARCHAR(160) NOT NULL,\r\n   ArtistId INTEGER NOT NULL,\r\n   FOREIGN KEY
(ArtistId) REFERENCES \"artists\" (ArtistId) \r\n\t\tON DELETE NO ACTION ON
UPDATE NO ACTION\r\n)\n\nCREATE INDEX IFK_TrackMediaTypeId ON \"tracks\" (Me
diaTypeId)\n\nCREATE TABLE \"genres\"\r\n(\r\n   GenreId INTEGER PRIMARY KE
Y AUTOINCREMENT NOT NULL,\r\n   Name NVARCHAR(120)\r\n)\n\nCREATE INDEX IFK
_PlaylistTrackTrackId ON \"playlist_track\" (TrackId)\n\nCREATE TABLE \"arti
sts\"\r\n(\r\n   ArtistId INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,\r\n
Name NVARCHAR(120)\r\n)\n\nCREATE TABLE \"playlist_track\"\r\n(\r\n   Playl
istId INTEGER NOT NULL,\r\n   TrackId INTEGER NOT NULL,\r\n   CONSTRAINT

PK_PlaylistTrack PRIMARY KEY (PlaylistId, TrackId),\r\n   FOREIGN KEY (Pla
ylistId) REFERENCES \"playlists\" (PlaylistId) \r\n\t\tON DELETE NO ACTION O
N UPDATE NO ACTION,\r\n   FOREIGN KEY (TrackId) REFERENCES \"tracks\" (Trac
kId) \r\n\t\tON DELETE NO ACTION ON UPDATE NO ACTION\r\n)\n\n\n===Additional
Context \n\nIn the chinook database invoice means order\n\n===Response Guide
lines \n1. If the provided context is sufficient, please generate a valid SQ
L query without any explanations for the question. \n2. If the provided cont
ext is almost sufficient but requires knowledge of a specific string in a pa
rticular column, please generate an intermediate SQL query to find the disti
nct strings in that column. Prepend the query with a comment saying intermed
iate_sql \n3. If the provided context is insufficient, please explain why it
can't be generated. \n4. Please use the most relevant table(s). \n5. If the
question has been asked and answered before, please repeat the answer exactl
y as it was given before. \n"}, {"role": "user", "content": " \n   There ar
e 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": "ass
istant", "content": "SELECT ar.Name AS ArtistName, COUNT(t.TrackId) AS Numbe
rOfTracks\nFROM artists ar\nLEFT JOIN albums al ON ar.ArtistId = al.ArtistId
\nLEFT JOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP BY ar.ArtistId\nORDER
BY NumberOfTracks DESC\nLIMIT 10"}, {"role": "user", "content": " \n   Lis
t all albums and their corresponding artist names \n"}, {"role": "assistan
t", "content": "SELECT a.Title AS AlbumTitle, ar.Name AS ArtistName\nFROM al
bums a\nJOIN artists ar ON a.ArtistId = ar.ArtistId"}, {"role": "user", "con
tent": " \n   List all genres and the number of tracks in each genre:\n"},
{"role": "assistant", "content": "SELECT g.Name AS Genre, COUNT(t.TrackId) A
S NumberOfTracks\nFROM genres g\nLEFT JOIN tracks t ON g.GenreId = t.GenreId
\nGROUP BY g.GenreId"}, {"role": "user", "content": " \n   Get all playli
sts containing at least 10 tracks and the total duration of those track
s:\n"}, {"role": "assistant", "content": "SELECT p.Name AS PlaylistName, SUM
(t.Milliseconds) AS TotalDuration\nFROM playlists p\nLEFT JOIN playlist_trac
k pt ON p.PlaylistId = pt.PlaylistId\nLEFT JOIN tracks t ON pt.TrackId = t.T
rackId\nGROUP BY p.PlaylistId\nHAVING COUNT(pt.TrackId) >= 10"}, {"role": "u
ser", "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.Custo
merId, c.FirstName, c.LastName, SUM(ii.Quantity) AS TotalAlbumQuantityBought
\nFROM customers c\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN inv
oice_items ii ON i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY
TotalAlbumQuantityBought DESC\nLIMIT 5"}, {"role": "user", "content": " \n
Find the customer who bought the most albums in total quantity (across all i
nvoices): \n"}, {"role": "assistant", "content": "SELECT c.CustomerId, c.Fir
stName, c.LastName, SUM(ii.Quantity) AS TotalAlbumsBought\nFROM customers c
\nJOIN invoices i ON c.CustomerId = i.CustomerId\nJOIN invoice_items ii ON
i.InvoiceId = ii.InvoiceId\nGROUP BY c.CustomerId\nORDER BY TotalAlbumsBough
t DESC\nLIMIT 1"}, {"role": "user", "content": " \n   Find the top 5 most
expensive tracks (based on unit price):\n"}, {"role": "assistant", "conten
t": "SELECT Name, UnitPrice\nFROM tracks\nORDER BY UnitPrice DESC\nLIMIT
5"}, {"role": "user", "content": " \n   Find all tracks with a name contai
ning \"What\" (case-insensitive)\n"}, {"role": "assistant", "content": "SELE
CT * \nFROM tracks \nWHERE LOWER(Name) LIKE '%what%'"}, {"role": "user", "co
ntent": "Can you list all tables in the SQLite database catalog?"}, {"role":
"assistant", "content": "SELECT name FROM sqlite_master WHERE type='tabl
e'"}, {"role": "user", "content": "How many customers are there"}, {"role":
"assistant", "content": "SELECT COUNT(*) AS NumberOfCustomers FROM customer
s"}, {"role": "user", "content": " \n   Identify artists who have albums

```
with tracks appearing in multiple genres:\n\n\n"}]
Info: Ollama Response:
{'model': 'deepseek-coder-v2:latest', 'created_at': '2024-07-30T03:32:02.664
38191Z', 'message': {'role': 'assistant', 'content': ' intermediate_sql\n```
sql\nSELECT ar.ArtistId, ar.Name, t.GenreId\nFROM artists ar\nJOIN albums al
ON ar.ArtistId = al.ArtistId\nJOIN tracks t ON al.AlbumId = t.AlbumId\nGROUP
BY ar.ArtistId, t.TrackId\nHAVING COUNT(DISTINCT t.GenreId) > 1;\n```'}, 'do
ne_reason': 'stop', 'done': True, 'total_duration': 50681666431, 'load_durat
ion': 760308, 'prompt_eval_count': 1512, 'prompt_eval_duration': 4286646700
0, 'eval_count': 84, 'eval_duration': 7136341000}
LLM Response:  intermediate_sql
```sql
SELECT ar.ArtistId, ar.Name, t.GenreId
FROM artists ar
JOIN albums al ON ar.ArtistId = al.ArtistId
JOIN tracks t ON al.AlbumId = t.AlbumId
GROUP BY ar.ArtistId, t.TrackId
HAVING COUNT(DISTINCT t.GenreId) > 1;
```
The LLM is not allowed to see the data in your database. Your question requi
res database introspection to generate the necessary SQL. Please set allow_l
lm_to_see_data=True to enable this.
Couldn't run sql: Execution failed on sql 'The LLM is not allowed to see th
e data in your database. Your question requires database introspection to ge
nerate the necessary SQL. Please set allow_llm_to_see_data=True to enable th
is.': near "The": syntax error
```

Check completion time

In [ ]:

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

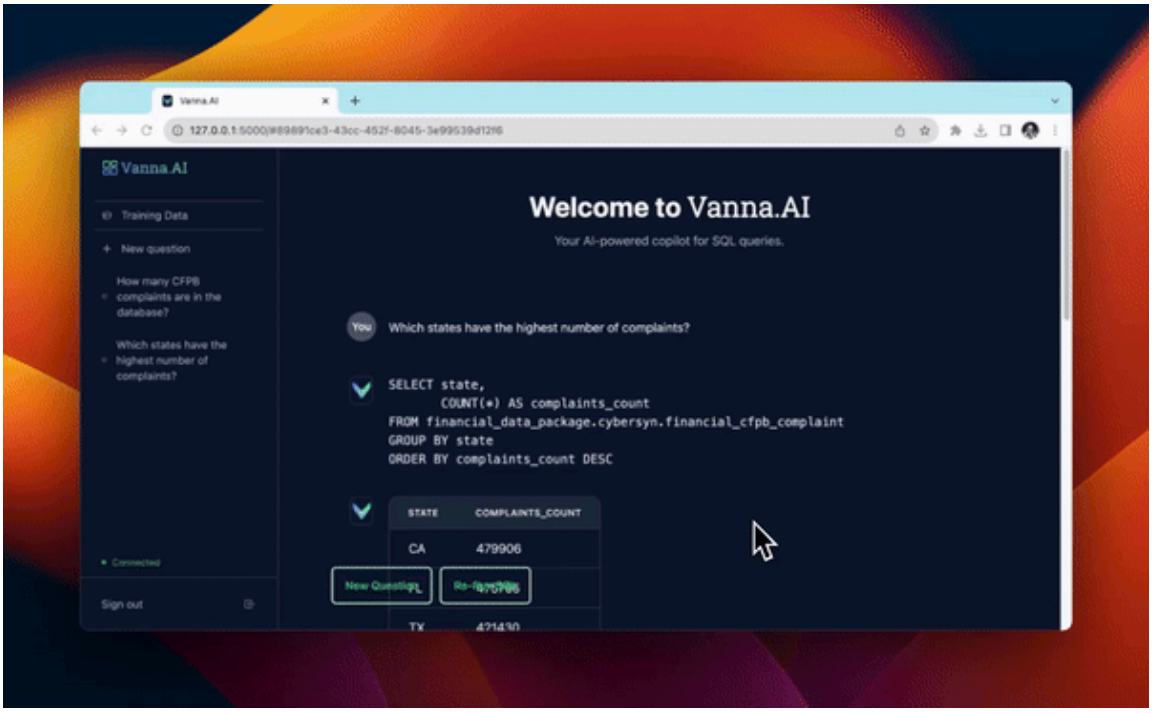
elapsed_time = ts_stop - ts_start
print(f"test running on '{hostname}' with '{model_name}' LLM took : {elapsed
```

```
test running on 'ducklover1' with 'deepseek-coder-v2' LLM took : 1303.60 sec
```

In [41]:
```
from datetime import datetime
print(datetime.now())
```

```
2024-07-29 23:32:02.679405
```

Launch the User Interface



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

Next Steps

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

- Streamlit app
- Flask app
- Slackbot