

[Home](#) [Modules](#) [Agents](#) [How-to](#) [Use ToolKits with OpenAI Functions](#)

Use ToolKits with OpenAI Functions

This notebook shows how to use the OpenAI functions agent with arbitrary toolkits.

```
from langchain import (
    LLMChain,
    OpenAI,
    SerpAPIWrapper,
    SQLDatabase,
    SQLDatabaseChain,
)
from langchain.agents import initialize_agent, Tool
from langchain.agents import AgentType
from langchain.chat_models import ChatOpenAI
from langchain.agents.agent_toolkits import SQLDatabaseToolkit
from langchain.schema import SystemMessage
```

Load the toolkit

```
db = SQLDatabase.from_uri("sqlite:///../../../../../notebooks/Chinook.db")
toolkit = SQLDatabaseToolkit(llm=ChatOpenAI(), db=db)
```



Set a system message specific to that toolkit

```
agent_kwargs = {  
    "system_message": SystemMessage(content="You are an expert SQL data analyst.")  
}
```

```
llm = ChatOpenAI(temperature=0, model="gpt-3.5-turbo-0613")  
agent = initialize_agent(  
    toolkit.get_tools(),  
    llm,  
    agent=AgentType.OPENAI_FUNCTIONS,  
    verbose=True,  
    agent_kwargs=agent_kwargs,  
)
```

```
agent.run("how many different artists are there?")
```

```
> Entering new chain...
```

```
Invoking: `sql_db_query` with `{'query': 'SELECT COUNT(DISTINCT artist_name) AS num_artists FROM  
artists'}`
```

```
Error: (sqlite3.OperationalError) no such table: artists  
[SQL: SELECT COUNT(DISTINCT artist_name) AS num_artists FROM artists]  
(Background on this error at: https://sqlalche.me/e/20/e3q8)  
Invoking: `sql_db_list_tables` with `{}`
```

```
MediaType, Track, Playlist, sales_table, Customer, Genre, PlaylistTrack, Artist, Invoice, Album, InvoiceLine, Employee
```

```
Invoking: `sql_db_query` with `{'query': 'SELECT COUNT(DISTINCT artist_id) AS num_artists FROM Artist'}`
```

```
Error: (sqlite3.OperationalError) no such column: artist_id
```

```
[SQL: SELECT COUNT(DISTINCT artist_id) AS num_artists FROM Artist]
```

```
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

```
Invoking: `sql_db_query` with `{'query': 'SELECT COUNT(DISTINCT Name) AS num_artists FROM Artist'}`
```

```
[(275,)]There are 275 different artists in the database.
```

```
> Finished chain.
```

```
'There are 275 different artists in the database.'
```