OPENPYXL & PYODBC

步骤1：安装openpyxl和pyodbc

在开始之前，我们需要先安装openpyxl和pyodbc库。可以使用以下命令在命令行中安装：

```
pip install openpyxl pyodbc
```

步骤2：连接数据库并执行SQL查询

首先，我们需要连接到数据库并执行SQL查询。我们可以使用上面提供的`execute_sql_statement`函数来执行SQL查询。以下是一个示例查询：

```python
import pyodbc

def execute_sql_statement(sql_statement, connection_dict):
    """
    Executes an SQL statement and returns the results.

    :param sql_statement: The SQL statement to execute.
    :param connection_dict: A dictionary containing the connection details for the database.
    :return: A list of tuples containing the query results.
    """
    # Construct the connection string from the dictionary
    connection_string = (
        f"DRIVER={{{connection_dict['driver']}}};"
        f"SERVER={connection_dict['server']};"
        f"DATABASE={connection_dict['database']};"
        f"UID={connection_dict['username']};"
        f"PWD={connection_dict['password']};"
    )

    # Connect to the database using the connection string
    conn = pyodbc.connect(connection_string)

    # Create a cursor object to execute the SQL statement
    cursor = conn.cursor()

    # Execute the SQL statement
    cursor.execute(sql_statement)

    # Fetch the query results
    query_results = cursor.fetchall()

    # Close the cursor and connection
    cursor.close()
    conn.close()

    # Return the query results
    return query_results

# Define the SQL query to execute
sql_query = "SELECT * FROM students"

# Execute the SQL query and get the results
results = execute_sql_statement(sql_query, connection_dict)

# Print the results to verify that the query was successful
print(results)
```

步骤3：创建电子表格文件并写入数据

接下来，我们需要创建一个新的电子表格文件并将查询结果写入其中。我们可以使用openpyxl库来创建和操作电子表格文件。以下是一个示例代码：

```
from openpyxl import Workbook

# Create a new workbook
workbook = Workbook()

# Select the active worksheet
worksheet = workbook.active

# Define the headers for the worksheet
headers = ["ID", "Name", "Age", "Gender"]
```

```python
# Write the headers to the worksheet
worksheet.append(headers)

# Write the query results to the worksheet
for row in results:
    worksheet.append(row)

# Save the workbook
workbook.save("students.xlsx")
```

我们首先创建了一个新的电子表格文件，然后选择了活动工作表。接下来，我们定义了工作表的标题，并将其写入工作表中。最后，我们将查询结果写入工作表中，并将工作表保存为一个名为“students.xlsx”的文件。

步骤4：完整代码演示

下面是完整的代码演示，包括连接到数据库、执行SQL查询、创建电子表格文件并写入数据的所有步骤：

```python
import pyodbc
from openpyxl import Workbook

# Define the connection details for the database
connection_dict = {
    "driver": "SQL Server",
    "server": "localhost",
    "database": "xhedu",
    "username": "sa",
    "password": "123456",
}

def execute_sql_statement(sql_statement, connection_dict):
    """
    Executes an SQL statement and returns the results.

    :param sql_statement: The SQL statement to execute.
    :param connection_dict: A dictionary containing the connection details for the database.
    :return: A list of tuples containing the query results.
    """
    # Construct the connection string from the dictionary
    connection_string = (
        f"DRIVER={{{connection_dict['driver']}}};"
        f"SERVER={connection_dict['server']};"
        f"DATABASE={connection_dict['database']};"
        f"UID={connection_dict['username']};"
        f"PWD={connection_dict['password']};"
    )

    # Connect to the database using the connection string
    conn = pyodbc.connect(connection_string)

    # Create a cursor object to execute the SQL statement
    cursor = conn.cursor()

    # Execute the SQL statement
    cursor.execute(sql_statement)

    # Fetch the query results
    query_results = cursor.fetchall()

    # Close the cursor and connection
    cursor.close()
    conn.close()

    # Return the query results
    return query_results

# Define the SQL query to execute
sql_query = "SELECT * FROM students"

# Execute the SQL query and get the results
results = execute_sql_statement(sql_query, connection_dict)

# Create a new workbook
workbook = Workbook()

# Select the active worksheet
worksheet = workbook.active
```

```python
# Define the headers for the worksheet
headers = ["ID", "Name", "Age", "Gender"]

# Write the headers to the worksheet
worksheet.append(headers)

# Write the query results to the worksheet
for row in results:
    worksheet.append(row)

# Save the workbook
workbook.save("students.xlsx")
```

这个示例代码将查询结果写入了一个名为“students.xlsx”的电子表格文件中。你可以根据自己的需要修改代码，以适应不同的查询和电子表格文件格式。