1. Set the variable test1 to the string 'This is a test of the emergency text system,' and save test1 to a file named test.txt.

2. Read the contents of the file test.txt into the variable test2. Is there a difference between test 1 and test 2?

3. Create a CSV file called books.csv by using these lines:

title,author,year

The Weirdstone of Brisingamen,Alan Garner,1960

Perdido Street Station,China Miéville,2000

Thud!,Terry Pratchett,2005

The Spellman Files,Lisa Lutz,2007

Small Gods,Terry Pratchett,1992

4. Use the sqlite3 module to create a SQLite database called books.db, and a table called books with these fields: title (text), author (text), and year (integer).

5. Read books.csv and insert its data into the book table.

6. Select and print the title column from the book table in alphabetical order.

7. From the book table, select and print all columns in the order of publication.

8. Use the sqlalchemy module to connect to the sqlite3 database books.db that you just made in exercise 6.

9. Install the Redis server and the Python redis library (pip install redis) on your computer. Create a Redis hash called test with the fields count (1) and name ('Fester Bestertester'). Print all the fields for test.

10. Increment the count field of test and print it.

Answers

1. **Create a Text File and Write to It**:
   * We’ll set the variable test1 to the given string and save it to a file named test.txt.

**Python**

test1 = 'This is a test of the emergency text system.'

with open('test.txt', 'w') as file:

file.write(test1)

print(f"File 'test.txt' created and written successfully.")

1. **Read Contents of the File into Another Variable**:
   * We’ll read the contents of the file test.txt into the variable test2.

**Python**

with open('test.txt', 'r') as file:

test2 = file.read()

# Check if there's any difference between test1 and test2

if test1 == test2:

print("Contents of test1 and test2 are the same.")

else:

print("Contents of test1 and test2 are different.")

1. **Create a CSV File Called books.csv**:
   * We’ll create a CSV file named books.csv with the given data.

**Python**

import csv

data = [

['title', 'author', 'year'],

['The Weirdstone of Brisingamen', 'Alan Garner', 1960],

['Perdido Street Station', 'China Miéville', 2000],

['Thud!', 'Terry Pratchett', 2005],

['The Spellman Files', 'Lisa Lutz', 2007],

['Small Gods', 'Terry Pratchett', 1992]

]

with open('books.csv', 'w', newline='') as file:

writer = csv.writer(file)

writer.writerows(data)

print("CSV file 'books.csv' created successfully.")

1. **Create an SQLite Database and Table**:
   * We’ll create an SQLite database called books.db and a table called books with the specified fields.

**Python**

import sqlite3

conn = sqlite3.connect('books.db')

cursor = conn.cursor()

cursor.execute('''

CREATE TABLE IF NOT EXISTS books (

title TEXT,

author TEXT,

year INTEGER

)

''')

conn.commit()

conn.close()

print("SQLite database 'books.db' created successfully.")

1. **Read books.csv and Insert Data into the Book Table**:
   * We’ll read the data from books.csv and insert it into the books table.

**Python**

with open('books.csv', 'r') as file:

reader = csv.reader(file)

next(reader) # Skip the header row

for row in reader:

cursor.execute('INSERT INTO books VALUES (?, ?, ?)', row)

conn.commit()

print("Data inserted into the 'books' table.")

1. **Select and Print the Title Column in Alphabetical Order**:
   * We’ll select and print the title column from the books table in alphabetical order.

**Python**

cursor.execute('SELECT title FROM books ORDER BY title')

for row in cursor.fetchall():

print(row[0])

1. **Select and Print All Columns in the Order of Publication**:
   * We’ll select and print all columns from the books table.

**Python**

cursor.execute('SELECT \* FROM books')

for row in cursor.fetchall():

print(row)

1. **Use SQLAlchemy to Connect to the SQLite Database**:
   * We’ll use SQLAlchemy to connect to the books.db database.

**Python**

from sqlalchemy import create\_engine

engine = create\_engine('sqlite:///books.db')

connection = engine.connect()

print("Connected to the SQLite database using SQLAlchemy.")

1. **Create a Redis Hash and Print Its Fields**:
   * We’ll create a Redis hash called test with the specified fields and print all the fields.

**Python**

import redis

r = redis.Redis(host='localhost', port=6379, db=0)

r.hset('test', 'count', 1)

r.hset('test', 'name', 'Fester Bestertester')

fields = r.hgetall('test')

print("Fields for 'test':")

for key, value in fields.items():

print(f"{key.decode('utf-8')}: {value.decode('utf-8')}")

1. **Increment the Count Field of the Redis Hash**:
   * We’ll increment the count field of the test hash and print its updated value.

**Python**

r.hincrby('test', 'count', 1)

updated\_count