**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.**

Ans.

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

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

file.write(test1)

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

Ans.

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

test2 = file.read()

print(test1 == test2)

There is no difference between test1 and test2, so the comparison test1 == test2 evaluates to True

**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).**

Ans.

import sqlite3

# Connect to the database

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

cursor = conn.cursor()

# Create the 'books' table

cursor.execute('''

CREATE TABLE IF NOT EXISTS books (

title TEXT,

author TEXT,

year INTEGER

)

''')

# Commit changes and close the connection

conn.commit()

conn.close()

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

**Ans.**

import csv

import sqlite3

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

csv\_reader = csv.reader(file)

next(csv\_reader) # Skip the header row

# Connect to the database

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

cursor = conn.cursor()

# Insert data into the 'books' table

for row in csv\_reader:

title, author, year = row

cursor.execute('INSERT INTO books (title, author, year) VALUES (?, ?, ?)', (title, author, int(year)))

# Commit changes and close the connection

conn.commit()

conn.close()

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

Ans.

import sqlite3

# Connect to the database

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

cursor = conn.cursor()

# Select and print the 'title' column from the 'books' table in alphabetical order

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

for row in cursor.fetchall():

print(row[0])

# Close the connection

conn.close()

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

**Ans.**

import sqlite3

# Connect to the database

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

cursor = conn.cursor()

# Select and print all columns from the 'books' table in the order of publication

cursor.execute('SELECT title, author, year FROM books')

for row in cursor.fetchall():

print(row)

# Close the connection

conn.close()

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

**Ans.**

# Step 8

from sqlalchemy import create\_engine

# Connect to the database using the sqlalchemy engine

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

**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.**

**Ans.**

import redis

# Connect to the Redis server

redis\_client = redis.StrictRedis(host='localhost', port=6379, db=0)

# Create a Redis hash called 'test' with the fields 'count' (1) and 'name' ('Fester Bestertester')

redis\_client.hmset('test', {'count': 1, 'name': 'Fester Bestertester'})

# Print all the fields for 'test'

print(redis\_client.hgetall('test'))

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

**Ans.**

# Increment the 'count' field of 'test'

redis\_client.hincrby('test', 'count', 1)

# Print the updated 'count' field of 'test'

print(redis\_client.hget('test', 'count'))