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.

Ans1

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?

Ans2

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

test2 = file.read()

print(test1)

print(test2)

print(test1 == test2)

output

This is a test of the emergency text system

This is a test of the emergency text system

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

Ans3

import csv

header = ['title', 'author', 'year']

data = [

['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.writerow(header)

for book in data:

writer.writerow(book)

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).

Ans4

import sqlite3

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

conn.execute('''CREATE TABLE books

(title TEXT, author TEXT, year INTEGER)''')

conn.commit()

conn.close()

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

Ans5

import csv

import sqlite3

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

cursor = conn.cursor()

cursor.execute('''CREATE TABLE IF NOT EXISTS books

(title TEXT, author TEXT, year INTEGER)''')

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

reader = csv.DictReader(csvfile)

for row in reader:

title = row['title']

author = row['author']

year = int(row['year'])

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

conn.commit()

conn.close()

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

Ans6

import sqlite3

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

cursor = conn.cursor()

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

for row in cursor.fetchall():

print(row[0])

conn.close()

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

Ans7

import sqlite3

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

cursor = conn.cursor()

cursor.execute("SELECT title, author, year FROM books ORDER BY year")

for row in cursor.fetchall():

print(row)

conn.close()

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

Ans8

from sqlalchemy import create\_engine

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

result = engine.execute('SELECT \* FROM books')

for row in result:

print(row)

engine.dispose()

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.

Ans9

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')

for key, value in fields.items():

print(f'{key.decode()}: {value.decode()}')

r.close()

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

Ans10

import redis

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

new\_count = r.hincrby('test', 'count', 1)

print(f"New count: {new\_count}")

r.close()