Assignment-20

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(f"test1: {test1}")
print(f"test2: {test2}")
print(f"Are they equal? {test1 == test2}")
```

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

Ans.

```
import csv

books_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(books_data)
```

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

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

cursor.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.

Ans.

```
import sqlite3
import csv

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

with open('books.csv', 'r') as file:
    csv_reader = csv.reader(file)
    next(csv_reader)  # Skip header row
    for row in csv_reader:
        cursor.execute('INSERT INTO books VALUES (?, ?, ?)', (row[0], row[1], int(row[2])))

conn.commit()
conn.close()
```

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

Ans.

```
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.

Ans.

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


cursor.execute('SELECT * FROM books ORDER BY year')
for row in cursor.fetchall():
   print(f"{row[0]} by {row[1]} ({row[2]})")


conn.close()
```

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

Ans.

```
from sqlalchemy import create_engine, text

engine = create_engine('sqlite:///books.db')

with engine.connect() as connection:
   result = connection.execute(text('SELECT * FROM books'))
   for row in result:
      print(row)
```

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

r = redis.Redis(host='localhost', port=6379, decode_responses=True)
r.hset('test', mapping={'count': 1, 'name': 'Fester Bestertester'})

# Print all fields
print(r.hgetall('test'))
```

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

Ans.

```
r.hincrby('test', 'count', 1)
print(f"Updated count: {r.hget('test', 'count')}")
```