Sabanci University
2023-2024 Spring
CS306 Group Project
Phase3

TeamUp Music Streaming Platform

Osman Şah Yılmaz, 31316

Description

In this project, I have developed a Python application that interacts with a database to perform CRUD (Create, Read, Update, Delete) operations. The application focuses on two tables: Artists and Songs_Belong_to. Each operation is implemented as a separate function to handle the specific requirements of these tables. The project includes inserting new records, retrieving existing records, updating records, and deleting records. The main goal is to demonstrate the ability to manipulate database data using Python while preventing SQL injection attacks through parameterized queries.

NOTE: We cannot create 2 tables and insert them because it references the 3rd or 4th table with the id. That's why we can't do it without using common tables. That's why we created the other tables required for the two different tables we chose, but everyone executed the steps from the two main tables they chose. Yücel Saygın hoca knows the situation and confirmed it.

Chosen Tables and Data

Artists Table:

Columns: artist_id, name, biography, genre, monthly_listening

Songs_Belong_to Table:

Columns: sid, length, sname, art_id, gid

Sample Data of Artists Table:

art_id	name	biography	monthly_listening
20001	Canozan	canozan@cnzn	11111
20002	Teoman	teoman@tmn	22222
20003	Pinhani	pinhani@pnh	33333
20004	Melike Sahin	melikesahin@mlkshn	30040
20005	Sezen Aksu	sezenaksu@sznks	59000
20006	Can Bonomo	canbonomo@cnbnm	40909
20007	Ajda Pekkan	ajdapekkan@ajdpkkn	60000
20008	Sena Sener	senasener@snsnr	39000
20009	Duman	duman@dmn	50909

Sample Data of Songs_Belong_to Table:

sid	length	sname	art_id	gid	alb_id
40001	180	Bazen	20001	30001	50001
40002	170	Gonullcelen	20002	30002	50002
40003	160	Nehirler Durmaz	20003	30003	50003
40004	240	Pusulam Ruzgar	20004	30004	50004
40005	240	Vay	20005	30005	50005
40006	235	Tukeniyor Omrum	20006	30006	50006
40007	195	Haykiracak Nefesim Kalmasa Bile	20007	30007	50007
40008	195	Teni Tenime	20008	30008	50008

Operations and Queries

For connecting to the database: // From connect.py

```
def create_connection():
    connection = None
    try:
        connection = mysql.connector.connect(
            host='localhost',
            database='test',
            user='root',
            password='616161.Oo'
        )

        print("Connection to MySQL DB successful")
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Something is wrong with your user name or password")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("Database does not exist")
        else:
            print(err)
    return connection
```

For executing the Query: // From cursor_utils.py

```
def execute_query(query, params=None):
    cnx = create_connection()
    cursor = cnx.cursor()
    try:
        cursor.execute(query, params)
        cnx.commit()
    except mysql.connector.Error as err:
        print(f"Error: {err}")
    finally:
        cursor.close()
        cnx.close()
```

For fetching all data: // From cursor_utils.py

```python
def fetch_all(query, params=None):
    cnx = create_connection()
    cursor = cnx.cursor()
    try:
        cursor.execute(query, params)
        result = cursor.fetchall()
        return result
    except mysql.connector.Error as err:
        print(f"Error: {err}")
        return None
    finally:
        cursor.close()
        cnx.close()
```

// From main.py

```python
# Utility function to print the contents of a table
def print_table(table_name):
    query = f"SELECT * FROM {table_name} LIMIT 10"
    records = fetch_all(query)
    print(f"Latest Contents of {table_name}:")
    for record in records:
        print(record)
    print("\n")
```

// From main.py

```python
# CRUD operations for Artists table
def create_artist(art_id, name, biography, monthly_listening):
    print_table("Artists")  # Print before operation
    query = "INSERT INTO Artists (art_id, name, biography, monthly_listening) VALUES (%s, %s, %s, %s)"
    params = (art_id, name, biography, monthly_listening)
    execute_query(query, params)
    print_table("Artists")  # Print after operation

def update_artist(art_id, new_name):
    print_table("Artists") # Print before operation
    query = "UPDATE Artists SET name = %s WHERE art_id = %s"
    params = (new_name,art_id)
    execute_query(query, params)
    print_table("Artists") # Print after operation

def delete_artist(art_id):
    print_table("Artists") # Print before operation
    query = "DELETE FROM Artists WHERE art_id = %s"
    params = (art_id,)
    execute_query(query, params)
    print_table("Artists") # Print after operation
```

// From main.py

```python
# CRUD operations for Songs_Belong_to table

def create_song(sid, length, sname, art_id, gid, alb_id):
    print_table("Songs_Belong_to")  # Print before operation
    query = """INSERT INTO Songs_Belong_to (sid, length, sname, art_id, gid, alb_id)
               VALUES (%s, %s, %s, %s, %s, %s)"""
    params = (sid, length, sname, art_id, gid, alb_id)
    execute_query(query, params)
    print_table("Songs_Belong_to")  # Print after operation

def read_songs():
    query = "SELECT * FROM Songs_Belong_to"
    return fetch_all(query)

def update_song(sid, new_length, new_sname, new_art_id, new_gid, new_alb_id):
    print_table("Songs_Belong_to")  # Print before operation
    query = """UPDATE Songs_Belong_to
               SET length = %s, sname = %s, art_id = %s, gid = %s, alb_id = %s
               WHERE sid = %s"""
    params = (new_length, new_sname, new_art_id, new_gid, new_alb_id, sid)
    execute_query(query, params)
    print_table("Songs_Belong_to")  # Print after operation

def delete_song(sid):
    print_table("Songs_Belong_to")  # Print before operation
    query = "DELETE FROM Songs_Belong_to WHERE sid = %s"
    params = (sid,)
    execute_query(query, params)
    print_table("Songs_Belong_to")  # Print after operation
```

Screenshots of Before and After Logs

Example of Creating New Artist:

Python Code:

```
create_artist("20004", "Melike Sahin", "melikesahin@mlkshn", 30040)
```

Printing of Artists Table Before and After Inserting:

```
Connection to MySQL DB successful
Latest Contents of Artists:
('20000', 'Motive', 'motive@mob', 12321)
('20001', 'Canozan', 'canozan@cnzn', 11111)
('20002', 'Teoman', 'teoman@tmn', 22222)
('20003', 'Pinhani', 'pinhani@pnh', 33333)


Connection to MySQL DB successful
Connection to MySQL DB successful
Latest Contents of Artists:
('20000', 'Motive', 'motive@mob', 12321)
('20001', 'Canozan', 'canozan@cnzn', 11111)
('20002', 'Teoman', 'teoman@tmn', 22222)
('20003', 'Pinhani', 'pinhani@pnh', 33333)
('20004', 'Melike Sahin', 'melikesahin@mlkshn', 30040)
```

Example of Creating New Song:

Python Code:

```
create_song("40004", 240, "Pusulam Ruzgar", "20004", "30004", "50004")
```

Printing of Songs Table Before and After Inserting:

```
Connection to MySQL DB successful
Latest Contents of Songs_Belong_to:
('40000', 190, 'LOADED', '20000', '30000', '50000')
('40001', 180, 'Bazen', '20001', '30001', '50001')
('40002', 170, 'Gonullcelen', '20002', '30002', '50002')
('40003', 160, 'Nehirler Durmaz', '20003', '30003', '50003')


Connection to MySQL DB successful
Connection to MySQL DB successful
Latest Contents of Songs_Belong_to:
('40000', 190, 'LOADED', '20000', '30000', '50000')
('40001', 180, 'Bazen', '20001', '30001', '50001')
('40002', 170, 'Gonullcelen', '20002', '30002', '50002')
('40003', 160, 'Nehirler Durmaz', '20003', '30003', '50003')
('40004', 240, 'Pusulam Ruzgar', '20004', '30004', '50004')
```

Example of Updating Song:

Python Code:

```
# Update song
update_song("40009", 250, "Updated Song", "20009", "30009", "50009")
```

Printing of Songs Table Before and After Updating:

```
Connection to MySQL DB successful
Latest Contents of Songs_Belong_to:
('40000', 190, 'LOADED', '20000', '30000', '50000')
('40001', 180, 'Bazen', '20001', '30001', '50001')
('40002', 170, 'Gonullcelen', '20002', '30002', '50002')
('40003', 160, 'Nehirler Durmaz', '20003', '30003', '50003')
('40004', 240, 'Pusulam Ruzgar', '20004', '30004', '50004')
('40005', 240, 'Vay', '20005', '30005', '50005')
('40006', 235, 'Tukeniyor Omrum', '20006', '30006', '50006')
('40007', 195, 'Haykiracak Nefesim Kalmasa Bile', '20007', '30007', '50007')
('40008', 195, 'Teni Tenime', '20008', '30008', '50008')
('40009', 250, 'Yurek', '20009', '30009', '50009')


Connection to MySQL DB successful
Connection to MySQL DB successful
Latest Contents of Songs_Belong_to:
('40000', 190, 'LOADED', '20000', '30000', '50000')
('40001', 180, 'Bazen', '20001', '30001', '50001')
('40002', 170, 'Gonullcelen', '20002', '30002', '50002')
('40003', 160, 'Nehirler Durmaz', '20003', '30003', '50003')
('40004', 240, 'Pusulam Ruzgar', '20004', '30004', '50004')
('40005', 240, 'Vay', '20005', '30005', '50005')
('40006', 235, 'Tukeniyor Omrum', '20006', '30006', '50006')
('40007', 195, 'Haykiracak Nefesim Kalmasa Bile', '20007', '30007', '50007')
('40008', 195, 'Teni Tenime', '20008', '30008', '50008')
('40009', 250, 'Updated Song', '20009', '30009', '50009')
```

Example of Deleting Song:

Python Code:

```
# Delete song
delete_song("40009")
```

Printing of Songs Table Before and After Deletion:

```
Connection to MySQL DB successful
Latest Contents of Songs_Belong_to:
('40000', 190, 'LOADED', '20000', '30000', '50000')
('40001', 180, 'Bazen', '20001', '30001', '50001')
('40002', 170, 'Gonullcelen', '20002', '30002', '50002')
('40003', 160, 'Nehirler Durmaz', '20003', '30003', '50003')
('40004', 240, 'Pusulam Ruzgar', '20004', '30004', '50004')
('40005', 240, 'Vay', '20005', '30005', '50005')
('40006', 235, 'Tukeniyor Omrum', '20006', '30006', '50006')
('40007', 195, 'Haykiracak Nefesim Kalmasa Bile', '20007', '30007', '50007')
('40008', 195, 'Teni Tenime', '20008', '30008', '50008')
('40009', 250, 'Updated Song', '20009', '30009', '50009')


Connection to MySQL DB successful
Connection to MySQL DB successful
Latest Contents of Songs_Belong_to:
('40000', 190, 'LOADED', '20000', '30000', '50000')
('40001', 180, 'Bazen', '20001', '30001', '50001')
('40002', 170, 'Gonullcelen', '20002', '30002', '50002')
('40003', 160, 'Nehirler Durmaz', '20003', '30003', '50003')
('40004', 240, 'Pusulam Ruzgar', '20004', '30004', '50004')
('40005', 240, 'Vay', '20005', '30005', '50005')
('40006', 235, 'Tukeniyor Omrum', '20006', '30006', '50006')
('40007', 195, 'Haykiracak Nefesim Kalmasa Bile', '20007', '30007', '50007')
('40008', 195, 'Teni Tenime', '20008', '30008', '50008')
```

Example of Updating Artist:

Python Code:

```
# Update artist
update_artist("20000", "Moti(Updated)")
```

Printing of Artists Table Before and After Updating:

```
Connection to MySQL DB successful
Latest Contents of Artists:
('20000', 'Motive', 'motive@mob', 12321)
('20001', 'Canozan', 'canozan@cnzn', 11111)
('20002', 'Teoman', 'teoman@tmn', 22222)
('20003', 'Pinhani', 'pinhani@pnh', 33333)
('20004', 'Melike Sahin', 'melikesahin@mlkshn', 30040)
('20005', 'Sezen Aksu', 'sezenaksu@sznks', 59000)
('20006', 'Can Bonomo', 'canbonomc@cnbnm', 40909)
('20007', 'Ajda Pekkan', 'ajdapekkan@ajdpkkn', 60000)
('20008', 'Sena Sener', 'senasener@snsnr', 39000)
('20009', 'Duman', 'duman@dmn', 50909)


Connection to MySQL DB successful
Connection to MySQL DB successful
Latest Contents of Artists:
('20000', 'Moti(Updated)', 'motive@mob', 12321)
('20001', 'Canozan', 'canozan@cnzn', 11111)
('20002', 'Teoman', 'teoman@tmn', 22222)
('20003', 'Pinhani', 'pinhani@pnh', 33333)
('20004', 'Melike Sahin', 'melikesahin@mlkshn', 30040)
('20005', 'Sezen Aksu', 'sezenaksu@sznks', 59000)
('20006', 'Can Bonomo', 'canbonomc@cnbnm', 40909)
('20007', 'Ajda Pekkan', 'ajdapekkan@ajdpkkn', 60000)
('20008', 'Sena Sener', 'senasener@snsnr', 39000)
('20009', 'Duman', 'duman@dmn', 50909)
```

Example of Deleting Artist:

Python Code:

```
# Delete artists
delete_artist("20000")
```

Printing of Artists Table Before and After Deletion:

```
Connection to MySQL DB successful
Latest Contents of Artists:
('20000', 'Moti(Updated)', 'motive@mob', 12321)
('20001', 'Canozan', 'canozan@cnzn', 11111)
('20002', 'Teoman', 'teoman@tmn', 22222)
('20003', 'Pinhani', 'pinhani@pnh', 33333)
('20004', 'Melike Sahin', 'melikesahin@mlkshn', 30040)
('20005', 'Sezen Aksu', 'sezenaksu@sznks', 59000)
('20006', 'Can Bonomo', 'canbonomc@cnbnm', 40909)
('20007', 'Ajda Pekkan', 'ajdapekkan@ajdpkkn', 60000)
('20008', 'Sena Sener', 'senasener@snsnr', 39000)
('20009', 'Duman', 'duman@dmn', 50909)


Connection to MySQL DB successful
Connection to MySQL DB successful
Latest Contents of Artists:
('20001', 'Canozan', 'canozan@cnzn', 11111)
('20002', 'Teoman', 'teoman@tmn', 22222)
('20003', 'Pinhani', 'pinhani@pnh', 33333)
('20004', 'Melike Sahin', 'melikesahin@mlkshn', 30040)
('20005', 'Sezen Aksu', 'sezenaksu@sznks', 59000)
('20006', 'Can Bonomo', 'canbonomc@cnbnm', 40909)
('20007', 'Ajda Pekkan', 'ajdapekkan@ajdpkkn', 60000)
('20008', 'Sena Sener', 'senasener@snsnr', 39000)
('20009', 'Duman', 'duman@dmn', 50909)
```