

NIM : 5210411383

Nama : ALLAN BIL FAQIH

## Tugas Pertemuan 11

### SQLite Query

Nama File : NIM\_Nama\_Tugas11.pdf

---

1. Buatlah tabel dengan format seperti berikut,

Table name : employee

Field Name	Type	Length	Desc
employee_id	Integer	5	Auto increment, Primary Key
first_name	String	50	Not null
last_name	String	50	Not null
address	Text		Null
status	Boolean	1	Not null
age	Integer	2	Not null

Query :

```
CREATE TABLE "employee" (  
  "employee_id" INTEGER,  
  "first_name" STRING(50) NOT NULL,  
  "last_name" STRING(50) NOT NULL,  
  "address" TEXT,  
  "status" BOOLEAN(1) NOT NULL,  
  "age" INTEGER(2) NOT NULL,  
  PRIMARY KEY("employee_id" AUTOINCREMENT)  
)
```

Screenshot Coding :

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

cur.execute("""CREATE TABLE "employee"
              ("employee_id" INTEGER,
               "first_name" STRING(50) NOT NULL,
               "last_name" STRING(50) NOT NULL,
               "address" TEXT,"status" BOOLEAN(1) NOT NULL,
               "age" INTEGER(2) NOT NULL,
               PRIMARY KEY("employee_id" AUTOINCREMENT))""")
```

- *dilihat dari db browser*

▼	Tables (2)		
▼	employee		CREATE TABLE "employee" ( "employee_id" :
	employee_id	INTEGER	"employee_id" INTEGER
	first_name	STRING(50)	"first_name" STRING(50) NOT NULL
	last_name	STRING(50)	"last_name" STRING(50) NOT NULL
	address	TEXT	"address" TEXT
	status	BOOLEAN	"status" BOOLEAN NOT NULL
	age	INTEGER(2)	"age" INTEGER(2) NOT NULL

2. Masukkan 5 data berbeda ke table ! ( Data bebas )

Query :

```
insert into "employee" (first_name,last_name,address,status,age)
values('erik', 'salvia','big city salvia',1,21),
('sven', 'bardlike','riverwood, whiterun',0,19),
('karla', 'usurper','lakeside, hakali',0,18),
('jargon', 'sejargon','bardic city, bardic county',1,33),
('sizan', 'tsuzinsky','rowsland, uzkerbak',0,24)
```

Screenshot Coding:

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

cur.execute("""insert into "employee"
              (first_name,last_name,address,status,age)
              values('erik', 'salvia','big city salvia',1,21),
              ('sven', 'bardlike','riverwood, whiterun',0,19),
              ('karla', 'usurper','lakeside, hakali',0,18),
              ('jargon', 'sejargon','bardic city, bardic county',1,33),
              ('sizan', 'tsuzinsky','rowsland, uzkerbak',0,24) """)

conn.commit()
```

- dilihat dari db browser

Database Structure   Browse Data   Edit Pragmas   Execute SQL						
Table: <span>employee</span>						
	employee_id	first_name	last_name	address	status	age
	Filter	Filter	Filter	Filter	Filter	Filter
1	3	erik	salvia	big city salvia	1	21
2	4	sven	bardlike	riverwood, whiterun	0	19
3	5	karla	usurper	lakeside, hakali	0	18
4	6	jargon	sejargon	bardic city, bardic county	1	33
5	7	sizan	tsuzinsky	rowsland, uzkerbak	0	24

3. Select semua data yang ada di table dan tampilkan !.

Query :

```
select * from employee
```

Screenshot Coding :

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

qExec = cur.execute("select * from employee")

print('\n=====')
for row in qExec:
    print(row)
print('=====')
```

- dilihat dari terminal

```
=====
(1, 'erik', 'salvia', 'big city salvia', 1, 21)
(2, 'sven', 'bardlike', 'riverwood, whiterun', 0, 19)
(3, 'karla', 'usurper', 'lakeside, hakali', 0, 18)
(4, 'jargon', 'sejargon', 'bardic city, bardic county', 1, 33)
(5, 'sizan', 'tsuzinsky', 'rowsland, uzkerbak', 0, 24)
=====
```

#### 4. Select data menggunakan where ( data bebas )

Penjelasan :

WHERE digunakan untuk melakukan filter atau menyaring data, WHERE ini digunakan untuk mengekstrak data yang hanya memenuhi kondisi tertentu(kondisi yang disebutkan).

Query yang saya gunakan adalah untuk menyaring data dari table yang memiliki status boolean yang berisi nol/0.

Query :

```
select * from employee
where status='0'
```

Screenshot Coding:

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

qExec = cur.execute("select * from employee where status='0'")

print('\n=====')
for row in qExec:
    print(row)
print('=====')
```

- *dilihat dari terminal*

```
=====
(2, 'sven', 'bardlike', 'riverwood, whiterun', 0, 19)
(3, 'karla', 'usurper', 'lakeside, hakali', 0, 18)
(5, 'sizan', 'tsuzinsky', 'rowsland, uzkerbak', 0, 24)
=====
```

## 5. Hapus data yang ada di table ( Data Bebas )

Penjelasan :

DELETE digunakan untuk menghapus data dari table.

Pada query yang saya gunakan, data yang dihapus adalah data dari table employee dimana row yang akan dihapus memiliki column first\_name yang berisi "sizan", dan pada ss-nya dapat terlihat id number 8 sudah tidak ada karena id 8 itu merupakan miliknya delthis.

Query :

```
delete from employee where first_name='sizan'
```








Screenshot Coding :

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

qExec = cur.execute("delete from employee where first_name='sizan'")
conn.commit()
```

- dilihat dari db browser

Table:	employee							
	employee_id	first_name	last_name	address	status	age		
	Filter	Filter	Filter	Filter	Filter	Filter		
1	1	erik	salvia	big city...	1	21		
2	2	sven	bardlike	riverw...	0	19		
3	3	karla	usurper	lakesid...	0	18		
4	4	jargon	sejargon	bardic ...	1	33		

## 6. Update data yang ada di table ( Data Bebas )

Penjelasan :

UPDATE digunakan untuk mengubah data yang ada dalam tabel.

Pada query yang saya gunakan, data yang diubah adalah data dari table employee dimana row yang akan diubah memiliki column employee\_id yang berisi "1", dan data yang diubah adalah column status dari "1" ke "0", dan address dari "big city salvia" ke "sterlani, movarizo".

Query :

```
UPDATE employee
SET status = '0', address= 'sterlani, movarizo'
WHERE employee_id = 1;
```

Screenshot Coding:

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

qExec = cur.execute("""UPDATE employee SET status = '0',
                        address= 'sterlani, movarizo'
                        WHERE employee_id = 1""")
conn.commit()
```

- dilihat dari db browser

Table: employee						
	employee_id	first_name	last_name	address	status	age
	Filter	Filter	Filter	Filter	Filter	Filter
1	1	erik	salvia	sterlani, movarizo	0	21
2	2	sven	bardlike	riverwood, whiterun	0	19
3	3	karla	usurper	lakeside, hakali	0	18
4	4	jargon	sejargon	bardic city, bardic county	1	33

## 7. Select data menggunakan LIKE

Penjelasan :

Operator LIKE digunakan di WHERE untuk mencari pola tertentu dalam kolom, ada dua wildcard yang sering digunakan bersama dengan operator LIKE:

- Tanda persen (%) mewakili nol, satu, atau beberapa karakter
- Tanda garis bawah (\_) mewakili satu karakter tunggal

Pada query saya, saya ingin mencari data dari column first\_name yang memiliki huruf "e" pada nama mereka dengan menggunakan wildcard % (percent).

Query :

```
select * from employee
where first_name like '%e%'
```

Screenshot Coding :

```
import sqlite3 as db

conn = db.connect("database.db")
cur = conn.cursor()

qExec = cur.execute("select * from employee where first_name like '%e%'")

print('\n=====')
for row in qExec:
    print(row)
print('=====')
```

- *dilihat dari terminal*

```
=====
(1, 'erik', 'salvia', 'sterlani, movarizo', 0, 21)
(2, 'sven', 'bardlike', 'riverwood, whiterun', 0, 19)
=====
```