

Наше дослідження полягає у зберіганні даних та їхньому аналізі, тому ми вирішили, що найкраще для цього підійде використання бази даних. Python надає можливість використовувати SQL запити завдяки бібліотеці sqlite3.

```
import sqlite3
```

```
conn = sqlite3.connect('example.db')
```

```
c = conn.cursor()
```

```
# Create table
```

```
c.execute("""CREATE TABLE stocks
            (date text, trans text, symbol text, qty real, price real)""")
```

```
# Insert a row of data
```

```
c.execute("INSERT INTO stocks VALUES ('2006-01-05','BUY','RHAT',100,35.14)")
```

```
t = ('RHAT',)
```

```
c.execute('SELECT * FROM stocks WHERE symbol=?', t)
```

```
print c.fetchone()
```

```
# Larger example that inserts many records at a time
```

```
purchases = [('2006-03-28', 'BUY', 'IBM', 1000, 45.00),
```

```
              ('2006-04-05', 'BUY', 'MSFT', 1000, 72.00),
```

```
              ('2006-04-06', 'SELL', 'IBM', 500, 53.00),
```

```
              ]
```

```
c.executemany('INSERT INTO stocks VALUES (?, ?, ?, ?, ?)', purchases)
```

```
# Save (commit) the changes
```

```
conn.commit()
```

```
# We can also close the connection if we are done with it.  
# Just be sure any changes have been committed or they will be lost.  
conn.close()
```

Для роботи з цією бібліотекою ми також розробили абстрактний клас Database, в якому є методи які дозволяють оновити інформацію (для нашої програми це поле, в якому зберігається загальна сума витрат користувача в певній категорії) та додати новий запис в базу даних (категорію, в якій здійснено покупку, дата, місце, сума).

Також було розроблено клас Receipt, який надає можливість конвертувати зображення рядок та отримати необхідну інформацію (катгорія, місце покупки та сума). У класі використано додаткові бібліотеки PIL та pytesseract.