Week 1 Coding Practice

Submitted by - Atharv Jangam

```
import sqlite3
import pandas as pd
```

```
read_emp = pd.read_csv(r'emp.csv')
read_emp.to_sql('emp', conn, if_exists='append', index = False) # Insert the values from the csv file into the table 'emp'
```

```
     14
```

STEP 3. read dept.csv and create a table dept

```
read_dept = pd.read_csv(r'dept.csv')
read_dept.to_sql('dept', conn, if_exists='append', index = False) # Insert the values from the csv file into the table 'dept'
```

```
     4
```

Execution Examples

SQL statements will be executed with

c.execute(''' SQL code ''')

```
#Example 1
for row in c.execute('''
select * from emp
'''):
    print(row)
```

```
(7369, 'SMITH', 'CLERK', 7902.0, '17-Dec-05', 800, None, 20)
(7499, 'ALLEN', 'SALESMAN', 7698.0, '20-Feb-06', 1600, 300.0, 30)
(7521, 'WARD', 'SALESMAN', 7698.0, '22-Feb-06', 1250, 500.0, 30)
(7566, 'JONES', 'MANAGER', 7839.0, '2-Apr-06', 2975, None, 20)
(7654, 'MARTIN', 'SALESMAN', 7698.0, '28-Sep-06', 1250, 1400.0, 30)
(7698, 'BLAKE', 'MANAGER', 7839.0, '1-May-06', 2850, None, 30)
(7782, 'CLARK', 'MANAGER', 7839.0, '9-Jun-06', 2450, None, 10)
(7788, 'SCOTT', 'ANALYST', 7566.0, '9-Dec-07', 3000, None, 20)
(7839, 'KING', 'PRESIDENT', None, '17-Nov-06', 5000, None, 10)
(7844, 'TURNER', 'SALESMAN', 7698.0, '8-Sep-06', 1500, 0.0, 30)
(7876, 'ADAMS', 'CLERK', 7788.0, '12-Jan-08', 1100, None, 20)
(7900, 'JAMES', 'CLERK', 7698.0, '3-Dec-06', 950, None, 30)
(7902, 'FORD', 'ANALYST', 7566.0, '3-Dec-06', 3000, None, 20)
(7934, 'MILLER', 'CLERK', 7782.0, '23-Jan-07', 1300, None, 10)
```

```
colnames = c.description
for row in colnames:
    print(row[0])
```

```
EMPNO
ENAME
JOB
MGR
HIREDATE
SAL
COMM
DEPTNO
```

To print a table, use fetchall() to collect data and add column names thaht you have selected.

```
# Example 2
c.execute('''
select * from emp
''')

df = pd.DataFrame(c.fetchall(), columns=['EMPNO',
'ENAME',
'JOB',
'MGR',
'HIREDATE',
'SAL',
'COMM',
'DEPTNO'])
print(df)
```

```
     EMPNO   ENAME        JOB     MGR   HIREDATE   SAL    COMM  DEPTNO
0     7369   SMITH      CLERK  7902.0  17-Dec-05   800     NaN      20
1     7499   ALLEN   SALESMAN  7698.0  20-Feb-06  1600   300.0      30
2     7521    WARD   SALESMAN  7698.0  22-Feb-06  1250   500.0      30
3     7566   JONES    MANAGER  7839.0   2-Apr-06  2975     NaN      20
4     7654  MARTIN   SALESMAN  7698.0  28-Sep-06  1250  1400.0      30
5     7698   BLAKE    MANAGER  7839.0   1-May-06  2850     NaN      30
6     7782   CLARK    MANAGER  7839.0   9-Jun-06  2450     NaN      10
7     7788   SCOTT    ANALYST  7566.0   9-Dec-07  3000     NaN      20
8     7839    KING  PRESIDENT     NaN  17-Nov-06  5000     NaN      10
9     7844  TURNER   SALESMAN  7698.0   8-Sep-06  1500     0.0      30
10    7876   ADAMS      CLERK  7788.0  12-Jan-08  1100     NaN      20
11    7900   JAMES      CLERK  7698.0   3-Dec-06   950     NaN      30
12    7902    FORD    ANALYST  7566.0   3-Dec-06  3000     NaN      20
13    7934  MILLER      CLERK  7782.0  23-Jan-07  1300     NaN      10
```

Basics of SQL Queries

SELECT: Statement used to select rows and columns from a database.

FROM: Specifies which table in the database you want to direct your query to.

WHERE: Clause for filtering for specified value(s).

GROUP BY: Aggregating data. Needs to be used in conjunction with SQL aggregating functions like `SUM` and `COUNT`.

ORDER BY: Sorting columns in the database.

JOIN: Joins are used to combine tables with one another.

UNION, **INTERSECT/EXCEPT**: Set operations. Unioning in SQL allows one to append tables on top of one another.

Step 5. Practice Chapter 1

```
## Your turn
```

Step 6. Close the connection

```
conn.close()
```

Opening connection with database

```
conn = sqlite3.connect('week1.db')
c = conn.cursor()
```

```
## You can continue working with SQL coding now
```

```
for row in c.execute('''select * from emp'''):
    print(row)
```

```
(7369, 'SMITH', 'CLERK', 7902.0, '17-Dec-05', 800, None, 20)
(7499, 'ALLEN', 'SALESMAN', 7698.0, '20-Feb-06', 1600, 300.0, 30)
(7521, 'WARD', 'SALESMAN', 7698.0, '22-Feb-06', 1250, 500.0, 30)
(7566, 'JONES', 'MANAGER', 7839.0, '2-Apr-06', 2975, None, 20)
(7654, 'MARTIN', 'SALESMAN', 7698.0, '28-Sep-06', 1250, 1400.0, 30)
(7698, 'BLAKE', 'MANAGER', 7839.0, '1-May-06', 2850, None, 30)
(7782, 'CLARK', 'MANAGER', 7839.0, '9-Jun-06', 2450, None, 10)
(7788, 'SCOTT', 'ANALYST', 7566.0, '9-Dec-07', 3000, None, 20)
(7839, 'KING', 'PRESIDENT', None, '17-Nov-06', 5000, None, 10)
(7844, 'TURNER', 'SALESMAN', 7698.0, '8-Sep-06', 1500, 0.0, 30)
(7876, 'ADAMS', 'CLERK', 7788.0, '12-Jan-08', 1100, None, 20)
(7900, 'JAMES', 'CLERK', 7698.0, '3-Dec-06', 950, None, 30)
```

```
(7902, 'FORD', 'ANALYST', 7566.0, '3-Dec-06', 3000, None, 20)
(7934, 'MILLER', 'CLERK', 7782.0, '23-Jan-07', 1300, None, 10)
```

```
for row in c.execute('''select * from emp where deptno = 10'''):
    print(row)
```

```
(7782, 'CLARK', 'MANAGER', 7839.0, '9-Jun-06', 2450, None, 10)
(7839, 'KING', 'PRESIDENT', None, '17-Nov-06', 5000, None, 10)
(7934, 'MILLER', 'CLERK', 7782.0, '23-Jan-07', 1300, None, 10)
```

1.3 Finding Rows That Satisfy Multiple Conditions

```
 for row in c.execute('''select * from emp where deptno = 10 or comm is not null or sal <= 2000 and deptno=20'''):
    print(row)
```

```
(7369, 'SMITH', 'CLERK', 7902.0, '17-Dec-05', 800, None, 20)
(7499, 'ALLEN', 'SALESMAN', 7698.0, '20-Feb-06', 1600, 300.0, 30)
(7521, 'WARD', 'SALESMAN', 7698.0, '22-Feb-06', 1250, 500.0, 30)
(7654, 'MARTIN', 'SALESMAN', 7698.0, '28-Sep-06', 1250, 1400.0, 30)
(7782, 'CLARK', 'MANAGER', 7839.0, '9-Jun-06', 2450, None, 10)
(7839, 'KING', 'PRESIDENT', None, '17-Nov-06', 5000, None, 10)
(7844, 'TURNER', 'SALESMAN', 7698.0, '8-Sep-06', 1500, 0.0, 30)
(7876, 'ADAMS', 'CLERK', 7788.0, '12-Jan-08', 1100, None, 20)
(7934, 'MILLER', 'CLERK', 7782.0, '23-Jan-07', 1300, None, 10)
```

```
for row in c.execute('''select * from emp where  ( deptno = 10 or comm is not null or sal <= 2000 ) and deptno=20'''):
    print(row)
```

```
(7369, 'SMITH', 'CLERK', 7902.0, '17-Dec-05', 800, None, 20)
(7876, 'ADAMS', 'CLERK', 7788.0, '12-Jan-08', 1100, None, 20)
```

1.4 Retrieving a Subset of Columns from a Table

```
for row in c.execute('''select  ename,deptno,sal from emp'''):
    print(row)
```

```
('SMITH', 20, 800)
('ALLEN', 30, 1600)
('WARD', 30, 1250)
('JONES', 20, 2975)
('MARTIN', 30, 1250)
('BLAKE', 30, 2850)
('CLARK', 10, 2450)
('SCOTT', 20, 3000)
('KING', 10, 5000)
('TURNER', 30, 1500)
('ADAMS', 20, 1100)
('JAMES', 30, 950)
('FORD', 20, 3000)
('MILLER', 10, 1300)
```

1.5 Providing Meaningful Names for Columns

```
for row in c.execute('''select sal as salary, comm as commission from emp'''):
    print(row)
```

```
(800, None)
(1600, 300.0)
(1250, 500.0)
(2975, None)
(1250, 1400.0)
(2850, None)
(2450, None)
(3000, None)
(5000, None)
(1500, 0.0)
(1100, None)
(950, None)
(3000, None)
(1300, None)
```