



تمرین دوم درس آزمایشگاه پایگاه داده

زمان کلاس: سه شنبه ها ساعت ۱۶.۳۰

عطیه براتی نیا ۹۶۳۱۰۱۰

*کدهای پایتون در دو نوع .py و .ipynb. کنار این فایل قابل دسترس می باشد.

سوال ۱

تغییر نام ستون بر روی جدول student و اضافه کردن یک ستون جدید به course

The screenshot shows a Jupyter Notebook interface with a code cell containing Python code to connect to a PostgreSQL database and execute an SQL command to add a new column to the 'course' table. The code is as follows:

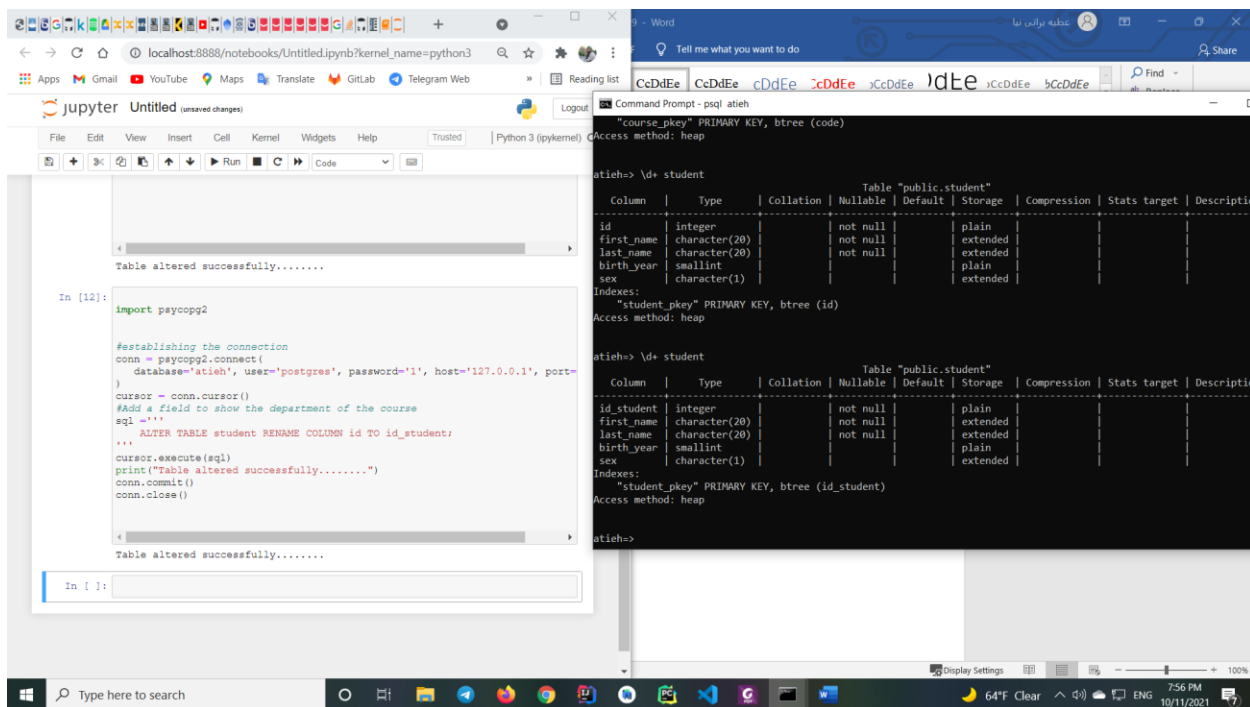
```
In [11]:  
  
import psycopg2  
  
#establishing the connection  
conn = psycopg2.connect(  
    database='atiah', user='postgres', password='1', host='localhost'  
)  
cursor = conn.cursor()  
#Add a field to show the department of the course  
sql = '''  
    ALTER TABLE course ADD COLUMN department CHAR(20);  
    '''  
cursor.execute(sql)  
print("Table altered successfully.....")  
conn.commit()  
conn.close()  
  
Table altered successfully.....  
  
In [ ]:
```

Below the code cell, the output of the SQL command is displayed in a terminal window. The output shows the 'List of relations' and the 'Table "public.course"' structure.

Command Prompt - psql atiah

```
ERROR: schema "atiah" does not exist  
atiah=> select * from course;  
ERROR: permission denied for table course  
atiah=> select * from student;  
ERROR: permission denied for table student  
atiah=> \dt+ course  
  
List of relations  
Schema | Name | Type | Owner | Persistence | Access method | Size | Description  
-----+-----+-----+-----+-----+-----+-----+-----  
public | course | table | postgres | permanent | heap | 0 bytes |  
(1 row)  
  
atiah=> \dt+ course  
  
Table "public.course"  
Column | Type | Collation | Nullable | Default | Storage | Compression | Statistics  
-----+-----+-----+-----+-----+-----+-----+-----  
code | integer | | not null | | plain | |  
name | character(20) | | not null | | extended | |  
code_teacher | character(20) | | not null | | extended | |  
time_class | time without time zone | | | | plain | |  
department | character(20) | | | | extended | |  
Indexes:  
"course_pkey" PRIMARY KEY, btree (code)  
Access method: heap  
atiah=>
```

شکل ۱ جدول course



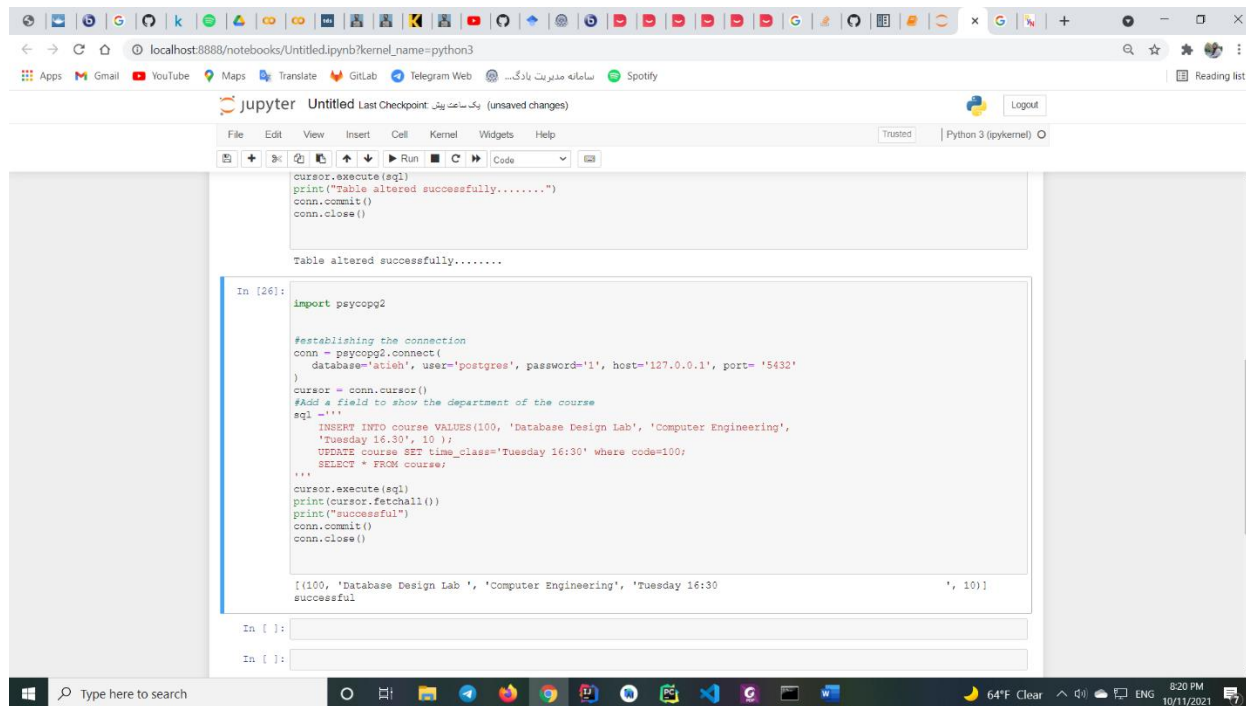
شکل ۲ جدول student

سوال ۲

یک جدول به نام course داریم که اطلاعات درس‌ها در آن قرار می‌گیرد و یک جدول به نام teacher داریم که اطلاعات اساتید در آن قرار می‌گیرد. جدول course یک ستون به نام id_teacher دارد که کلید خارجی به id جدول teacher است. بدین صورت هر درس با داشتن کلید استاد آن، می‌توان به اطلاعات استاد آن درس دسترسی پیدا کرد.

سوال ۳

اجرای سه دستور insert, update و select بر روی جدول course دیده می شود.



The screenshot shows a Jupyter Notebook running on a local host. The browser address bar indicates the URL is localhost:8888/notebooks/Untitled.ipynb?kernel_name=python3. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, cell execution, and output viewing. The code editor contains two cells. The first cell executes a SQL query to alter a table, and the second cell imports the psycopg2 library, establishes a database connection, and executes a series of SQL commands including INSERT, UPDATE, and SELECT. The output of the second cell shows the successful execution of the queries and the resulting data from the SELECT statement.

```
cursor.execute(sql)
print("Table altered successfully.....")
conn.commit()
conn.close()

Table altered successfully.....

In [26]:
import psycopg2

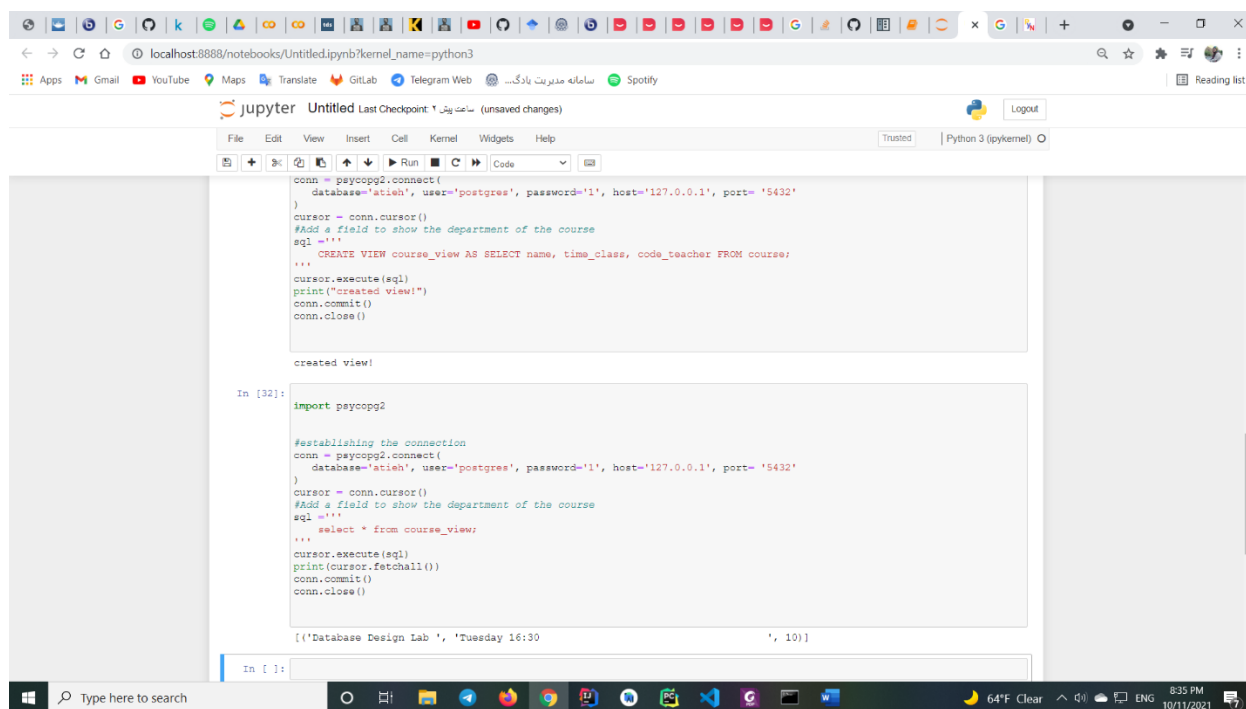
#establishing the connection
conn = psycopg2.connect(
    database='latih', user='postgres', password='1', host='127.0.0.1', port= '5432'
)
cursor = conn.cursor()
#Add a field to show the department of the course
sql = """
INSERT INTO course VALUES(100, 'Database Design Lab', 'Computer Engineering',
'Tuesday 16.30', 10 );
UPDATE course SET time_class='Tuesday 16:30' where code=100;
SELECT * FROM course;
...
cursor.execute(sql)
print(cursor.fetchall())
print("successful")
conn.commit()
conn.close()

[(100, 'Database Design Lab ', 'Computer Engineering', 'Tuesday 16:30
successful', 10)]

In [ ]:
In [ ]:
```

سوال ۴

یک ویو روی جدول course ساخته شد.



```
conn = psycopg2.connect(
    database='atish', user='postgres', password='1', host='127.0.0.1', port= '5432'
)
cursor = conn.cursor()
#Add a field to show the department of the course
sql = """
CREATE VIEW course_view AS SELECT name, time_class, code_teacher FROM course;
...
cursor.execute(sql)
print("created view!")
conn.commit()
conn.close()

created view!

In [32]:
import psycopg2

#establishing the connection
conn = psycopg2.connect(
    database='atish', user='postgres', password='1', host='127.0.0.1', port= '5432'
)
cursor = conn.cursor()
#Add a field to show the department of the course
sql = """
select * from course_view;
...
cursor.execute(sql)
print(cursor.fetchall())
conn.commit()
conn.close()

[('Database Design Lab ', 'Tuesday 16:30', 10)]
```

سوال ۵

حسابی: این نوع عملگرها برای محاسبه بین مقادیر ستون‌های مختلف هستند. مثلاً تعداد ساعت کاری یک فرد در یک ستون و دستمزد ساعتی او در ستونی دیگر قرار می‌گیرد. با ضرب این دو ستون می‌توان به حقوق او دست پیدا کرد.

مقایسه‌ای: همانطور که از اسم آن مشخص است این نوع عملگرها برای مقایسه استفاده می‌شود مثلاً می‌خواهیم حقوق افرادی که ساعتی بیش از ۵۰ هزار تومان می‌گیرند را پیدا کنیم. در این مساله از مقایسه‌ی >50 استفاده خواهد شد.