Ethan Engel

COMP 3421

Week 8 Assignment

Description

This Python file opens the secondary_school database in mysql. It then takes data and loads it into the following tables:

Students(sid*, grade, name, contact_info, address)

Courses(course_id*, name, credits)

Enrolled_in(sid, course_id, term), where sid and course_id are foreign keys connected to the students and courses tables, respectively.

The program then initializes a variable, which will be the maximum course ID number for which freshmen will be allowed to register, in this case 120. It then finds the violations of this registration constraint, stores them in a temporary table, and then disenrolls (deletes) them from the enrolled_in table. To demonstrate the functionality, the same select statement is performed prior to and after the deletions, the latter rendering an empty list. As further evidence, counts are printed from enrolled_in, before and after the deletions. This shows that just the enrollment violations were deleted, and no more.

Python code

```
import mysql.connector


mydb = mysql.connector.connect(
  user='ethan1',    # could be root, or a user you created, I created 'testuser'
  passwd='',  # the password for that use
  database='secondary_school',   # the database to connect to
  host='127.0.0.1',   # localhost
  allow_local_infile='1'  # needed so can load local files
)


print(mydb)
myc = mydb.cursor()   # myc name short for "my cursor"
```

```
myc.execute('set global local_infile = 1;')

myc.execute ("use secondary_school")

myc.execute("drop table if exists temp_enrolled_in;")
myc.execute("drop table if exists enrolled_in;")
myc.execute("drop table if exists courses;")
myc.execute("drop table if exists students;")

myc.execute("""
create table students(
  sid int,
  grade int,
  name varchar(25),
  contact_info varchar(50),
  address varchar(50),
  Primary Key (sid) ) ;
""")

myc.execute("""
create table courses (
  course_id int,
  name varchar(25),
  credits float,
  PRIMARY KEY (course_id) ) ;
""")

myc.execute("""
create table enrolled_in (
```

```
  sid int,
  course_id int,
  term varchar(20),
  FOREIGN KEY(sid) references students(sid),
  FOREIGN KEY(course_id) references courses(course_id))
  ;
""")


myc.execute("""
  load data local infile '/Users/elain/Desktop/COMP3421/data_students.txt' into table students
  fields terminated by ','
  lines terminated by '\n' ;
""")


myc.execute("""
load data local infile '/Users/elain/Desktop/COMP3421/data_courses.txt' into table courses
  fields terminated by ','
  lines terminated by '\n' ;
""")

myc.execute("""
load data local infile '/Users/elain/Desktop/COMP3421/data_enrolled_in.txt' into table enrolled_in
  fields terminated by ','
  lines terminated by '\n' ;
""")

myc.execute('select count(*) from enrolled_in;')
```

```
before = myc.fetchone()
print("\n Before deletions total enrolled_in count= ")
print(before)


myc.execute('set @freshmax=120')
myc.execute("""
Select E.sid, E.course_id from students S, enrolled_in E
where S.sid=E.sid and S.grade=9 and E.course_id>@freshmax
""")
overmax = myc.fetchall()
print ("\n These are the student id's of freshmen that have registered for a course over the allowable course id maximum.  The corresponding course id is also listed.  \n")
print (overmax)
print("\n Total enrollment violations = ", len(overmax))



myc.execute("""
create table temp_enrolled_in as
(Select E.sid, E.course_id, E.term from students S, enrolled_in E
where S.sid=E.sid and S.grade=9 and E.course_id>@freshmax)
""")

print("\n Now deleting:")
myc.execute("""
DELETE FROM enrolled_in
WHERE EXISTS (
SELECT *
FROM temp_enrolled_in
where temp_enrolled_in.sid=enrolled_in.sid
```

```
and temp_enrolled_in.course_id=enrolled_in.course_id
and temp_enrolled_in.term= enrolled_in.term)
""")

myc.execute("""
Select E.sid, E.course_id from students S, enrolled_in E
where S.sid=E.sid and S.grade=9 and E.course_id>@freshmax
""")
newovermax = myc.fetchall()
print (newovermax)

myc.execute('select count(*) from enrolled_in;')
after = myc.fetchone()
print("\n After deletions total enrolled_in count= ")
print (after)

mydb.commit()
mydb.close()
```

Output

```
C:\Users\elain\Desktop\COMP3421>python3 createAndLoadsecondary_school.py
<mysql.connector.connection.MySQLConnection object at 0x0000023825E884C0>

 Before deletions total enrolled_in count=
(1000,)
```

These are the student id's of freshmen that have registered for a course over the allowable course id maximum. The corresponding course id is also listed.

[(2, 149), (2, 190), (2, 133), (2, 198), (2, 185), (2, 185), (2, 174), (2, 176), (2, 134), (2, 163), (6, 192), (6, 180), (6, 151), (6, 121), (6, 167), (6, 193), (6, 160), (6, 134), (6, 147), (7, 192), (7, 122), (7, 139), (7, 169), (7, 133), (7, 145), (20, 195), (20, 137), (20, 145), (20, 146), (20, 194), (25, 184), (25, 122), (25, 144), (25, 184), (25, 126), (25, 140), (25, 172), (25, 179), (25, 140), (25, 190), (27, 187), (27, 122), (27, 147), (27, 178), (30, 167), (30, 156), (30, 147), (30, 190), (30, 161), (30, 156), (30, 149), (30, 152), (30, 191), (35, 133), (35, 131), (35, 134), (35, 164), (35, 125), (35, 123), (35, 177), (35, 125), (38, 194), (38, 158), (38, 159), (38, 198), (38, 164), (41, 172), (41, 160), (41, 151), (41, 164), (41, 186), (41, 174), (49, 131), (49, 182), (49, 181), (49, 136), (52, 197), (52, 143), (52, 123), (52, 164), (52, 135), (52, 133), (52, 195), (52, 186), (52, 149), (52, 149), (52, 168), (60, 165), (60, 142), (60, 195), (60, 147), (60, 126), (64, 133), (64, 143), (64, 124), (64, 121), (64, 153), (64, 173), (64, 155), (64, 193), (64, 191), (64, 158), (66, 194), (66, 152), (66, 153), (66, 196), (68, 135), (68, 133), (68, 132), (68, 186), (68, 136), (68, 182), (68, 182), (68, 148), (71, 137), (71, 193), (71, 138), (71, 183), (71, 197), (72, 188), (72, 188), (72, 128), (72, 180), (72, 154), (72, 170), (72, 195), (72, 174), (72, 159), (73, 144), (73, 166), (73, 153), (73, 183), (73, 180), (73, 145), (73, 132), (73, 138), (73, 159), (73, 192), (74, 153), (74, 143), (74, 175), (74, 177), (74, 196), (74, 145), (76, 189), (76, 153), (76, 126), (76, 122), (77, 190), (77, 169), (77, 147), (77, 123), (77, 133), (77, 157), (77, 193), (84, 197), (84, 127), (84, 194), (84, 131), (84, 130), (84, 170), (84, 175), (84, 129), (85, 134), (85, 176), (85, 159), (85, 174), (85, 144), (85, 200), (85, 190), (85, 165), (85, 121), (85, 128), (85, 135), (85, 159), (90, 144), (90, 124), (90, 136), (90, 133), (90, 184), (90, 164), (90, 125), (93, 152), (93, 196), (93, 165), (93, 172), (93, 141), (93, 183), (93, 180), (93, 199), (93, 165), (93, 139), (93, 179)]

Total enrollment violations = 193

Now deleting:

[]

After deletions total enrolled_in count=

(807,)