Populate_myDB

May 14, 2021

[ ]:
```
import mysql.connector
import random
import names
import randominfo
from faker import Faker
```

[ ]:
```
#here i create the connection

db=mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="00000000",
    auth_plugin='mysql_native_password'

)

mycursor=db.cursor(buffered=True)

print(db)
```

[ ]:
```
person=[]
seller_person=[]

fake=Faker()
number_customer=50
number_seller=20
domain='@ecommerce.ch'
cities=['Zurich','Lugano','Geneva','Lausanne','Bellinzona','Luzern']
zip_code=['8000','6900','1201','1001','6500','6003']

customer_id=[i for i in range(1,number_customer*2,2)]
seller_id=[e for e in range(100,number_seller*4+100,4)]

user_id_customer=random.sample(
    [a for a in range(1,number_customer+1)],50)
user_id_seller=random.sample(
    [a for a in range(60,number_seller+61)],20)
```

```
age=[a for a in range(16,50)]
sex=['male','female']


user_list=user_id_customer+user_id_seller
```

[ ]:
```
for i in user_list:

    print("INSERT INTO ecommerce.users(user_id) VALUES (%s)"%i)
    mycursor.execute("INSERT INTO ecommerce.users(user_id) VALUES (%s)"%i)

db.commit()
```

I created some function to generate random customer,sellers and so on. These function will generate all the things that will populate my DB. I used the connection with the library MySQL and after create a cursor to write on the tables of my database.

[ ]:
```
def customer_person(number_c):

    counter=0
    id_customer=customer_id[counter]
    condition=True

    for i in range(1,number_c):
        single_customer=()
        condition=True

        while condition==True:

            sex_temp=random.choice(sex)
            single_customer=single_customer+(user_id_customer[counter],)
            single_customer=single_customer+(names.get_last_name()[0:5].
↪lower()+domain,)
            single_customer=single_customer+(customer_id[counter],)

            single_customer=single_customer+(names.
↪get_full_name(gender=sex_temp),)
            single_customer=single_customer+(sex_temp,)
            single_customer=single_customer+(random.choice(age),)
            single_customer=single_customer+(randominfo.random_password(),)
            single_customer=single_customer+(fake.address()[3:17],)
            zip_temp=random.choice(zip_code)
            index=zip_code.index(zip_temp)
            city=cities[index]
            single_customer=single_customer+(zip_temp,city)
```

```
            person.append(single_customer)

            counter+=1

            condition=False



customer_person(number_customer)


for i in person:
    print("INSERT INTO ecommerce.
 ↪customer(user_id,email_c,id_customer,customer_name,password_c,address,zip_code,city_c)␣
 ↪VALUES"+str(i)+";")
```

[ ]:
```
sql="INSERT INTO ecommerce.
 ↪customer(user_id,email_c,id_customer,customer_name,sex_customer,age_customer,password_c,add
 ↪VALUES (%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)"
print(sql)
print(mycursor.executemany(sql,person))
db.commit()
```

[ ]:
```
def sellers_maker(number_s):

    counter=0
    id_seller=seller_id[counter]
    condition=True

    for i in range(1,number_s):
        single_seller=()
        condition=True

        while condition==True:

            sex_temp=random.choice(sex)
            single_seller=single_seller+(user_id_seller[counter],)
            single_seller=single_seller+(names.get_last_name()[0:5].
 ↪lower()+domain,)
            single_seller=single_seller+(seller_id[counter],)
            single_seller=single_seller+(names.get_full_name(gender=sex_temp),)
            single_seller=single_seller+(random.choice(age),)
            single_seller=single_seller+(sex_temp,)
```

```
            single_seller=single_seller+(fake.address()[3:17],)
            zip_temp=random.choice(zip_code)
            index=zip_code.index(zip_temp)
            city=cities[index]
            single_seller=single_seller+(city,)
            single_seller=single_seller+(randominfo.random_password(),)
            seller_person.append(single_seller)

            counter+=1

            condition=False



sellers_maker(number_seller)


for e in seller_person:
    print("INSERT INTO ecommerce.
 ↪sellers(user_id,email_s,id_seller,seller_name,address,city_s,password_seller)␣
 ↪VALUES"+str(e)+';')
```

[ ]:
```
sql="INSERT INTO ecommerce.
 ↪sellers(user_id,email_s,id_seller,seller_name,age_seller,sex_seller,address,city_s,password
 ↪VALUES (%s,%s,%s,%s,%s,%s,%s,%s,%s)"
print(sql)
print(mycursor.executemany(sql,seller_person))
db.commit()
```

[ ]:
```
supplier_list=[]

def suppliers_maker():

    counter=0

    condition=True
    supplier_name=['Computer Ltd','Shoes Inc','Adidas Inc','Apple Inc',
                   'Library Fantasy Ltd','DC Comics Ltd','HomeLife Ltd']


  ␣
 ↪phone_number=['420989849','42759095','42898695','38768940','23303930','90202029',
                  '900303920']

    for i in range(0,len(supplier_name)):
```

```
        single_supplier=()
        condition=True

        while condition==True:

            single_supplier=single_supplier+(str(counter)+'40SUP',)
            single_supplier=single_supplier+(supplier_name[counter],)
            single_supplier=single_supplier+(random.choice(cities),)
            single_supplier=single_supplier+(phone_number[counter],)


            supplier_list.append(single_supplier)

            counter+=1

            condition=False


suppliers_maker()


for i in supplier_list:
    print("INSERT INTO ecommerce.
 ↪suppliers(id_supp,supplier_name,city_supplier,phone_number) VALUEs"+str(i))
```

[ ]:
```
sql="INSERT INTO ecommerce.
 ↪suppliers(id_supp,supplier_name,city_supplier,phone_number)␣
 ↪VALUES(%s,%s,%s,%s)"
print(sql)
print(mycursor.executemany(sql,supplier_list))
db.commit()
```

[ ]:
```
warehouse_list=[]
def make_warehouse():

    counter=1

    condition=True
    warehouse_name=['SpaceRE','DarkGain','ConnectW','FormalWay',
                    'BlueLake','WoodForest']

    adress_w=['32 Strasse','67 Avenue','50 Sud Tirol','34 Gorge Street',
```

```
              '40 Lake Street','90 Dare Street']

    for i in range(1,len(warehouse_name)):
        single_warehouse=()
        condition=True

        while condition==True:

            single_warehouse=single_warehouse+(str(counter)+'56WH',)
            single_warehouse=single_warehouse+(warehouse_name[counter],)
            single_warehouse=single_warehouse+(adress_w[counter],)
            single_warehouse=single_warehouse+(random.choice(cities),)


            warehouse_list.append(single_warehouse)

            counter+=1

            condition=False

make_warehouse()

for i in warehouse_list:
    print("INSERT INTO ecommerce.warehouse(id_place,w_name,address_w,city_w)␣
 ↪VALUES"+str(i))
```

[ ]:
```
sql="INSERT INTO ecommerce.warehouse(id_place,w_name,address_w,city_w)␣
 ↪VALUES(%s,%s,%s,%s)"
print(sql)
print(mycursor.executemany(sql,warehouse_list))
db.commit()
```

[ ]:
```
d_agents_list=[]

def make_agents(n_agents):

    counter=1

    company_names=['Transport Ltd','Connection Ltd','SpeedTrain Inc']

    for i in range(1,n_agents):
```

```
        single_agents=()
        condition=True

        while condition==True:

            single_agents=single_agents+(str(counter)+'30A',)
            single_agents=single_agents+(random.choice(company_names),)

            d_agents_list.append(single_agents)

            counter+=1

            condition=False



make_agents(10)


for i in d_agents_list:
    print("INSERT INTO ecommerce.delivery_agents(id_agent,company_name)␣
 ↪VALUES"+str(i)+';')
```

[ ]:
```
sql="INSERT INTO ecommerce.delivery_agents(id_agent,company_name) VALUES(%s,%s)"
print(sql)
print(mycursor.executemany(sql,d_agents_list))
db.commit()
```

[ ]:
```
class_list=[]

def make_class():

    counter=0
    class_names=['Premium','Standard','Executive']
    id_class=['1','2','3']
    prices_ranges=['100-500','0-100','500>']

    for i in range(0,len(id_class)):
        single_class=()
        condition=True

        while condition==True:

            single_class=single_class+(id_class[counter]+'-C',)
            single_class=single_class+(class_names[counter],)
            single_class=single_class+(prices_ranges[counter],)
```

```
            class_list.append(single_class)

            counter+=1

            condition=False

make_class()


for e in class_list:
    print("INSERT INTO ecommerce.classes(id_class,class_name,prices_range)␣
 ↪VALUES"+str(e))
```

[ ]:
```
sql="INSERT INTO ecommerce.classes(id_class,class_name,prices_range)␣
 ↪VALUES(%s,%s,%s)"
print(sql)
print(mycursor.executemany(sql,class_list))
db.commit()
```

[ ]:
```

```