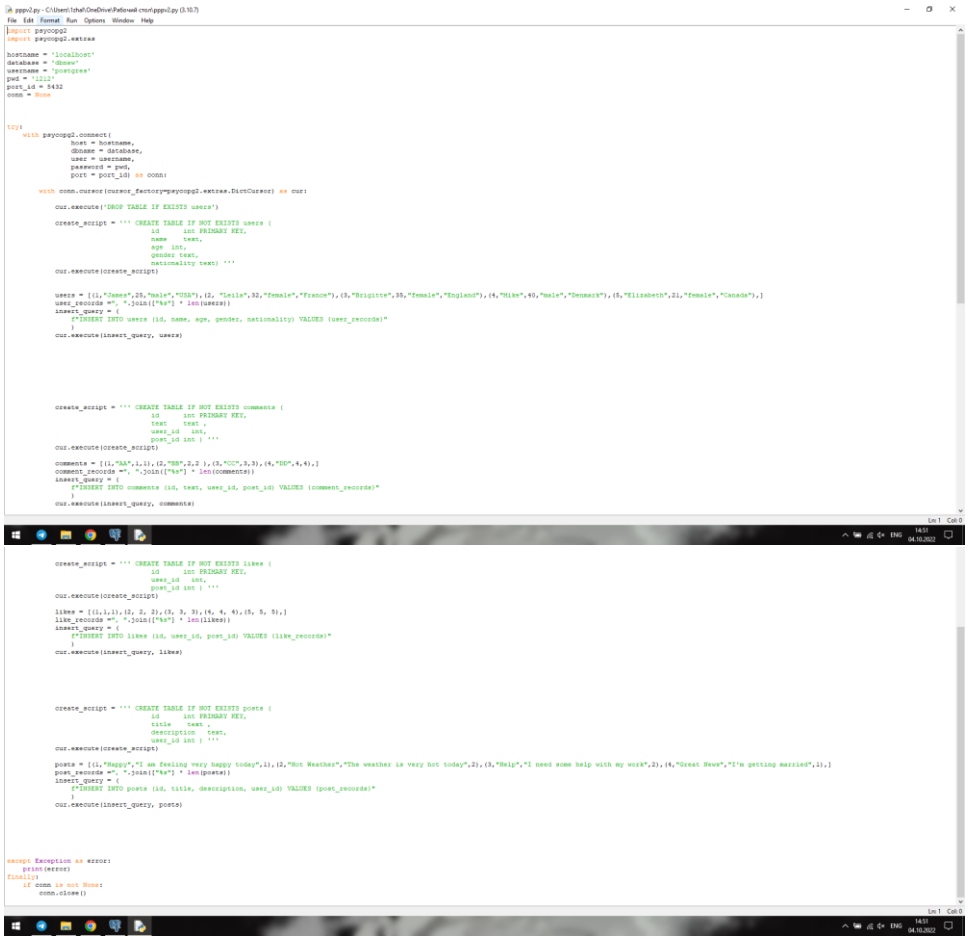

CODE:

```
import psycopg2
import psycopg2.extras


hostname = 'localhost'
database = 'dbnew'
username = 'postgres'
pwd = '1212'
port_id = 5432
conn = None
```

```
try:
    with psycopg2.connect(
            host = hostname,
            dbname = database,
            user = username,
            password = pwd,
            port = port_id) as conn:

        with conn.cursor(cursor_factory=psycopg2.extras.DictCursor) as cur:

            cur.execute('DROP TABLE IF EXISTS users')

            create_script = ''' CREATE TABLE IF NOT EXISTS users (
                                id      int PRIMARY KEY,
                                name    text,
                                age  int,
                                gender text,
                                nationality text) '''
            cur.execute(create_script)



            users = [(1,"James",25,"male","USA"),(2,
"Leila",32,"female","France"),(3,"Brigitte",35,"female","England"),(4,"Mike",40,"male","Denmark"),(
5,"Elizabeth",21,"female","Canada"),]
            user_records =", ".join(["%s"] * len(users))
            insert_query = (
                f"INSERT INTO users (id, name, age, gender, nationality) VALUES {user_records}"
                )
            cur.execute(insert_query, users)
```

```
create_script = ''' CREATE TABLE IF NOT EXISTS comments (
                    id      int PRIMARY KEY,
                    text    text ,
                    user_id   int,
                    post_id int ) '''
cur.execute(create_script)

comments = [(1,"AA",1,1),(2,"BB",2,2 ),(3,"CC",3,3),(4,"DD",4,4),]
comment_records =", ".join(["%s"] * len(comments))
insert_query = (
   f"INSERT INTO comments (id, text, user_id, post_id) VALUES {comment_records}"
   )
cur.execute(insert_query, comments)
```

```
create_script = ''' CREATE TABLE IF NOT EXISTS likes (
                    id      int PRIMARY KEY,
                    user_id   int,
                    post_id int ) '''
cur.execute(create_script)

likes = [(1,1,1),(2, 2, 2),(3, 3, 3),(4, 4, 4),(5, 5, 5),]
```

```
like_records =", ".join(["%s"] * len(likes))
insert_query = (
   f"INSERT INTO likes (id, user_id, post_id) VALUES {like_records}"
   )
cur.execute(insert_query, likes)
```

```
create_script = ''' CREATE TABLE IF NOT EXISTS posts (
                    id      int PRIMARY KEY,
                    title    text ,
                    description   text,
                    user_id int ) '''
cur.execute(create_script)
```

```
posts = [(1,"Happy","I am feeling very happy today",1),(2,"Hot Weather","The weather is very hot today",2),(3,"Help","I need some help with my work",2),(4,"Great News","I'm getting married",1),]
post_records =", ".join(["%s"] * len(posts))
insert_query = (
   f"INSERT INTO posts (id, title, description, user_id) VALUES {post_records}"
   )
cur.execute(insert_query, posts)
```

```
except Exception as error:
    print(error)
finally:
    if conn is not None:
        conn.close()
```