|  |
| --- |
| cur.execute(""" |
|  | CREATE TABLE IF NOT EXISTS course( |
|  | id SERIAL PRIMARY KEY, |
|  | name VARCHAR(40) UNIQUE |
|  | ); |
|  | """) |
|  | cur.execute(""" |
|  | CREATE TABLE IF NOT EXISTS homework( |
|  | id SERIAL PRIMARY KEY, |
|  | number INTEGER NOT NULL, |
|  | description TEXT NOT NULL, |
|  | course\_id INTEGER NOT NULL REFERENCES course(id) |
|  | ); |
|  | """) |
|  | conn.commit() # фиксируем в БД |
|  |  |
|  | # наполнение таблиц (C из CRUD) |
|  | cur.execute(""" |
|  | INSERT INTO course(name) VALUES('Python'); |
|  | """) |
|  | conn.commit() # фиксируем в БД |
|  |  |
|  | cur.execute(""" |
|  | INSERT INTO course(name) VALUES('Java') RETURNING id; |
|  | """) |
|  | print(cur.fetchone()) # запрос данных автоматически зафиксирует изменения |
|  |  |
|  | cur.execute(""" |
|  | INSERT INTO homework(number, description, course\_id) VALUES(1, 'простое дз', 1); |
|  | """) |
|  | conn.commit() # фиксируем в БД |
|  |  |
|  | # извлечение данных (R из CRUD) |
|  | cur.execute(""" |
|  | SELECT \* FROM course; |
|  | """) |
|  | print('fetchall', cur.fetchall()) # извлечь все строки |
|  |  |
|  | cur.execute(""" |
|  | SELECT \* FROM course; |
|  | """) |
|  | print(cur.fetchone()) # извлечь первую строку (аналог LIMIT 1) |
|  |  |
|  | cur.execute(""" |
|  | SELECT \* FROM course; |
|  | """) |
|  | print(cur.fetchmany(3)) # извлечь первые N строк (аналог LIMIT N) |
|  |  |
|  | cur.execute(""" |
|  | SELECT name FROM course; |
|  | """) |
|  | print(cur.fetchall()) |
|  |  |
|  | cur.execute(""" |
|  | SELECT id FROM course WHERE name='Python'; |
|  | """) |
|  | print(cur.fetchone()) |
|  |  |
|  | cur.execute(""" |
|  | SELECT id FROM course WHERE name='{}'; |
|  | """.format("Python")) # плохо - возможна SQL инъекция |
|  | print(cur.fetchone()) |
|  |  |
|  | cur.execute(""" |
|  | SELECT id FROM course WHERE name=%s; |
|  | """, ("Python",)) # хорошо, обратите внимание на кортеж |
|  | print(cur.fetchone()) |
|  |  |
|  | def get\_course\_id(cursor, name: str) -> int: |
|  | cursor.execute(""" |
|  | SELECT id FROM course WHERE name=%s; |
|  | """, (name,)) |
|  | return cur.fetchone()[0] |
|  | python\_id = get\_course\_id(cur, 'Python') |
|  | print('python\_id', python\_id) |
|  |  |
|  | cur.execute(""" |
|  | INSERT INTO homework(number, description, course\_id) VALUES(%s, %s, %s); |
|  | """, (2, "задание посложнее", python\_id)) |
|  | conn.commit() # фиксируем в БД |
|  |  |
|  | cur.execute(""" |
|  | SELECT \* FROM homework; |
|  | """) |
|  | print(cur.fetchall()) |
|  |  |
|  | # обновление данных (U из CRUD) |
|  | cur.execute(""" |
|  | UPDATE course SET name=%s WHERE id=%s; |
|  | """, ('Python Advanced', python\_id)) |
|  | cur.execute(""" |
|  | SELECT \* FROM course; |
|  | """) |
|  | print(cur.fetchall()) # запрос данных автоматически зафиксирует изменения |
|  |  |
|  | # удаление данных (D из CRUD) |
|  | cur.execute(""" |
|  | DELETE FROM homework WHERE id=%s; |
|  | """, (1,)) |
|  | cur.execute(""" |
|  | SELECT \* FROM homework; |
|  | """) |
|  | print(cur.fetchall()) # запрос данных автоматически зафиксирует изменения |