

Отчет по практической работе №15

Тема: составление программ с использованием регулярных выражений в IDE PyCharm Community.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ для работы с базами данных в IDE PyCharm Community.

Вариант 5

Тип алгоритма: базами данных

Текст программы:

```
import sqlite3 as sq
import info

with sq.connect('apteka.db') as con:
    cur = con.cursor()
    cur.execute("""CREATE TABLE IF NOT EXISTS lekarstvennie_sredstva (
        id_preparata INTEGER PRIMARY KEY,
        nazvanie_preparata VARCHAR,
        primeneniye VARCHAR,
        strana VARCHAR,
        cost FLOAT
    )""")

with sq.connect("apteka.db") as con:
    cur = con.cursor()
    cur.executemany("INSERT INTO lekarstvennie_sredstva VALUES (?, ?, ?, ?, ?)",
info.lekarstva)

with sq.connect('apteka.db') as con:
    cur = con.cursor()
    cur.execute("""CREATE TABLE IF NOT EXISTS nalichie_na_sklade (
        id_preparata INTEGER,
        kolichestvo INTEGER,
        date_ispolzovaniya DATE,
        FOREIGN KEY (id_preparata) REFERENCES lekarstvennie_sredstva
(id_preparata)
    )""")

with sq.connect("apteka.db") as con:
    cur = con.cursor()
    cur.executemany("INSERT INTO nalichie_na_sklade VALUES (?, ?, ?)",
info.nalichie)

with sq.connect('apteka.db') as con:
    cur = con.cursor()
```

```
cur.execute("""CREATE TABLE IF NOT EXISTS aptechniy_punkt (  
    id_punkt INTEGER PRIMARY KEY,  
    adress VARCHAR,  
    id_preparata INTEGER,  
    zayavki INTEGER,  
    date_zayavki DATE,  
    summa_zakaza FLOAT,  
    FOREIGN KEY (id_preparata) REFERENCES lekarstvennie_sredstva  
(id_preparata)  
)""")
```

```
with sq.connect("apteka.db") as con:  
    cur = con.cursor()  
    cur.executemany("INSERT INTO aptechniy_punkt VALUES (?, ?, ?, ?, ?, ?)",  
info.aptechniy_punkt)
```

1.1

Вывести список всех препаратов с указанием количества их наличия на складе.

```
# cur.execute("SELECT nazvanie_preparata, kolichestvo FROM  
lekarstvennie_sredstva INNER JOIN nalichie_na_sklade ON  
lekarstvennie_sredstva.id_preparata = nalichie_na_sklade.id_preparata")  
# result = cur.fetchall()  
# print(result)
```

1.2

**# Вывести список всех препаратов, имеющих на складе в количестве менее 10
штук.**

```
# cur.execute("SELECT nazvanie_preparata, kolichestvo FROM  
lekarstvennie_sredstva INNER JOIN nalichie_na_sklade ON  
lekarstvennie_sredstva.id_preparata = nalichie_na_sklade.id_preparata WHERE  
kolichestvo < 10")  
# result = cur.fetchall()  
# print(result)
```

1.3

Вывести список всех препаратов, которые производятся в России.

```
# cur.execute("SELECT nazvanie_preparata FROM lekarstvennie_sredstva  
WHERE strana = 'Россия')  
# result = cur.fetchall()  
# print(result)
```

1.4

**# Вывести список всех аптечных пунктов с указанием адреса и количества наличия
препаратов в каждом пункте.**

```
# cur.execute("SELECT adress, kolichestvo FROM aptechniy_punkt INNER JOIN  
nalichie_na_sklade ON aptechniy_punkt.id_preparata =  
nalichie_na_sklade.id_preparata")  
# result = cur.fetchall()  
# print(result)
```

1.5

**# Вывести список всех лекарственных препаратов, цена которых меньше
1000 руб.(12 долларов),
отсортированных по названию**

```
# cur.execute("SELECT * FROM lekarstvennie_sredstva WHERE cost < 12 ORDER  
BY nazvanie_preparata")  
# result = cur.fetchall()  
# print(result)
```

1.6

Вывести список лекарственных препаратов и их наличие на складе

```
# cur.execute("SELECT nazvanie_preparata, kolichestvo FROM  
lekarstvennie_sredstva INNER JOIN nalichie_na_sklade ON  
lekarstvennie_sredstva.id_preparata = nalichie_na_sklade.id_preparata")  
# result = cur.fetchall()  
# print(result)
```

1.7

**# Вывести список лекарственных препаратов, которые заканчиваются на
складе**

```
# cur.execute("SELECT nazvanie_preparata, kolichestvo FROM  
lekarstvennie_sredstva INNER JOIN nalichie_na_sklade ON  
lekarstvennie_sredstva.id_preparata = nalichie_na_sklade.id_preparata WHERE  
kolichestvo < 2")  
# result = cur.fetchall()  
# print(result)
```

1.8

**# Вывести список аптечных пунктов, в которых есть хотя бы одно
лекарственное
средство срок годности которого истекает в этом месяце.**

```
# cur.execute("SELECT adress FROM aptechniy_punkt INNER JOIN  
nalichie_na_sklade ON aptechniy_punkt.id_preparata =  
nalichie_na_sklade.id_preparata WHERE date_ispolzovaniya < '2023-04-01'")  
# result = cur.fetchall()  
# print(result)
```

2.1

Обновить количество препарата "Аспирин" на складе до 100 штук

```
# with sq.connect("apteka.db") as con:  
#     cur = con.cursor()  
#     cur.execute("UPDATE nalichie_na_sklade SET kolichestvo = 100 WHERE  
id_preparata LIKE '1001'")
```

2.2

Изменить дату использования для всех препаратов производства Индии

```
# with sq.connect("apteka.db") as con:  
#     cur = con.cursor()  
#     cur.execute("UPDATE nalichie_na_sklade SET date_ispolzovaniya = '2023-  
03-10' WHERE id_preparata LIKE '1007' OR id_preparata LIKE '1014'")
```

2.3

Увеличить цену препарата "Метформин" производства Германии на 20%:

```
# with sq.connect("apteka.db") as con:  
#     cur = con.cursor()  
#     cur.execute("UPDATE lekarstvennie_sredstva SET cost = cost * 1.2 WHERE  
id_preparata LIKE '1001' OR id_preparata LIKE '1013'")
```

2.4

Обновление цены для всех препаратов в таблице "Лекарственные средства" с учетом налога в размере 20%

```
# with sq.connect("apteka.db") as con:  
#     cur = con.cursor()  
#     cur.execute("UPDATE lekarstvennie_sredstva SET cost = cost * 1.2")
```

2.5

Обновление наличия препарата на складе в таблице "Наличие на складе" с учетом отгрузки некоторого количества препарата

```
# with sq.connect("apteka.db") as con:
```

```
# cur = con.cursor()
# cur.execute("UPDATE nalichie_na_sklade SET kolichestvo = 24 WHERE
id_preparata LIKE '1006'")
```

2.6

**# Обновление наличия препарата на складе в таблице "Наличие на складе" с учетом
поставки новых партий препарата:**

```
# with sq.connect("apteka.db") as con:
# cur = con.cursor()
# cur.execute("UPDATE nalichie_na_sklade SET kolichestvo = kolichestvo + 3
WHERE id_preparata LIKE '1008' OR id_preparata LIKE '1015'")
```

2.7

Обновление цены препарата в таблице "Лекарственные средства":

```
# with sq.connect("apteka.db") as con:
# cur = con.cursor()
# cur.execute("UPDATE lekarstvennie_sredstva SET cost = 121.2 WHERE
id_preparata LIKE '1010'")
```

2.8

Обновление даты использования препарата в таблице "Наличие на складе"

```
# with sq.connect("apteka.db") as con:
# cur = con.cursor()
# cur.execute("UPDATE nalichie_na_sklade SET date_ispolzovaniya = '2022-
09-10' WHERE id_preparata LIKE '1002'")
```

2.9

Обновление данных о препарате в таблице "Лекарственные средства"

```
# with sq.connect("apteka.db") as con:
# cur = con.cursor()
# cur.execute("UPDATE lekarstvennie_sredstva SET cost = 28.8 WHERE
id_preparata LIKE '1002'")
```

2.10

**# Обновление данных о препарате в таблице "Лекарственные средства" при изменении
наименования препарата:**

```
# with sq.connect("apteka.db") as con:
# cur = con.cursor()
```

```
# cur.execute("UPDATE lekarstvennie_sredstva SET nazvanie_preparata =  
'ЭвОдин' WHERE id_preparata LIKE '1009'")
```

2.11

Обновление цены препарата в таблице "Лекарственные средства"

```
# with sq.connect("apteka.db") as con:  
# cur = con.cursor()  
# cur.execute("UPDATE lekarstvennie_sredstva SET cost = 63.6 WHERE  
id_preparata LIKE '1005'")
```

2.12

**# Обновление данных о препарате в таблице "Лекарственные средства"
при изменении
страны-производителя**

```
# with sq.connect("apteka.db") as con:  
# cur = con.cursor()  
# cur.execute("UPDATE lekarstvennie_sredstva SET strana = 'Германия'  
WHERE id_preparata LIKE '1012'")
```

2.13

Обновление данных о препарате в таблице "Лекарственные средства"

```
# with sq.connect("apteka.db") as con:  
# cur = con.cursor()  
# cur.execute("UPDATE lekarstvennie_sredstva SET cost = 21.6 WHERE  
id_preparata LIKE '1004'")
```

3.1

Удалить все препараты, которые просрочены на дату 2021-09-30:

```
# with sq.connect("apteka.db") as con:  
# cur = con.cursor()  
# cur.execute("DELETE FROM nalichie_na_sklade WHERE date_ispolzovaniya <  
'2021-09-30'")
```

3.2

Удалить все препараты производства США с ценой ниже 10 долларов:

```
# with sq.connect("apteka.db") as con:  
# cur = con.cursor()  
# cur.execute("DELETE FROM lekarstvennie_sredstva WHERE strana = 'США'  
AND cost < 10")
```

3.3

Удалить все записи о заявках, где были заказаны препараты с ценой выше 50 долларов:

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM aptechniy_punkt WHERE summa_zakaza > 50")
```

3.4

Удалить записи о заявках для препаратов, которых нет на складе

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM aptechniy_punkt WHERE id_preparata IN  
(SELECT zayavki FROM aptechniy_punkt INNER JOIN nalichie_na_sklade ON  
aptechniy_punkt.id_preparata = nalichie_na_sklade.id_preparata WHERE  
kolichество = 0)")
```

3.5

Удалить все записи о препаратах производства России:

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM lekarstvennie_sredstva WHERE strana =  
'Россия'")
```

3.6

Удалить все записи о заявках, где было заказано более 10 единиц одного препарата

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM aptechniy_punkt WHERE zayavki > 10")
```

3.7

Удалить все записи о препаратах, которые не используются после даты 2021-10-01

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM nalichie_na_sklade WHERE date_ispolzovaniya <  
'2021-10-01'")
```

3.8

Удалить записи о препаратах, цена на которые была установлена выше 100 долларов:

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM lekarstvennie_sredstva WHERE cost > 100")
```

3.9

**# Удалить все записи о заявках, где были заказаны препараты,
произведенные в
Германии:**

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM aptechniy_punkt WHERE id_preparata IN  
(SELECT zayavki FROM aptechniy_punkt INNER JOIN lekarstvennie_sredstva ON  
aptechniy_punkt.id_preparata = lekarstvennie_sredstva.id_preparata WHERE  
strana = 'Германия')")
```

3.10

**# Удалить все записи о препаратах, название которых начинается на букву
"А":**

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM lekarstvennie_sredstva WHERE  
nazvanie_preparata LIKE 'A%'")
```

3.11

**# Удалить записи в таблице "Лекарственные
средства", которых нет в таблице "Аптечный пункт":**

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM lekarstvennie_sredstva WHERE id_preparata  
LIKE '1007' OR id_preparata LIKE '1013'")
```

3.12

**# Удалить все записи о препаратах, производимых в США и стоимостью
выше 50
долларов**

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM lekarstvennie_sredstva WHERE strana = 'США'  
AND cost > 50")
```

3.13

Удалить все записи о заявках в аптечных пунктах на препараты,

**произведенные в
России**

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM aptechniy_punkt WHERE id_preparata IN  
(SELECT zayavki FROM aptechniy_punkt INNER JOIN lekarstvennie_sredstva ON  
aptechniy_punkt.id_preparata = lekarstvennie_sredstva.id_preparata WHERE  
strana = 'Россия')")
```

3.14

Удалить все записи о препаратах, цена на которые не указана:

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM lekarstvennie_sredstva WHERE cost = 0")
```

3.15

**# Удалить записи о заявках для препаратов, которые отсутствуют на
складе**

```
# with sq.connect("apteka.db") as con:  
#   cur = con.cursor()  
#   cur.execute("DELETE FROM aptechniy_punkt WHERE id_preparata IN (SELECT  
zayavki FROM aptechniy_punkt INNER JOIN nalichie_na_sklade ON  
aptechniy_punkt.id_preparata = nalichie_na_sklade.id_preparata WHERE  
kolichество = 0)")
```

**Протокол работы программы: [('Аспирин', 34), ('Метформин', 2), ('Нурофен', 11),
('Прозак', 1), ('Терафлю', 3), ('Зодак', 25), ('Ацетаминофен', 1), ('Алпразолам', 6),
('Эводин', 23), ('Детралекс', 7), ('Кардиомагнил', 0), ('ЛинАква', 17), ('Ренгалин', 8),
('Адаксинт', 0), ('Ринза', 4), ('Завицефта', 9)]**

**[('Метформин', 2), ('Прозак', 1), ('Терафлю', 3), ('Ацетаминофен', 1), ('Алпразолам', 6),
('Детралекс', 7), ('Кардиомагнил', 0), ('Ренгалин', 8), ('Адаксинт', 0), ('Ринза', 4),
('Завицефта', 9)]**

[('Эводин',), ('Детралекс',), ('Ринза',)]

**[('ул. Волшебная, 44', 2), ('ул. Ленина, 12', 3), ('ул. Кирпичная, 31/5', 34), ('ул.
Просветления, 9', 23), ('ул. Лечебная, 3', 7)]**

**[(1014, 'Адаксинт', 'Способствует устранению избытка соляной кислоты', 'Индия', 10.0),
(1001, 'Аспирин', 'Противовоспалительное и жаропонижающее средство', 'Германия',
9.0), (1007, 'Ацетаминофен', 'Анальгетик и жаропонижающее средство', 'Индия', 1.0),
(1011, 'Кардиомагнил', 'Профилактика инфаркта и тромбоза с защитой желудка', 'США',
3.0), (1012, 'ЛинАква', 'Для промывания и орошения носа', 'Канада', 6.0), (1003,
'Нурофен', 'Противовоспалительное и жаропонижающее средство', 'Великобритания',
8.0), (1013, 'Ренгалин', 'Лечение кашля', 'Германия', 2.0), (1015, 'Ринза', 'При простуде и
гриппе', 'Россия', 7.0)]**

[('Аспирин', 34), ('Метформин', 2), ('Нурофен', 11), ('Прозак', 1), ('Терафлю', 3), ('Зодак', 25), ('Ацетаминофен', 1), ('Алпразолам', 6), ('Эведин', 23), ('Детралекс', 7), ('Кардиомагнил', 0), ('ЛинАква', 17), ('Ренгалин', 8), ('Адаксинт', 0), ('Ринза', 4), ('Завицефта', 9)]

[('Прозак', 1), ('Ацетаминофен', 1), ('Кардиомагнил', 0), ('Адаксинт', 0)]

[('ул. Волшебная, 44',), ('ул. Ленина, 12',), ('ул. Просветления, 9',), ('ул. Лечебная, 3',)]

Process finished with exit code 0

Вывод: В процессе выполнения практического занятия выработал приобрести навыки составление программ для работы с базами данных в IDE PyCharm Community.