Python

17

Join

JOIN - Операция соединения, предназначена для обеспечения выборки данных из двух таблиц и включения этих данных в один результирующий набор.

Создание данных для join

```
artist1 = Artist(name='Sunflower')
artist2 = Artist(name='Mushrooms')
artist3 = Artist(name='Trees')
album1 = Album(name='Only sun we love', artist=artist1)
album2 = Album(name='Only moon we love', artist=artist2)
album3 = Album(name='Only sun we love', artist=artist3)
session.add_all([artist1, artist2, artist3, album1, album2, album3])
session.commit()
```

Join sql

```sql
SELECT *
FROM artist JOIN album ON artist.id = album.artist_id
WHERE album.name = 'Only sun we love';
```

Join sqlalchemy

```
artists_albums = session.query(Artist, Album).join(
   Album, Artist.id == Album.artist_id).filter(
   Album.name == 'Only sun we love').all()
```

Задание 17.01

Создать книгу. Получить все группы, где есть студенты с именем Александр. Добавить всем полученным студентам новую книгу.

Задание 17.02

Создать три таблицы. Департамент(Department, name). Работник(Employee, firstname, lastname). Работник может быть без департамента.

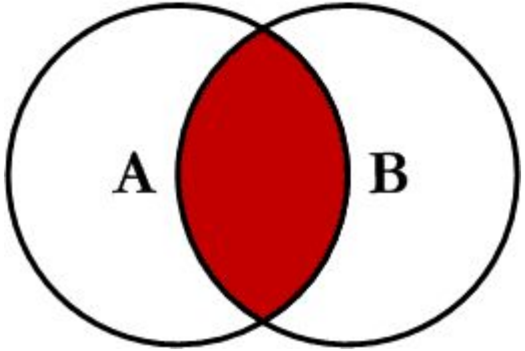
Пример

```sql
CREATE TABLE basket_a (
  id INT PRIMARY KEY,
  fruit VARCHAR (100) NOT NULL
);

INSERT INTO basket_a (id, fruit)
VALUES
  (1, 'Apple'),
  (2, 'Orange'),
  (3, 'Banana'),
  (4, 'Cucumber');
```

```sql
CREATE TABLE basket_b (
  id INT PRIMARY KEY,
  fruit VARCHAR (100) NOT NULL
);

INSERT INTO basket_b (id, fruit)
VALUES
  (1, 'Orange'),
  (2, 'Apple'),
  (3, 'Watermelon'),
  (4, 'Pear');
```

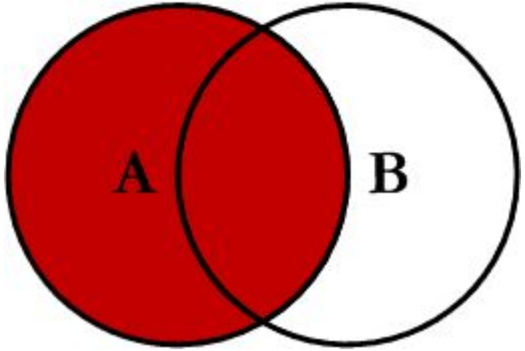
Inner join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
INNER JOIN basket_b b ON a.fruit = b.fruit;
```



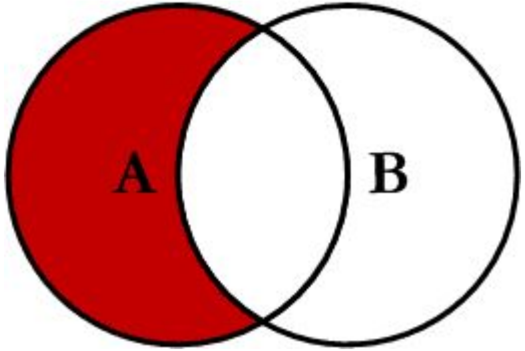
Left join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
LEFT JOIN basket_b b ON a.fruit = b.fruit;
```



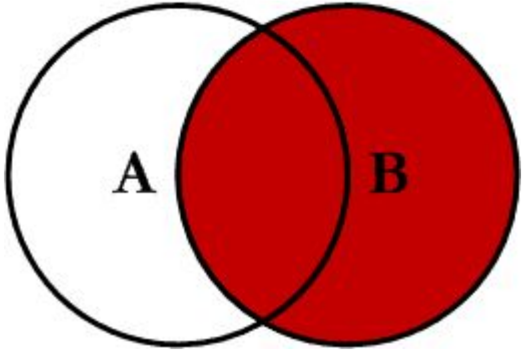
Left outer join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
LEFT JOIN basket_b b ON a.fruit = b.fruit
WHERE b.id IS NULL;
```



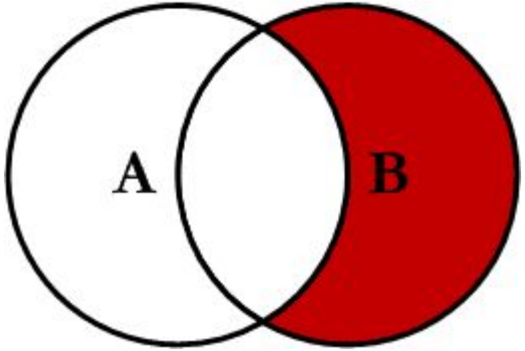
Right join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
RIGHT JOIN basket_b b ON a.fruit =
b.fruit;
```



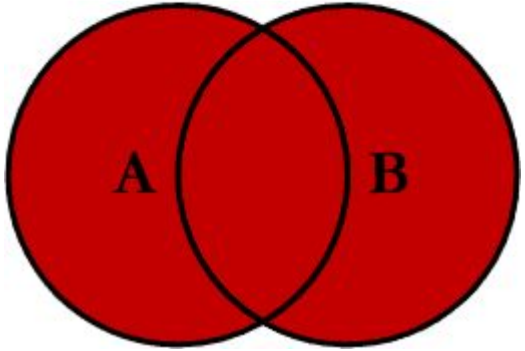
Right outer join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
RIGHT JOIN basket_b b ON a.fruit =
b.fruit
WHERE a.id IS NULL;
```



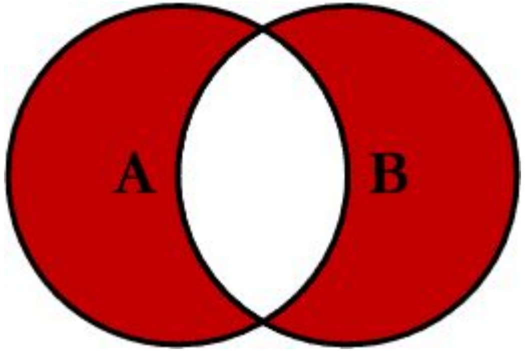
Full outer join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
FULL OUTER JOIN basket_b b ON a.fruit
= b.fruit;
```



Full outer excluding join

```
SELECT
   a.id id_a,
   a.fruit fruit_a,
   b.id id_b,
   b.fruit fruit_b
FROM
   basket_a a
FULL JOIN basket_b b ON a.fruit = b.fruit
 WHERE a.id IS NULL OR b.id IS NULL;
```



Cross join

```
SELECT
  a.id id_a,
  a.fruit fruit_a,
  b.id id_b,
  b.fruit fruit_b
FROM
  basket_a a
CROSS JOIN basket_b as b;
```