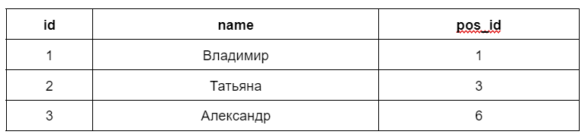


# SQL

**1) Написать запрос SQL.**

Есть 2 таблицы

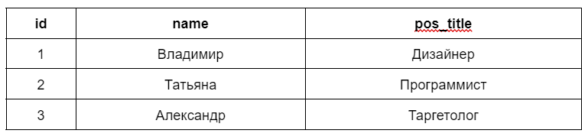
Persons (Сотрудники)



Positions (должности)



Написать запрос, чтобы получилась таблица вида:



**Ответ:**

SELECT

pers.id AS id,

name,

title AS pos\_title

FROM Person AS pers

LEFT JOIN Position AS pos ON pers.pos\_id = pos.id

**2) Дана пустая таблица publisher с колонками id и name. На id стоит автоинкремент, а Синтаксис name имеет тип varchar(40). Какие из следующих запросов корректно отработают:**

1. INSERT INTO publisher (name) VALUES ('OREILLY');
2. INSERT INTO publisher (name) VALUES ('О Reilly');
3. INSERT INTO publisher (name) VALUES ('O'Reilly');
4. INSERT INTO publisher (name) VALUES ('O-Reilly');

**Ответ: 1, 2, 4**

**3) Результат объединения запросов выведет все значения из таблицы tab?**

SELECT \* FROM tab WHERE b = 'a'

UNION

SELECT \* FROM tab WHERE с != 'a'

1. Да
2. Нет

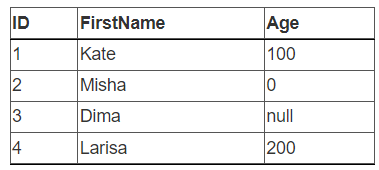
**Ответ: 2**

**4) С помощью какого update-запроса можно обновить значения более чем одной колонки?**

1. UPDATE table\_name SET col1\_name = 'col1val', col2\_name = 'col2val';
2. UPDATE table\_name SET col1\_name = 'col1val' AND SET col2\_name = 'col2val';
3. UPDATE table\_name SET col1\_name = 'col1val' SET col2\_name = 'col2val';
4. Одним запросом можно обновить не более одной колонки

**Ответ: 1**

**5) Дана таблица PERSONS:**



**Каков будет результат выполнения следующего запроса:**

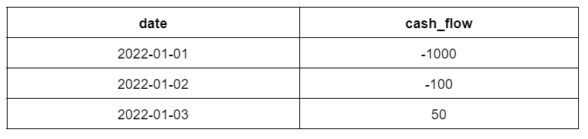
SELECT AVG(Age) FROM Persons

1. 150
2. 75
3. UNKNOWN, поскольку присутствует NULL
4. Запрос не выполнится (завершится ошибкой), поскольку присутствует NULL
5. 100

**Ответ: 5**

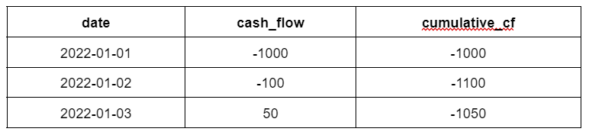
**6) Нарастающий итог**

Есть таблица transactions в таком виде:



Где cash\_flow — это выручка минус затраты за каждый день.

Нужно написать запрос, чтобы получить нарастающий итог для денежного потока каждый день таким образом, чтобы в конечном итоге получилась таблица в такой форме:



**Ответ:**

SELECT

date,

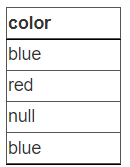
cash\_flow,

SUM(cash\_flow) OVER (ORDER BY date ASC) AS cumulative\_cf

FROM transactions

ORDER BY date

**7) Дана таблица CARS:**



**Каким будет результат следующего запроса:**

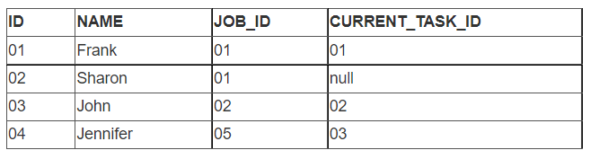
SELECT COUNT (DISTINCT color) FROM cars

1. red blue null
2. 3
3. 2 1 1
4. 2

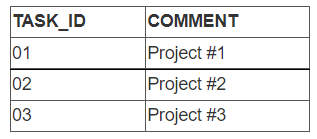
**Ответ: 2**

**8) Даны две таблицы:**

EMPLOYEES:



TASK:

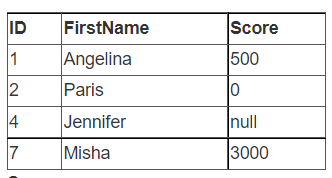


**Таблица EMPLOYEES имеет поле CURRENT\_TASK\_ID, которое является внешним ключом и ссылается на поле TASK\_ID таблицы TASKS. Вывести список, состоящий из имен всех имеющихся сотрудников и их текущих занятий.**

1. SELECT NAME, COMMENT FROM EMPLOYEES LEFT JOIN TASKS ON CURRENT\_TASK\_ID = TASK\_ID
2. SELECT NAME, COMMENT FROM EMPLOYEES, TASKS WHERE CURRENT\_TASK\_ID = TASK\_ID
3. SELECT NAME, COMMENT FROM EMPLOYEES, TASKS WHERE JOB\_ID = TASK\_ID
4. SELECT NAME, COMMENT FROM EMPLOYEES RIGHT JOIN TASKS ON CURRENT\_TASK\_ID = TASK\_ID

**Ответ: 1**

**9) Имеется таблица WOMEN**



**Сколько строк вернет запрос**

SELECT \* FROM Women

WHERE Score &gt;= ALL (SELECT Score

FROM Women

WHERE FirstName='Eva');

1. Запрос содержит ошибку и не выполнится
2. 2
3. 1
4. 3
5. 4

**Ответ: 5**

**10) Какая ошибка в этом запросе?**

UPDATE books SET sales\_1999 &gt;

(SELECT SUM(qty \* price)

FROM sales

WHERE sales.book\_id = books.id

AND sales.date BETWEEN '01/01/1999' AND '12/31/1999')

1. Вы не можете использовать арифметические операторы для функции SUM() (например SUM(qty \* price))
2. Следует использовать знак (=) для подзапроса
3. Вы не можете использовать подзапрос для SET внутри оператора UPDATE
4. Ключевое слово BETWEEN нельзя использовать для символов или дат

**Ответ: 2**