

1.

-- Пусть в таблице users поля created\_at и updated\_at оказались незаполненными.  
Заполните их текущими датой и временем.

```
SET SQL_SAFE_UPDATES=0;  
UPDATE users SET created_at = NOW(), updated_at = NOW();  
SET SQL_SAFE_UPDATES=1;
```

2.

/\*

Таблица users была неудачно спроектирована.

Записи created\_at и updated\_at были заданы типом VARCHAR и в них долгое время помещались значения в формате 20.10.2017 8:10.

Необходимо преобразовать поля к типу DATETIME, сохранив введенные ранее значения.

\*/

```
CREATE TABLE `users_copy` LIKE `users`;
```

```
ALTER TABLE `users_copy`  
CHANGE COLUMN `created_at` `created_at` DATETIME NOT NULL DEFAULT NOW(),  
CHANGE COLUMN `updated_at` `updated_at` DATETIME NOT NULL DEFAULT NOW();
```

```
INSERT INTO  
    `users_copy`  
SELECT  
    `id`,  
    `name`,  
    `birthday_at`,  
    STR_TO_DATE(`created_at`, '%d.%m.%Y %H:%i') AS `created_at`,  
    STR_TO_DATE(`updated_at`, '%d.%m.%Y %H:%i') AS `updated_at`  
FROM  
    `users`;
```

```
DROP TABLE `users`;  
ALTER TABLE `users_copy` RENAME `users`;
```

3.

/\*

В таблице складских запасов storehouses\_products в поле value могут встречаться самые разные цифры: 0,

если товар закончился и выше нуля, если на складе имеются запасы.

Необходимо отсортировать записи таким образом, чтобы они выводились в порядке увеличения значения value.

Однако нулевые запасы должны выводиться в конце, после всех записей.

\*/

SELECT

    `value`

FROM

    `storehouses\_products`

ORDER BY

    `value` = 0, `value`;

4.

/\*

(по желанию) Из таблицы users необходимо извлечь пользователей, родившихся в августе и мае.

Месяцы заданы в виде списка английских названий (may, august)

\*/

-- Вариант 1.

-- SELECT

-- \*

-- FROM

-- users

-- WHERE

-- CASE

-- WHEN DATE\_FORMAT(birthday\_at, '%m') = 5 THEN 'may'

-- WHEN DATE\_FORMAT(birthday\_at, '%m') = 8 THEN 'august'

-- END IN ('may', 'august')

-- Вариант 2.

SELECT

\*

FROM

users

WHERE

DATE\_FORMAT(birthday\_at, '%M') IN ('may', 'august')

5.

/\*

(по желанию) Из таблицы catalogs извлекаются записи при помощи запроса.

SELECT \* FROM catalogs WHERE id IN (5, 1, 2);

Отсортируйте записи в порядке, заданном в списке IN.

\*/

SELECT \* FROM catalogs WHERE id IN (5, 1, 2) ORDER BY id = 2, id = 1, id = 5;

## Практическое задание теме «Агрегация данных»

1.

-- Подсчитайте средний возраст пользователей в таблице users.

```
SELECT
    AVG(TIMESTAMPDIFF(YEAR, birthday_at, NOW()))
FROM
    users
```

2.

/\*

Подсчитайте количество дней рождения, которые приходятся на каждый из дней недели.

Следует учесть, что необходимы дни недели текущего года, а не года рождения

\*/

```
SELECT
    DATE_FORMAT(STR_TO_DATE(CONCAT(YEAR(CURRENT_DATE),
    DATE_FORMAT(birthday_at, '-%m-%d')), '%Y-%m-%d'), '%W') as week_day,
    COUNT(*)
FROM
    users
GROUP BY
    week_day
```

3.

-- (по желанию) Подсчитайте произведение чисел в столбце таблицы.

```
SELECT  
    ROUND(EXP(SUM(LOG(value))),1)  
FROM  
    test -- Таблица с тестовыми данными
```