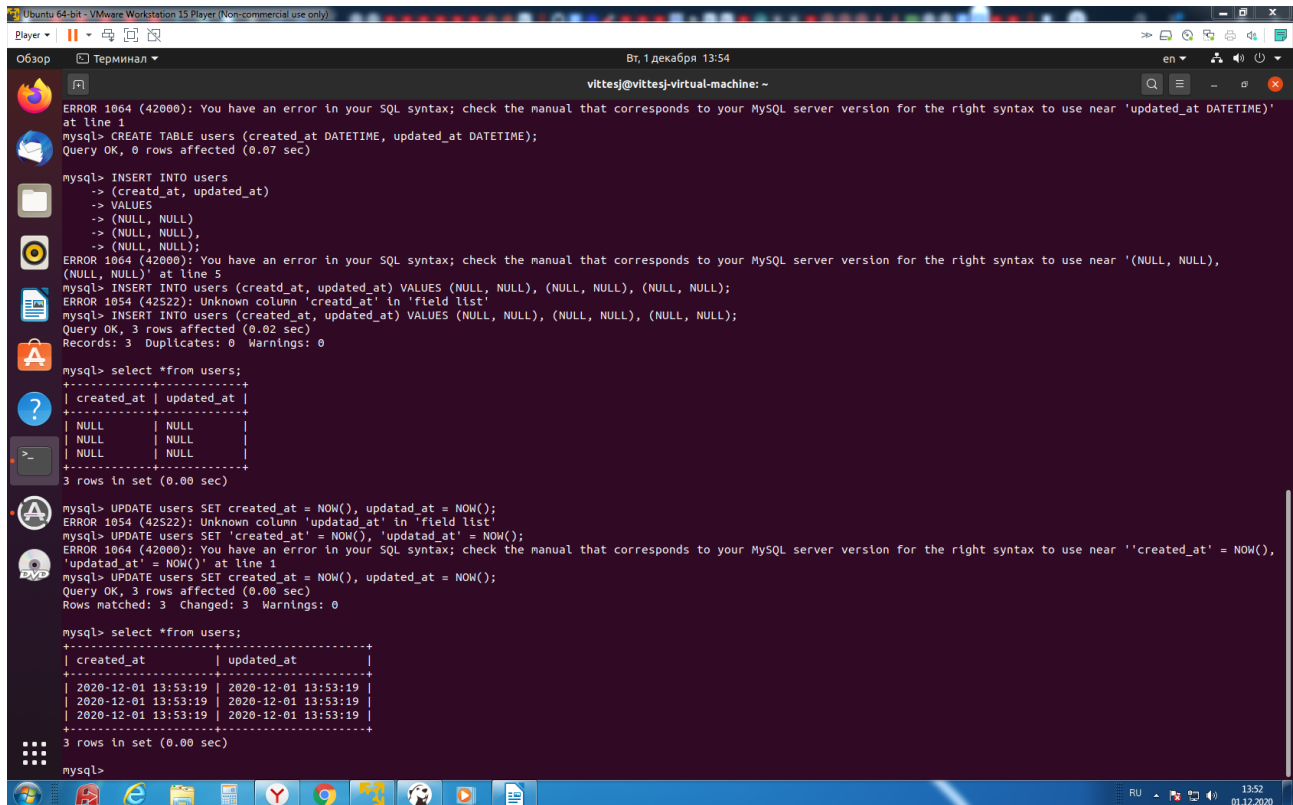


# Домашнее задание №5

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



```
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'updated_at DATETIME)' at line 1
mysql> CREATE TABLE users (created_at DATETIME, updated_at DATETIME);
Query OK, 0 rows affected (0.07 sec)

mysql> INSERT INTO users
-> (creatd_at, updated_at)
-> VALUES
-> (NULL, NULL),
-> (NULL, NULL),
-> (NULL, NULL);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '(NULL, NULL), (NULL, NULL)' at line 5
mysql> INSERT INTO users (creatd_at, updated_at) VALUES (NULL, NULL), (NULL, NULL), (NULL, NULL);
ERROR 1054 (42S22): Unknown column 'creatd_at' in 'field list'
mysql> INSERT INTO users (created_at, updated_at) VALUES (NULL, NULL), (NULL, NULL), (NULL, NULL);
Query OK, 3 rows affected (0.02 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from users;
+-----+-----+
| created_at | updated_at |
+-----+-----+
| NULL      | NULL      |
| NULL      | NULL      |
| NULL      | NULL      |
+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE users SET created_at = NOW(), updatad_at = NOW();
ERROR 1054 (42S22): Unknown column 'updatad_at' in 'field list'
mysql> UPDATE users SET 'created_at' = NOW(), 'updatad_at' = NOW();
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''created_at' = NOW(), 'updatad_at' = NOW()' at line 1
mysql> UPDATE users SET created_at = NOW(), updated_at = NOW();
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3 Changed: 3 Warnings: 0

mysql> select *from users;
+-----+-----+
| created_at | updated_at |
+-----+-----+
| 2020-12-01 13:53:19 | 2020-12-01 13:53:19 |
| 2020-12-01 13:53:19 | 2020-12-01 13:53:19 |
| 2020-12-01 13:53:19 | 2020-12-01 13:53:19 |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

Ранее мы создаем новую базу и таблицу занося туда нужные столбцы с неуказанными данными, после чего редактируем данные в столбцах

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.

```
Ubuntu 64-bit - VMware Workstation 15 Player (Non-commercial use only)
Player
Обзор Терминал
Br, 1 декабря 14:18
vitesj@vitesj-virtual-machine: ~

--> (created_at, updated_at)
--> VALUES
--> ('10.07.1990 1:30', '06.03.1991 8:30'),
--> ('11.07.1990 1:35', '06.03.1981 9:36'),
--> ('11.07.2004 1:35', '06.03.1975 9:36');
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> UPDATE users SET created_at = STR_TO_DATE(created_at, "%d.%m.%Y %k:%i");
--> UPDATE users SET created_at = STR_TO_DATE(created_at, "%d.%m.%Y %k:%i");
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'UPDATE users SET creat
ed_at = STR_TO_DATE(created_at, "%d.%m.%Y %k:%i")' at line 2
mysql> UPDATE users SET created_at = STR_TO_DATE(created_at, "%d.%m.%Y %k:%i"),
--> updated_at = STR_TO_DATE(updated_at, "%d.%m.%Y %k:%i");
Query OK, 3 rows affected (0.02 sec)
Rows matched: 3 Changed: 3 Warnings: 0

mysql> select *from users;
+-----+-----+
| created_at | updated_at |
+-----+-----+
| 1990-07-10 01:30:00 | 1991-03-06 08:30:00 |
| 1990-07-11 01:35:00 | 1981-03-06 09:36:00 |
| 2004-07-11 01:35:00 | 1975-03-06 09:36:00 |
+-----+-----+
3 rows in set (0.01 sec)

mysql> ALTER TABLE users MODIFY created_at DATETIME, updated_at DATETIME;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'updated_at DATETIME' a
t line 1
mysql> ALTER TABLE users MODIFY created_at DATETIME;
Query OK, 3 rows affected (0.08 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE users MODIFY updated_at DATETIME;
Query OK, 3 rows affected (0.07 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from users;
+-----+-----+
| created_at | updated_at |
+-----+-----+
| 1990-07-10 01:30:00 | 1991-03-06 08:30:00 |
| 1990-07-11 01:35:00 | 1981-03-06 09:36:00 |
| 2004-07-11 01:35:00 | 1975-03-06 09:36:00 |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

Источник который помог справиться с заданием <https://quares.ru/?id=100771>

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.

```
Ubuntu 64-bit - VMware Workstation 15 Player (Non-commercial use only)
Player
Обзор Терминал
Br, 1 декабря 16:19
vitesj@vitesj-virtual-machine: ~

0 |
2500 |
0 |
30 |
500 |
1 |
+-----+
6 rows in set (0.00 sec)

mysql> select * from storehouses_products ORDER BY value;
+-----+
| value |
+-----+
| 0 |
| 0 |
| 1 |
| 30 |
| 500 |
| 2500 |
+-----+
6 rows in set (0.00 sec)

mysql> select * from storehouses_products ORDER BY value, value = 0;
+-----+
| value |
+-----+
| 0 |
| 0 |
| 1 |
| 30 |
| 500 |
| 2500 |
+-----+
6 rows in set (0.00 sec)

mysql> select * from storehouses_products ORDER BY value=0, value;
+-----+
| value |
+-----+
| 1 |
| 30 |
| 500 |
| 2500 |
| 0 |
| 0 |
+-----+
6 rows in set (0.00 sec)

mysql>
```

## Агрегация

Подсчитайте средний возраст пользователей в таблице users.

```
Ubuntu 64-bit - VMware Workstation 15 Player (Non-commercial use only)
Player
Обзор Терминал
vitesj@vitesj-virtual-machine: ~
Br, 1 декабря 22:44
en
mysql> SELECT FLOOR(age) FROM users;
ERROR 1054 (42S22): Unknown column 'age' in 'field list'
mysql> select *from users;
+-----+-----+
| birthday_at | updated_at |
+-----+-----+
| 1990-07-10 01:30:00 | 1991-03-06 08:30:00 |
| 1990-07-11 01:35:00 | 1981-03-06 09:36:00 |
| 2004-07-11 01:35:00 | 1975-03-06 09:36:00 |
+-----+-----+
3 rows in set (0.00 sec)

mysql> SELECT COUNT(1) FROM users;
+-----+
| COUNT(1) |
+-----+
| 3 |
+-----+
1 row in set (0.04 sec)

mysql> SELECT floor((TO_DAYS(NOW())-TO_DAYS(birthday_at)) / 365.25)/COUNT(1) FROM users;
ERROR 1140 (42000): In aggregated query without GROUP BY, expression #1 of SELECT list contains nonaggregated column 'sample.users.birthday_at'; this is incompatible with sql_m
de=only_full_group_by
mysql> SELECT FLOOR((TO_DAYS(NOW())-TO_DAYS(birthday_at)) / 365.25)/COUNT(1) FR
OM users;
ERROR 1140 (42000): In aggregated query without GROUP BY, expression #1 of SELECT list contains nonaggregated column 'sample.users.birthday_at'; this is incompatible with sql_m
de=only_full_group_by
mysql> SELECT FLOOR((TO_DAYS(NOW())-TO_DAYS(birthday_at)) / 365.25) AS age FROM users;
+-----+
| age |
+-----+
| 30 |
| 30 |
| 16 |
+-----+
3 rows in set (0.00 sec)

mysql> SELECT COUNT(1) FROM age;
ERROR 1146 (42S02): Table 'sample.age' doesn't exist
mysql> SELECT AVG(TIMESTAMPDIFF(YEAR, birthday_at, NOW())) AS age FROM users;
+-----+
| age |
+-----+
| 25.3333 |
+-----+
1 row in set (0.00 sec)

mysql>
```

И у меня возник вопрос почему не работает данная команда  
`SELECT FLOOR((TO_DAYS(NOW())-TO_DAYS(birthday_at)) / 365.25)/COUNT(1) FROM users;`  
вроде бы по логики тут все верно, если не затруднит ,дайте пожалуйста пояснение к этому.

Подсчитайте количество дней рождения, которые приходятся на каждый из дней недели. Следует учесть, что необходимы дни недели текущего года, а не года рождения.

Сначала вырезал из таблицы users базы vk столбец birthday, далее работаем только с ним:

Ubuntu 64-bit - VMware Workstation 15 Player (Non-commercial use only)

Player ▾ | Cp, 2 декабря 23:39 | en | 23:38 02.12.2020

Обзор | Терминал ▾ | vitesj@vitesj-virtual-machine: ~

```
mysql> select * from birth limit 10;
+-----+
| birthday |
+-----+
| 1986-12-21 |
| 1983-07-28 |
| 1982-12-24 |
| 1974-02-05 |
| 2009-12-11 |
| 1970-02-08 |
| 1977-04-14 |
| 2016-01-12 |
| 1985-02-04 |
| 1971-01-21 |
+-----+
10 rows in set (0.01 sec)

mysql> SELECT DAYNAME(CONCAT(YEAR(NOW()), '-', SUBSTRING(birthday, 6, 10))) AS weekday, COUNT(DAYNAME(CONCAT(YEAR(NOW()), '-', SUBSTRING(birthday, 6, 10)))) AS amount_of_birthday FROM birth GROUP BY weekday;
+-----+-----+
| weekday | amount_of_birthday |
+-----+-----+
| Monday | 13 |
| Tuesday | 16 |
| Thursday | 16 |
| Wednesday | 7 |
| Friday | 18 |
| Saturday | 11 |
| Sunday | 19 |
+-----+-----+
7 rows in set (0.00 sec)

mysql> SELECT birthday, CONCAT(YEAR(NOW()), '-', SUBSTRING(birthday, 6, 10)) AS in_year, DAYNAME(CONCAT(YEAR(NOW()), '-', SUBSTRING(birthday, 6, 10))) AS weekday FROM birth ORDER BY weekday LIMIT 10;
+-----+-----+-----+
| birthday | in_year | weekday |
+-----+-----+-----+
| 1970-06-26 | 2020-06-26 | Friday |
| 1970-04-10 | 2020-04-10 | Friday |
| 2007-07-24 | 2020-07-24 | Friday |
| 1988-07-24 | 2020-07-24 | Friday |
| 2009-12-11 | 2020-12-11 | Friday |
| 1981-11-13 | 2020-11-13 | Friday |
| 1999-07-17 | 2020-07-17 | Friday |
| 1989-09-18 | 2020-09-18 | Friday |
| 1999-03-27 | 2020-03-27 | Friday |
| 2017-08-14 | 2020-08-14 | Friday |
+-----+-----+-----+
```

вот еще данные

Ubuntu 64-bit - VMware Workstation 15 Player (Non-commercial use only)

Player ▾ | Cp, 2 декабря 23:40 | en | 23:39 02.12.2020

Обзор | Терминал ▾ | vitesj@vitesj-virtual-machine: ~

```
+-----+-----+-----+
| birthday | in_year | weekday |
+-----+-----+-----+
| 2009-12-11 | 2020-12-11 | Friday |
| 1988-07-24 | 2020-07-24 | Friday |
| 1999-07-17 | 2020-07-17 | Friday |
| 2015-03-16 | 2020-03-16 | Monday |
| 2000-03-23 | 2020-03-23 | Monday |
| 1972-06-22 | 2020-06-22 | Monday |
| 2010-01-13 | 2020-01-13 | Monday |
| 1986-12-21 | 2020-12-21 | Monday |
| 2005-03-16 | 2020-03-16 | Monday |
| 1989-12-21 | 2020-12-21 | Monday |
| 2018-06-22 | 2020-06-22 | Monday |
| 1989-01-20 | 2020-01-20 | Monday |
| 1988-08-17 | 2020-08-17 | Monday |
| 1996-08-31 | 2020-08-31 | Monday |
| 1978-07-06 | 2020-07-06 | Monday |
| 1985-06-08 | 2020-06-08 | Monday |
| 2015-03-07 | 2020-03-07 | Saturday |
| 1971-06-20 | 2020-06-20 | Saturday |
| 1975-06-13 | 2020-06-13 | Saturday |
| 1977-07-18 | 2020-07-18 | Saturday |
| 1970-02-08 | 2020-02-08 | Saturday |
| 2015-04-11 | 2020-04-11 | Saturday |
| 1985-07-18 | 2020-07-18 | Saturday |
| 1992-11-28 | 2020-11-28 | Saturday |
| 1998-10-10 | 2020-10-10 | Saturday |
| 2007-09-26 | 2020-09-26 | Saturday |
| 2015-05-23 | 2020-05-23 | Saturday |
| 1986-03-01 | 2020-03-01 | Sunday |
| 2006-08-02 | 2020-08-02 | Sunday |
| 2016-07-26 | 2020-07-26 | Sunday |
| 1994-01-26 | 2020-01-26 | Sunday |
| 2019-02-23 | 2020-02-23 | Sunday |
| 2011-09-27 | 2020-09-27 | Sunday |
| 2014-02-09 | 2020-02-09 | Sunday |
| 1981-05-31 | 2020-05-31 | Sunday |
| 2016-01-12 | 2020-01-12 | Sunday |
| 2020-11-01 | 2020-11-01 | Sunday |
| 1996-08-23 | 2020-08-23 | Sunday |
| 1991-12-13 | 2020-12-13 | Sunday |
| 2020-10-04 | 2020-10-04 | Sunday |
| 1975-02-02 | 2020-02-02 | Sunday |
| 2019-06-28 | 2020-06-28 | Sunday |
| 2005-05-10 | 2020-05-10 | Sunday |
| 1991-05-10 | 2020-05-10 | Sunday |
| 1978-04-19 | 2020-04-19 | Sunday |
| 1979-10-11 | 2020-10-11 | Sunday |
| 2012-03-05 | 2020-03-05 | Thursday |
| 1996-07-09 | 2020-07-09 | Thursday |
| 2016-06-18 | 2020-06-18 | Thursday |
| 1986-12-18 | 2020-12-18 | Thursday |
+-----+-----+-----+
```