

1. Пусть в таблице users поля created_at и updated_at оказались незаполненными. Заполните их текущими датой и временем.

```
| 90 | Bonnie | Schulist | lonny.daugherty@example.com | +44(2)6849631686 | 2004-10-06 23:08:58 | 2017-05-29 09:49:57 |
| 91 | Dallas | Lowe | oberbrunner.carli@example.org | 667-443-9007x4216 | 1991-10-15 22:29:46 | 1989-07-19 18:55:32 |
| 92 | Giles | Kunze | sienna.murazik@example.com | 199.555.3264x47103 | 2015-07-12 14:31:32 | 2009-09-29 00:26:05 |
| 93 | Berniece | Ziemann | general72@example.net | 1-165-566-1777 | 1975-06-07 12:28:23 | 2007-09-03 23:19:01 |
| 94 | Evelline | Schoen | phyllis68@example.net | 212.362.4314x001 | 2019-07-28 20:44:39 | 2019-01-14 16:57:25 |
| 95 | Dina | xtilman@example.net | 975-594-7338 | 2004-01-23 15:38:03 | 1991-05-10 14:12:01 |
| 96 | Ismael | Larkin | gusikowski.liana@example.com | 113-023-9005x3021 | 1975-09-04 01:41:55 | 2006-04-19 16:01:46 |
| 97 | Tony | Abshire | danya19@example.org | +39(8)2764321913 | 1980-07-02 18:52:48 | 1977-03-20 00:07:54 |
| 98 | Meredith | Murazik | dana67@example.org | 226.290.7975 | 1978-10-27 08:53:38 | 1979-08-31 22:48:56 |
| 99 | America | McGlynn | aniya.vandervort@example.com | 611.064.0979x7655 | 2008-03-29 04:16:29 | 2003-04-05 15:46:36 |
| 100 | Enrique | Farrell | qhoppe@example.org | 885.670.7028x3507 | 2016-02-18 09:48:48 | 1983-05-20 21:46:46 |
+-----+-----+-----+-----+-----+-----+-----+
100 rows in set (0,00 sec)

mysql> /* база
/*> заполнена
/*> сначала почистим ее
/*> , а у ж после заполним*/;
DROP;

No query specified

mysql> update users set created_at = null and updated_at = null;
Query OK, 0 rows affected (0,00 sec)

Rows matched: 100 Changed: 0 Warnings: 0

mysql> pdate users set created_at = case when created_at is null then now() else created_at end , updated_at = case when updated_at is null then now() else updated_at end;
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 'pdate users set creati
mysql> update users set created_at = case when created_at is null then now() else created_at end , updated_at = case when updated_at is null then now() else updated_at end;
Query OK, 0 rows affected (0,00 sec)

Rows matched: 100 Changed: 0 Warnings: 0

mysql> select * from users limit 10;
+----+-----+-----+-----+-----+-----+-----+
| id | firstname | lastname | email | phone | created_at | updated_at |
+----+-----+-----+-----+-----+-----+-----+
| 1 | Tevin | Mills | connelly.richie@example.com | (371)706-6924 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 2 | Sadie | Mueller | bednar.edison@example.org | 863-332-4448x077 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 3 | Abbie | Donnelly | bode.abelardo@example.com | 1-578-663-4064x2634 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 4 | Hardy | Davis | haven.bednar@example.org | 1-379-740-3408x6895 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 5 | Lon | Kertzmann | bernard.hane@example.net | 691-079-8993x507 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 6 | Lyric | Smith | ymedhurst@example.org | (577)533-0697x866 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 7 | Isaac | Senger | ruthe31@example.com | 988-201-9787x43463 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 8 | Julianne | Wolff | lhartmann@example.com | 845-602-2913x86437 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 9 | Jerry | Smitham | kheidenreich@example.com | 111-836-4622x1450 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
| 10 | Carmen | Schroeder | juanita47@example.org | 885-483-2195 | 2019-08-28 13:31:21 | 2019-08-28 13:30:34 |
+----+-----+-----+-----+-----+-----+-----+
10 rows in set (0,00 sec)
```

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.

```
[mysql> select * from users limit 10;
```

| | id | firstname | lastname | email | phone | created_at | updated_at |
|----|----|-----------|-----------|-----------------------------|---------------------|------------------|-----------------|
| 1 | 1 | Tevin | Mills | connelly.richie@example.com | (371)706-6924 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 2 | 2 | Sadie | Mueller | bednar.edison@example.org | 863-332-4448x077 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 3 | 3 | Abbie | Donnelly | bode.abelardo@example.com | 1-578-663-4064x2634 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 4 | 4 | Hardy | Davis | haven.bednar@example.org | 1-379-740-3408x6895 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 5 | 5 | Lon | Kertzmamn | bernard.hane@example.net | 691-079-8993x507 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 6 | 6 | Lyric | Smith | ymedhurst@example.org | (577)533-0597x866 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 7 | 7 | Isaac | Senger | ruthe31@example.com | 988-201-9787x43463 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 8 | 8 | Julianne | Wolff | lhartmann@example.com | 845-602-2913x86437 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 9 | 9 | Jerry | Smitham | kheidenreich@example.com | 111-836-4622x1450 | 28.08.2019 13:31 | 28.08.2019 13:5 |
| 10 | 10 | Carmen | Schroeder | juanita47@example.org | 885-483-2195 | 28.08.2019 13:31 | 28.08.2019 13:5 |

```
10 rows in set (0,00 sec)
```

```
[mysql> CREATE TABLE users_clone LIKE users;
```

```
Query OK, 0 rows affected (0,02 sec)
```

```
[mysql> insert into users_clone select * from users;
```

```
Query OK, 100 rows affected (0,01 sec)
```

```
Records: 100 Duplicates: 0 Warnings: 0
```

```
[mysql> truncate table users;
```

```
Query OK, 0 rows affected (0,02 sec)
```

```
mysql> ALTER TABLE users MODIFY created_at datetime DEFAULT CURRENT_TIMESTAMP;
```

```
Query OK, 0 rows affected (0,01 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql>
```

```
[mysql> ALTER TABLE users MODIFY updated_at datetime DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
```

```
Query OK, 0 rows affected (0,01 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
[mysql> insert into users select id, firstname, lastname, email, phone, STR_TO_DATE (created_at, '%d.%m.%Y %H:%i' ), STR_TO_DATE (updated_at, '%d.%m.%Y %H:%i' ) from users_clone;
```

```
Query OK, 100 rows affected (0,01 sec)
```

```
Records: 100 Duplicates: 0 Warnings: 0
```

```
[mysql> drop table users_clone;
```

```
Query OK, 0 rows affected (0,01 sec)
```

```
[mysql> select * from users limit 10;
```

```
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 '10' at line 1
```

```
[mysql> select * from users limit 10;
```

| | id | firstname | lastname | email | phone | created_at | updated_at |
|----|----|-----------|-----------|-----------------------------|---------------------|---------------------|---------------------|
| 1 | 1 | Tevin | Mills | connelly.richie@example.com | (371)706-6924 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 2 | 2 | Sadie | Mueller | bednar.edison@example.org | 863-332-4448x077 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 3 | 3 | Abbie | Donnelly | bode.abelardo@example.com | 1-578-663-4064x2634 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 4 | 4 | Hardy | Davis | haven.bednar@example.org | 1-379-740-3408x6895 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 5 | 5 | Lon | Kertzmamn | bernard.hane@example.net | 691-079-8993x507 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 6 | 6 | Lyric | Smith | ymedhurst@example.org | (577)533-0597x866 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 7 | 7 | Isaac | Senger | ruthe31@example.com | 988-201-9787x43463 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 8 | 8 | Julianne | Wolff | lhartmann@example.com | 845-602-2913x86437 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 9 | 9 | Jerry | Smitham | kheidenreich@example.com | 111-836-4622x1450 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |
| 10 | 10 | Carmen | Schroeder | juanita47@example.org | 885-483-2195 | 2019-08-28 13:31:00 | 2019-08-28 13:50:00 |

```
10 rows in set (0,00 sec)
```

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.

Взял таблицу с похожими значениями:

```
mysql> select help_keyword_id,name from help_keyword where help_keyword_id > 0 u
nion all select help_keyword_id,name from help_keyword where help_keyword_id = 0;
```

| help_keyword_id | name |
|-----------------|-----------------|
| 1 | HELP_VERSION |
| 2 | DEFAULT |
| 3 | SERIAL |
| 4 | VALUE |
| 5 | HELP |
| 6 | FALSE |
| 7 | TRUE |
| 8 | BOOL |
| 9 | BOOLEAN |
| 10 | INT1 |
| 11 | UNSIGNED |
| 12 | ZEROFILL |
| 697 | EXPORT |
| 698 | FLUSH |
| 699 | GENERAL |
| 700 | OPTIMIZER_COSTS |
| 701 | RELAY |
| 702 | SLOW |
| 703 | USER_RESOURCES |
| 704 | KILL |
| 705 | LEAVES |
| 706 | RESTART |
| 707 | DESCRIBE |
| 708 | EXPLAIN |
| 709 | FORMAT |
| 710 | TRADITIONAL |
| 0 | HELP_DATE |

711 rows in set (0,00 sec)

```
select * from storehouses_products where value > 0
union all
select * from storehouses_products where value = 0;
```

4. (по желанию) Из таблицы users необходимо извлечь пользователей, родившихся в августе и мае. Месяцы заданы в виде списка английских названий ('may', 'august')

Даны 2 варианта ответа ниже, если месяц непосредственно в поле то обращаемся к полю и и ищем по подстроке.

```
[mysql> select * from users;
```

| id | name | birthday_at | created_at | updated_at |
|----|-----------|-------------|---------------------|---------------------|
| 1 | Геннадий | 1990-10-05 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |
| 2 | Наталья | 1984-11-12 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |
| 3 | Александр | 1985-05-20 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |
| 4 | Сергей | 1988-02-14 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |
| 5 | Иван | 1998-01-12 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |
| 6 | Мария | 1992-08-29 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |

6 rows in set (0,00 sec)

```
[mysql> Select * from users where month(birthday_at )=8  
-> ;
```

| id | name | birthday_at | created_at | updated_at |
|----|-------|-------------|---------------------|---------------------|
| 6 | Мария | 1992-08-29 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |

1 row in set (0,00 sec)

```
[mysql> Select * from users where MONTHNAME(birthday_at )="August";
```

| id | name | birthday_at | created_at | updated_at |
|----|-------|-------------|---------------------|---------------------|
| 6 | Мария | 1992-08-29 | 2019-08-28 14:50:17 | 2019-08-28 14:50:17 |

1 row in set (0,00 sec)

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.

```
[mysql>
```

```
[mysql> SELECT * FROM catalogs WHERE id IN (5,1,2) ORDER BY FIELD(id,5,1,2);
```

| id | name |
|----|--------------------|
| 5 | Оперативная память |
| 1 | Процессоры |
| 2 | Материнские платы |

3 rows in set (0,00 sec)

Практическое задание теме “Агрегация данных”

1. Подсчитайте средний возраст пользователей в таблице users

```
mysql> SELECT YEAR(NOW()) - YEAR(birthday_at) from users;
```

| YEAR(NOW()) - YEAR(birthday_at) |
|---------------------------------|
| 29 |
| 35 |
| 34 |
| 31 |
| 21 |
| 27 |

6 rows in set (0,00 sec)

```
mysql> SELECT round( AVG( YEAR(NOW()) - YEAR(birthday_at))) from users;
```

| round(AVG(YEAR(NOW()) - YEAR(birthday_at))) |
|---|
| 30 |

1 row in set (0,00 sec)

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

```
mysql> Select _d, count(*)
-> from (
-> SELECT DAYNAME(
-> DATE_ADD(
-> birthday_at, INTERVAL YEAR(NOW()) - YEAR(birthday_at) YEAR)
-> ) as _d
-> from users
-> ) as day
-> group by _d;
```

| _d | count(*) |
|----------|----------|
| Saturday | 2 |
| Tuesday | 1 |
| Monday | 1 |
| Thursday | 2 |

4 rows in set (0,00 sec)

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

```
mysql> Select * from num
-> ;
```

| id |
|----|
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |

5 rows in set (0,00 sec)

```
mysql> SELECT @I:=@I*val Num
-> FROM
-> (SELECT id as val from num)a,
-> (SELECT @I := 1)I
-> ORDER BY num desc
-> Limit 1;
```

| Num |
|-----|
| 120 |

1 row in set, 2 warnings (0,00 sec)