**3. СОЗДАНИЕ ТАБЛИЦ**

**3.1. СОЗДАНИЕ ПРОСТЕЙШИХ ТАБЛИЦ**

**Задача 3.1.1 Таблица users**

Создайте таблицу **users** с полем **id** типа **INT** и двумя текстовыми полями, которые будут хранить имя (**first\_name**) и фамилию (**last\_name**). Длина имени и фамилии не превышает 50 символов.

Добавьте в таблицу трех пользователей: Дмитрия Иванова, Анатолия Белого и Дениса Давыдова.

Таблица users (итоговая таблица)

|  |  |  |
| --- | --- | --- |
| **id** | **first\_name** | **last\_name** |
| 1 | Дмитрий | Иванов |
| 2 | Анатолий | Белый |
| 3 | Денис | Давыдов |

**Примечания к задачам на создание таблиц**

Раньше для получения или изменения данных вы вставляли в редактор один SQL-запрос.  
В задачах на создание таблиц нужно одновременно выполнять **несколько SQL-запросов**:  
1. Создание таблицы (**CREATE TABLE**).  
2. Наполнение таблицы данными (**INSERT**).

Несколько SQL запросов, которые следуют друг за другом, нужно разделять точкой с запятой.  
Так запросы из урока «Создание простейших таблиц» могут выглядеть так:

CREATE TABLE users2 (

id INT,

first\_name VARCHAR(20),

last\_name VARCHAR(50),

birthday DATE,

age TINYINT,

active BOOL

);

INSERT INTO users2 (id, first\_name, last\_name, birthday, age, active)

VALUES (1, 'Дмитрий', 'Иванов', '1986-02-18', 31, True);

**Задача 3.1.2 Таблица orders**

Создайте таблицу **orders** с полем **id типа**INT, полем **state** для хранения статуса заказа и полем **amount** для хранения суммы заказа. Статус заказа умещается в строку в 10 символов, а сумма заказа имеет целочисленный тип данных.na

Заполните таблицу тремя заказами:

1. со статусом **new** на сумму 10000 рублей;
2. со статусом **new** на сумму 3400 рублей;
3. со статусом **delivery** на сумму 7300 рублей.

Таблица orders

|  |  |  |
| --- | --- | --- |
| **id** | **state** | **amount** |
| 1 | new | 10000 |
| 2 | new | 3400 |
| 3 | delivery | 7300 |

**Задача 3.1.3 Дни рождения**

Создайте таблицу **users** с полем **id** типа **INT** и двумя строковыми полями, которые будут хранить имя (**first\_name**) и фамилию (**last\_name**). Длина имени не превышает 20 символов, а фамилии 50 символов. Также добавьте в таблицу поле **birthday** типа **DATE**.

Заполните таблицу тремя пользователями:

1. Дмитрий Иванов, 12 августа 1995
2. Светлана Демчук, 8 июля 1993
3. Денис Антонов, 23 декабря 1996

Таблица users (итоговая таблица)

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **birthday** |
| 1 | Дмитрий | Иванов | 1995-08-12 |
| 2 | Светлана | Демчук | 1993-07-08 |
| 3 | Денис | Антонов | 1996-12-23 |

**Задача 3.1.4 Таблица messages**

Создайте таблицу **messages** с со следующими полями:

1. **id** типа **INT**;
2. **subject** типа **VARCHAR** длиной 100 символов
3. **message** тип **TEXT**;
4. **add\_date** типа **DATETIME**;
5. **is\_public** логического типа.

Добавьте в таблицу сообщение с темой «Первое сообщение» и текстом «Это мое первое сообщение!». Дату установите 12 декабря 2016 года 14 часов, 16 минут. Сообщение должно быть публичным.

Таблица messages (итоговая таблица)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **subject** | **message** | **add\_date** | **is\_public** |
| 1 | Первое сообщение | Это мое первое сообщение! | 2016-12-12 14:16:00 | 1 |

**Задача 3.1.5 Таблица rating**

Создайте таблицу **rating**

1. **id** типа **INT**;
2. **car\_id** типа **INT**;
3. **user\_id** типа **INT**;
4. **rating** типа **FLOAT**.

Добавьте в неё 4 записи так, чтобы получалась таблица ниже:

Таблица rating

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **car\_id** | **user\_id** | **rating** |
| 1 | 1 | 1 | 4.54 |
| 2 | 1 | 2 | 3.34 |
| 3 | 2 | 3 | 4.19 |
| 4 | 2 | 11 | 1.12 |

**3.2 ЧИСЛОВЫЕ ПОЛЯ**

**Задача 3.2.1 Товары в магазине**

Создайте таблицу **products** для хранения информации о товарах в магазине. Выберите оптимальные поля для хранения данных в соответствии с условиями:

1. **id** типа INT – только положительные числа;
2. **name** – символьный тип до 100 символов (включительно);
3. **count** – количество товара на складе (до 200 штук), целое положительное;
4. **price** – цена в рублях без копеек (не более 1 млн рублей). Только положительные значения.

Заполните таблицу тремя товарами:

* Холодильник, 10 штук, 50 000 рублей.
* Стиральная машина, 0 штук, 23 570 рублей.
* Утюг, 3 штуки, 7300 рублей

**Задача 3.2.2 Заказы со скидкой**

Создайте таблицу **orders** для хранения заказов в магазине. Выберите оптимальные поля для хранения данных в соответствии с условиями:

1. **id** – тип INT. Только положительные числа.
2. **product\_id** – для хранения номера товара. Только положительные числа от 0 до 4294967295.
3. **sale** – скидка. Целое положительное число от 0 до 100.
4. **amount** – сумма заказа. Денежный тип. Максимальная сумма заказа 999999.99 рублей.

Добавьте три записи так, чтобы получалась таблица ниже:

Таблица orders

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **product\_id** | **sale** | **amount** |
| 1 | 245 | 0 | 230.50 |
| 2 | 17 | 15 | 999999.99 |
| 3 | 145677 | 21 | 1ri |

**Задача 3.2.3 Фильмы**

Создайте таблицу **films** с информацией о фильмах. Выберете оптимальные поля для хранения данных в соответствии с условиями:

1. **id** типа INT, только положительные числа.
2. **name** – символьное поле длиной 100.
3. **rating** – рейтинг, вещественное число. Принимает положительные значения от 0 до 10.
4. **country** – страна фильма. Символьное поле, содержащее ровно 2 символа.

Добавьте в неё 3 записи так, чтобы получалась таблица ниже:

Таблица films

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **name** | **rating** | **country** |
| 1 | Большая буря | 3.45 | RU |
| 2 | Игра | 7.5714 | US |
| 3 | Война | 10.0 | RU |

**Задача 3.2.4 Файлы**

Создайте таблицу **files** для хранения информации о файлах. Выберете оптимальные поля исходя из условий ниже:

1. **id** — типа INT, только положительные числа.
2. **filename** — текстовое поле длиной 255 символов для хранения имени файла.
3. **size** — целочисленное поле для хранения размера файла в байтах. Только положительные числа. Могут храниться данные до 100 Гб.
4. **filetype** — поле для хранения типа файла, строка до 3 символов.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица files

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **filename** | **size** | **filetype** |
| 1 | big\_archive.zip | 81604378624 | zip |
| 2 | movie\_37.mp4 | 7838315315 | mp4 |
| 3 | music007.mp3 | 5242880 | mp3 |

**3.4 СТРОКОВЫЕ ПОЛЯ**

**Задача 3.4.1 Биография**

v1

**Задача 3.4.2 Книги**

Создайте таблицу **books** для хранения данных о книгах. В таблице должны быть следующие поля:

1. **id** – идентификатор, целое положительное.
2. **name** – название, строка длиной не более 100 символов.
3. **description** – описание книги длиной не более 1000 символов.
4. **isbn** – международный стандартный книжный номер. 10 или 13 знаков.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица books

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **name** | **description** | **isbn** |
| 1 | MySQL 5 | Хорошая книга. | 5941579284 |
| 2 | Изучаем SQL | Полезная книга. | 5932860510 |
| 3 | Изучаем Python. 4-е издание | Подробная книга о Python. | 9785932861592 |

**Задача 3.4.3 Автомобили Nissan**

Получите из таблицы **cars** информацию обо всех автомобилях марки Nissan выпущенных после 1990 года и мощностью более 120 лошадиных сил.

Таблица cars

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **mark** | **model** | **year** | **country** | **power** |
| 1 | Toyota | Camry | 2012 | JP | 139 |
| 2 | Mazda | Demio | 2004 | JP | 113 |
| 3 | ВАЗ | 2110 | 2010 | RU | 79 |
| 4 | Nissan | Almera | 2016 | JP | 130 |
| 5 | Nissan | Juke | 2016 | JP | 120 |
| 6 | Reno | Logan | 2009 | FR | 87 |
| 7 | Lada | Priora | 2017 | RU | 78 |
| 8 | Reno | Duster | 2011 | FR | 143 |
| 9 | Hyundai | Solaris | 2010 | KR | 98 |
| 10 | Nissan | Patrol | 2014 | JP | 200 |
| 11 | Reno | Logan | 2011 | FR | 90 |
| 12 | Nissan | Sunny | 1990 | JP | 67 |
| 13 | Reno | Dust | 2012 | FR | 130 |
| 14 | Nissan | Sunny | 1989 | JP | 140 |

**Задача 3.4.4 Квартиры**

Создайте таблицу **apartments** для хранения информации о квартирах на сайте жилищного комплекса. В таблице должны быть следующие поля:

1. **id** — идентификатор, целое положительное.
2. **image** — поле для хранения пути до изображения, строка не более 100 символов.
3. **price** — цена не более 99 999 999. Целое положительное число.
4. **square** — площадь квартиры. Не более 150 квадратных метров. Целое положительное число.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица apartments

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **image** | **price** | **square** |
| 1 | /apartments/1/cover.jpg | 5250000 | 90 |
| 2 | /apartments/2/cover-3.jpg | 7500000 | 103 |
| 3 |  | 2300000 | 56 |

**Задача 3.4.5 Японки и Пежо**

Получите из таблицы **cars** информацию обо всех японских автомобилях, а также об автомобилях марки Peugeot.

Выведите только марку, модель и год выпуска. Данные отсортируйте по году в обратном порядке.

Таблица cars

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **mark** | **model** | **year** | **country** | **power** |
| 1 | Toyota | Camry | 2012 | JP | 139 |
| 2 | Mazda | Demio | 2004 | JP | 113 |
| 3 | ВАЗ | 2110 | 2010 | RU | 79 |
| 4 | Nissan | Almera | 2016 | JP | 130 |
| 5 | Nissan | Juke | 2016 | JP | 120 |
| 6 | Reno | Logan | 2009 | FR | 87 |
| 7 | Peugeot | 3008 | 2016 | FR | 150 |
| 8 | Reno | Duster | 2011 | FR | 143 |
| 9 | Hyundai | Solaris | 2010 | KR | 98 |
| 10 | Nissan | Patrol | 2014 | JP | 200 |
| 11 | Reno | Logan | 2011 | FR | 90 |
| 12 | Nissan | Sunny | 1990 | JP | 67 |
| 13 | Peugeot | 307 | 2003 | FR | 110 |

**Задача 3.4.6 Путь до фотографии**

На сайте действует следующая система: сперва сотрудник заносит в таблицу URL-адрес фотографии в интернете, а после скрипт на языке Python скачивает файл из сети на жесткий диск и добавляет в эту же таблицу путь до него.

Создайте таблицу **files**, которая поможет сотруднику и скрипту выполнить свои задачи. Таблица должна содержать следующие поля:

1. **id** — целое положительное
2. **file\_url** — url файла, строка длиной до 500 символов.
3. **file\_path** — путь до файла на жестком диске, строка до 200 символов.
4. **size** — размер скачанного файла в байтах, целое положительное. Размер файла может достигать 100 Гб.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица files

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **file\_url** | **file\_path** | **size** |
| 1 | http://archives.com/archives/big\_archive.zip | files/2018/02/archive.zip | 81604378624 |
| 2 | http://movies.com/movies/movie.mp4 |  | 0 |
| 3 | http://movies.com/best-songs/song.mp3 | files/2018/03/song.mp3 | 5242880 |

**3.5 ДАТА И ВРЕМЯ**

**Задача 3.5.1 Регистрация и вход**

Создайте таблицу **users** для хранения информации о пользователях сайта.  
В таблице должны быть следующие поля:

1. **id** – идентификатор, целое положительное;
2. **email** – адрес электронной почты, строка не более 100 символов;
3. **date\_joined** – дата регистрации (достаточно хранить дату, без времени)
4. **last\_activity** – дата и время последней активности (с точностью до секунд).

Добавьте 3 записи, чтобы получалась таблица ниже:

Таблица users

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **email** | **date\_joined** | **last\_activity** |
| 1 | user1@domain.com | 2014-12-12 | 2016-04-08 12:34:54 |
| 2 | user2@domain.com | 2014-12-12 | 2017-02-13 11:46:53 |
| 3 | user3@domain.com | 2014-12-13 | 2017-04-04 05:12:07 |

**Задача 3.5.2 Календарь посетителей**

Создайте таблицу **calendar** для хранения календаря посетителей.  
В таблице должны быть следующие поля:

1. **id** – идентификатор записи в календаре, целое положительное;
2. **user\_id** – идентификатор пользователя, целое (INT) положительное;
3. **doctor\_id** – идентификатор доктора, целое (INT) положительное;
4. **visit\_date** – дата и время визита (точность до секунд).

Добавьте 5 записей, чтобы получалась таблица ниже:

Таблица calendar

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **user\_id** | **doctor\_id** | **visit\_date** |
| 1 | 1914 | 1 | 2017-04-08 12:00:00 |
| 2 | 12 | 1 | 2017-04-08 12:30:00 |
| 3 | 4641 | 2 | 2017-04-09 09:00:00 |
| 4 | 784 | 1 | 2017-04-08 13:00:00 |
| 5 | 15 | 2 | 2017-04-09 10:00:00 |

**Задача 3.5.3 Температура**

Создайте таблицу **temperature** для хранения записей о температуре. Добавьте в неё следующие поля:

1. **id** – идентификатор, целые положительные числа;
2. **city\_id** – идентификатор города, целые положительные числа;
3. **temperature** – температура, целое число в диапазоне от -100 до 100 градусов.
4. **wind\_speed** – целое положительное число для хранения скорости ветра. Историческая максимальная скорость ветра была 113 м/с.
5. **mdate** – дата и время измерения (с точностью до секунд).

Добавьте 5 записей, чтобы получалась таблица ниже:

Таблица temperature

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **city\_id** | **temperature** | **wind\_speed** | **mdate** |
| 1 | 456 | 17 | 7 | 2017-02-08 12:00:00 |
| 2 | 456 | 19 | 6 | 2017-02-08 12:10:00 |
| 3 | 456 | 20 | 6 | 2017-02-08 12:20:00 |
| 4 | 471 | -7 | 12 | 2017-02-08 12:20:01 |
| 5 | 44 | -43 | 17 | 2017-02-08 12:23:12 |

**Задача 3.5.4 Время прихода на работу**

Создайте таблицу **arrival** для хранения времени прихода сотрудников на работу. Добавьте в неё следующие поля:

1. **id** – идентификатор, целое положительное;
2. **user\_id** – идентификатор пользователя, целое положительное;
3. **a\_date** – дата прихода;
4. **a\_time** – время прихода.

Добавьте 4 записи, чтобы получалась таблица ниже:

Таблица arrival

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **user\_id** | **a\_date** | **a\_time** |
| ye | 10 | 2017-03-09 | 08:45:41 |
| 2 | 12 | 2017-03-09 | 08:46:12 |
| 3 | 7 | 2017-03-09 | 08:53:01 |
| 4 | 31 | 2017-03-09 | 09:01:17 |

**Задача 3.5.5 Автомобили дилера**

Создайте таблицу **cars** для хранения списка автомобилей дилерского центра. Таблица должна содержать следующие поля:

1. **id** – идентификатор, целое положительное;
2. **mark** – марка автомобиля, строка до 20 символов;
3. **model** – модель автомобиля, строка до 20 символов;
4. **year** – год выпуска, используйте тип YEAR;
5. **mileage** – пробег, целое число от 0 до 100 000.

Добавьте 3 автомобиля, чтобы получалась таблица ниже:

Таблица cars

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **mark** | **model** | **year** | **mileage** |
| 1 | Toyota | Camry | 2015 | 32000 |
| 2 | Mazda | CX-5 | 2016 | 17000 |
| 3 | Nissan | Patrol | 2016 | 60000 |

**Задача 3.5.6 Автомобили 2016 года**

Выберите из таблицы **cars** все автомобили с пробегом меньше 50 000, которые выпустили в 2016 году. Результаты отсортируйте по пробегу.

Таблица cars

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **mark** | **model** | **year** | **mileage** |
| 1 | Toyota | Camry | 2015 | 32000 |
| 2 | Mazda | CX-5 | 2016 | 17000 |
| 3 | Nissan | Patrol | 2016 | 60000 |
| 4 | Toyota | Prius | 2015 | 10000 |
| 5 | Mazda | Demio | 2003 | 98000 |
| 6 | Nissan | Murano | 2016 | 5000 |

**3.6 NULL**

**Задача 3.6.1 Пустой email**

Создайте таблицу **users** с двумя полями:

1. **id** — целое положительное, не может принимать NULL.
2. **email** — строка для хранения электронных адресов пользователей длиной до 100 символов, не может принимать NULL.

Добавьте 4 записи так, чтобы получалась таблица ниже:

Таблица users

|  |  |
| --- | --- |
| **id** | **email** |
| n | user1@domain.com |
| 2 | user2@domain.com |
| 3 | user3@domain.com |
| 4 | user4@domain.com |

**Задача 3.6.2 Таблица для товаров**

Создайте таблицу **products** со следующими полями:

1. **id** — номер товара, целое положительное, не может принимать NULL.
2. **name** — название товара до 120 символов, не может принимать NULL.
3. **category\_id** — категория товара, целое положительное, может принимать NULL.
4. **price** — цена, денежный тип из 10 знаков, из них два после запятой. Не может принимать NULL.

Добавьте 4 записи так, чтобы получалась таблица ниже:

Таблица products

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **name** | **category\_id** | **price** |
| 1 | Подгузники (12 шт) | 3 | 700.00 |
| 2 | Подгузники (24 шт) | 3 | 1250.00 |
| 3 | Спиннер | NULL | 250.40 |
| 4 | Пюре слива | 4 | 47.50 |

**Задача 3.6.3 Товары без категорий**

Выберите из таблицы **products** название, количество и цены всех товаров, у которых нет категории. Данные отсортируйте по цене.

Таблица products

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **category\_id** | **name** | **count** | **price** |
| 1 | 7 | Стиральная машина | 5 | 10000 |
| 2 | 12 | Холодильник | 0 | 10000 |
| 3 | 12 | Микроволновка | 3 | 4000 |
| 4 | 8 | Пылесос | 2 | 4500 |
| 5 | NULL | Вентилятор | 0 | 700 |
| 6 | 9 | Телевизор | 7 | 31740 |
| 7 | 12 | Тостер | 2 | 2500 |
| 8 | NULL | Принтер | 4 | 3000 |
| 9 | NULL | Активные колонки | 1 | 2900 |

**3.8 BOOL, ENUM, SET**

**Задача 3.8.1 Статьи**

Создайте таблицу **articles** для хранения данных о статьях. В таблице должны быть следующие поля:

* **id** — идентификатор, целое положительное, NULL запрещен
* **name** — название статьи, строка до 80 символов.
* **text** — текст статьи.
* **state** — статус статьи. Поле из 3 вариантов: draft (черновик), correction (корректура), public (опубликована).

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица articles

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **name** | **text** | **state** |
| 1 | Новое в Python 3.6 |  | draft |
| 2 | Оптимизация SQL запросов | При больших объемах данных ... | correction |
| 3 | Транзакции в MySQL | По долгу службы мне приходится ... | public |

**Задача 3.8.2 Номера в отеле**

Создайте таблицу **rooms** для хранения номеров в отеле:

1. **id** — идентификатор, целое положительное. NULL запрещен.
2. **number** — номер комнаты, целое положительное. Всего в отеле 107 комнат. NULL запрещен.
3. **beds** — количество спальных мест. Выбор из **1+1**, **2+1**, **2+2**. Можно выбрать только один вариант. NULL запрещен.
4. **additional** — дополнительные удобства в номере. Можно выбрать несколько вариантов из списка: **conditioner**, **bar**, **fridge** и **wifi**.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица rooms

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **number** | **beds** | **additional** |
| 1 | 10 | 1+1 | conditioner,bar,wifi |
| 2 | 12 | 2+1 |  |
| 3 | 24 | 2+2 | fridge,bar,wifi |

**Задача 3.8.3 Персональные данные**

Получите id, имена, фамилии и дни рождения всех пользователей у которых статус обработки персональных данных (**pers\_info**) не установлен или установлен в False.

Данные отсортируйте по дате рождения.

Таблица users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **birthday** | **pers\_info** |
| 1 | Дмитрий | Петров | 2000-03-14 | TRUE |
| 2 | Ольга | Антонова | 1999-12-01 | FALSE |
| 3 | Сергей | Васильев | 2002-02-20 | NULL |
| 4 | Констанин | Степаниденко | 2004-03-07 | TRUE |
| 5 | Алена | Шикова | 1999-08-17 | TRUE |
| 6 | Василина | Парамонова | 2000-10-10 | FALSE |
| 7 | Александр | Пузаков | 2002-02-22 | NULL |
| 8 | Алина | Антонова | 2002-01-01 | FALSE |

**Задача 3.8.4 Товары из России и Украины**

Выберите из таблицы **products** название, цену и страну всех товаров из России и Украины. Выбирайте только товары, которые есть в наличии.  
Данные отсортируйте сперва по стране, а после по цене.

Поле **country** относится к типу **ENUM('RU', 'UA', 'BY', 'KZ') NOT NULL**.

Таблица products

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **category\_id** | **name** | **count** | **price** | **country** |
| 1 | 7 | Стиральная машина | 5 | 10000 | RU |
| 2 | 12 | Холодильник | 0 | 10000 | BY |
| 3 | 12 | Микроволновка | 3 | 4000 | UA |
| 4 | 8 | Пылесос | 2 | 4500 | KZ |
| 5 | NULL | Вентилятор | 0 | 700 | BY |
| 6 | 9 | Телевизор | 7 | 31740 | RU |
| 7 | 12 | Тостер | 0 | 2500 | RU |
| 8 | NULL | Принтер | 4 | 3000 | UA |
| 9 | NULL | Активные колонки | 1 | 2900 | UA |

**Задача 3.8.5 Товары из России и Белоруссии**

Выберите из таблицы **products** название, цену и страны всех товаров из России и Белоруссии (в поле **country** обязательно должна присутствовать или Россия, или Белоруссия). Выбирайте только товары, у которых задана категория. Данные отсортируйте по цене в обратном порядке.

Поле **country** относится к типу **SET('RU', 'UA', 'BY', 'KZ') NOT NULL**.

Таблица products

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **category\_id** | **name** | **count** | **price** | **country** |
| 1 | 7 | Стиральная машина | 5 | 10000 | RU,UA |
| 2 | 12 | Холодильник | 0 | 11000 | BY |
| 3 | 12 | Микроволновка | 3 | 4000 | UA,KZ |
| 4 | 8 | Пылесос | 2 | 4500 | BY,KZ |
| 5 | NULL | Вентилятор | 0 | 700 | RU,BY |
| 6 | 9 | Телевизор | 7 | 31740 | RU |
| 7 | 12 | Тостер | 0 | 2500 | RU |
| 8 | NULL | Принтер | 4 | 3000 | UA,BY |
| 9 | NULL | Активные колонки | 1 | 2900 | UA |

**Задача 3.8.6 Менеджеры**

В таблице **users** содержится информация о сотрудниках компании.  
Выберите из неё имена, фамилии, дни рождения и роли всех менеджеров мужского пола, которые в данный момент работают (положительное поле active).

Данные отсортируйте сперва по фамилии, а затем по имени.

Поле **roles** относится к типу **SET('programmer', 'designer', 'manager', 'marketer') NOT NULL**.

Таблица users

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **birthday** | **active** | **sex** | **roles** |
| 1 | Дмитрий | Федьков | 2000-01-02 | True | m | manager |
| 2 | Светлана | Иванова | 1999-03-02 | True | w | designer |
| 3 | Никита | Трутнев | 2000-08-08 | True | m | programmer |
| 4 | Денис | Кац | 2000-08-09 | True | m | programmer,manager |
| 5 | Алена | Сорокина | 2001-01-02 | True | w | designer |
| 6 | Василий | Федьков | 2001-01-01 | False | m | manager |
| 7 | Ярослав | Сорокин | 2000-08-01 | True | m | manager,designer |
| 8 | Денис | Белый | 2000-12-09 | True | m | designer |
| 9 | Анатолий | Дейчман | 1998-08-08 | True | m | programmer |
| 10 | Юлия | Фёдорова | 2002-08-08 | False | w | marketer |
| 11 | Анастасия | Литвинова | 1991-06-01 | True | w | manager |

**Задача 3.8.7 Таблица заказов**

Создайте таблицу **orders** для хранения заказов в интернет-магазине:

1. **id** — идентификатор, целое положительное.
2. **user\_id** — идентификатор пользователя, который оформил заказ. Целое положительное, NULL запрещен.
3. **amount** — стоимость заказа. DECIMAL из 10 цифр, 2 цифры после десятичной точки.
4. **created** — дата и время создания заказа. NULL запрещен.
5. **state** – статус заказа.

Выбор из **new**, **cancelled**, **in\_progress**, **delivered**, **completed**. Можно выбрать только один вариант.

1. **options** — дополнительные опции заказа. Можно выбрать несколько вариантов из списка: **pack**, **fitting**, **call** и **intercom**.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица orders

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **user\_id** | **amount** | **created** | **state** | **options** |
| 456 | 763 | 14299.00 | 2018-02-01 17:45:59 | completed | pack,call |
| 457 | 1987 | 249.50 | 2018-02-01 18:23:04 | delivered | pack,intercom |
| 459 | 78 | 2300.12 | 2018-02-01 22:12:09 | in\_progress |  |

**Задача 3.8.8 Непроданные японские автомобили**

Получите из таблицы **cars** информацию обо всех японских автомобилях марки Nissan с 2010 по 2016 годы включительно Выбирайте только непроданные автомобили дилеров. Отметка об успешной продаже находится в поле **sold**, а информация о дилерах в поле **dealer\_id**.

Выведите только марку, модель, год и мощность автомобиля. Данные отсортируйте по мощности.

Таблица cars

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **dealer\_id** | **mark** | **model** | **year** | **country** | **power** | **sold** |
| 1 | 1 | Toyota | Camry | 2012 | JP | 139 | FALSE |
| 2 | 23 | Mazda | Demio | 2004 | JP | 113 | FALSE |
| 3 | 7 | ВАЗ | 2110 | 2010 | RU | 79 | TRUE |
| 4 | NULL | Nissan | Almera | 2016 | JP | 130 | FALSE |
| 5 | 1 | Nissan | Patrol | 2014 | JP | 200 | FALSE |
| 6 | 23 | Reno | Logan | 2009 | FR | 87 | TRUE |
| 7 | 7 | Lada | Priora | 2017 | RU | 78 | FALSE |
| 8 | 67 | Reno | Duster | 2011 | FR | 143 | FALSE |
| 9 | 13 | Hyundai | Solaris | 2010 | KR | 98 | FALSE |
| 10 | 5 | Nissan | Juke | 2016 | JP | 120 | FALSE |
| 11 | 2 | Reno | Logan | 2011 | FR | 90 | FALSE |
| 12 | NULL | Nissan | Sunny | 1990 | JP | 67 | FALSE |
| 13 | 1 | Nissan | Juke | 2012 | JP | 125 | TRUE |
| 14 | 5 | Nissan | Juke | 2009 | JP | 120 | FALSE |
| 15 | 24 | Nissan | Almera | 2017 | JP | 113 | FALSE |

**3.10 ЗНАЧЕНИЯ ПО УМОЛЧАНИЮ**

**Задача 3.10.1 Статус по умолчанию**

Создайте таблицу **orders** для хранения списка заказов:

1. **id** — идентификатор, целое положительное.
2. **user\_id** — идентификатор пользователя, который оформил заказ. Целое положительное, NULL запрещен.
3. **amount** — стоимость заказа. Целое положительное число не более 1 млн. NULL запрещен, по умолчанию 0.
4. **created** — дата и время создания заказа. NULL запрещен.
5. **state** — статус заказа.

Выбор из **new**, **cancelled**, **in\_progress**, **delivered**, **completed**. Можно выбрать только один вариант. NULL запрещен. По умолчанию должен стоять **new**.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица orders

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **user\_id** | **amount** | **created** | **state** |
| 1 | 56 | 5400 | 2018-02-01 17:46:59 | new |
| 2 | 90 | 249 | 2018-02-01 19:13:04 | new |
| 3 | 78 | 2200 | 2018-02-01 22:43:09 | new |

**Задача 3.10.2 Пользователи**

Создайте таблицу **users** для хранения списка пользователей сайта:

1. **id** — идентификатор, целое положительное.
2. **first\_name** — имя, строка до 20 символов. NULL запрещен.
3. **last\_name** — фамилия, строка до 20 символов. NULL запрещен.
4. **patronymic** — отчество, строка до 20 символов. NULL запрещен, по умолчанию пустая строка.
5. **is\_active** — отметка об активности пользователя. Логическое поле, по умолчанию TRUE.
6. **is\_superuser** — отметка администратора. Логическое поле, по умолчанию FALSE.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица users

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **first\_name** | **last\_name** | **patronymic** | **is\_active** | **is\_superuser** |
| 1 | Дмитрий | Иванов |  | TRUE | FALSE |
| 2 | Анатолий | Белый | Сергеевич | TRUE | TRUE |
| 3 | Андрей | Крючков |  | FALSE | FALSE |

**Задача 3.10.3 Товары**

Создайте таблицу **products** для хранения товаров в интернет магазине:

1. **id** — идентификатор, целое положительное.
2. **category\_id** — категория, целое положительное. Может принимать NULL. По умолчанию NULL.
3. **name** — название, строка до 100 символов. NULL запрещен.
4. **count** — количество, целое положительное до 255. NULL запрещен, по умолчанию 0.
5. **price** — цена типа DECIMAL с 10 знаками, 2 из которых выделены для копеек. NULL запрещен, по умолчанию 0.00.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица products

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **category\_id** | **name** | **count** | **price** |
| 1 | 1 | Кружка | 5 | 45.90 |
| 2 | 17 | Фломастеры | 0 | 78.00 |
| 3 | NULL | Сникерс | 12 | 50.80 |

**Задача 3.10.4 Логи**

Создайте таблицу **logs** для хранения логов посещения сайта:

1. **date** — время посещения в формате DATETIME, NULL запрещен, по умолчанию должно проставляться время создания записи.
2. **browser** — название браузера, строка до 500 символов. NULL запрещен, по умолчанию пустая строка.
3. **is\_bot** — отметка о том является ли посетитель ботом или нет. Логический тип, по умолчанию FALSE. NULL запрещен.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица logs

|  |  |  |
| --- | --- | --- |
| **date** | **browser** | **is\_bot** |
| 2018-03-19 19:50:01 | Chrome 64.0.1.417 | FALSE |
| 2018-03-19 19:55:11 | Firefox 55 (yandex bot) | True |
| 2018-03-19 19:56:12 | Chrome 63.0.0.471 | FALSE |

**Задача 3.10.5 Отзывы**

Создайте таблицу **reviews** с отзывами о компании Shultais Education:

1. **id** — идентификатор, целое положительное.
2. **user\_id** — идентификатор пользователя, который оставил отзыв. Целое положительное, NULL запрещен.
3. **date** — дата и время оставления отзыва, NULL запрещен, по умолчанию должно проставляться время создания записи.
4. **course** — курс, для которого оставлен отзыв. Выбор из **python**, **sql**, **all**. Можно выбрать только один вариант. NULL запрещен. Значение по умолчанию — **all**.
5. **text** — текст отзыва, поле типа TEXT. NULL запрещен.
6. **public** — отметка о том является ли отзыв публичным. Логический тип, по умолчанию FALSE. NULL запрещен.

Добавьте 3 записи так, чтобы получалась таблица ниже:

Таблица reviews

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **id** | **user\_id** | **date** | **course** | **text** | **public** |
| 1 | 817 | 2018-01-11 19:50:01 | python | Это прекрасная возможность получить новые очки в программировании. Доступное объяснение позволяет без проблем изучить желаемый материал | TRUE |
| 2 | 1289 | 2018-02-16 08:55:11 | python | Проект на мой взгляд отличный: 1. Короткие видеоролики без воды 2. Интересные задачи и практика 3. Очень быстрая обратная связь | TRUE |
| 3 | 2914 | 2018-03-19 12:56:12 | all | Хорошая затея. Но проект ещё нуждается в развитии. | TRUE |