

Задание

Требование к оформлению результата:

все результаты должны быть оформлены **в едином файле** без указания внутри него ссылок на другие ресурсы

1. Базы данных - тест

Для приведенных ниже вопросов выберите верные утверждения. Верным может быть одно или несколько утверждений.

1	Содержит ли какую-то информацию таблица, в которой нет полей?	<ul style="list-style-type: none">1. Содержит информацию о структуре БД2. Не содержит никакой информации3. Таблица без полей существовать не может4. Содержит информацию о будущих записях
2	В записи файла реляционной БД может содержаться:	<ul style="list-style-type: none">1. Исключительно однородная информация (данные только одного типа)2. Только текстовая информация3. Только логические величины4. Неоднородная информация (данные разных типов)5. Исключительно числовая информация

3	Чем первичный ключ отличается от внешнего ключа?	<ul style="list-style-type: none"> 1. Первичный ключ всегда состоит из множества столбцов, а внешний ключ состоит из одного столбца 2. Значения первичного ключа всегда должны быть уникальными и не могут быть null, значения внешнего ключа могут повторяться 3. Внешний ключ является идентификатором строки, а первичный ключ используется для связи между таблиц 4. Первичный ключ является идентификатором для строки, а внешний ключ используется для связывания таблиц
4	В какой нормальной форме говорится о том, что все атрибуты зависят от первичного ключа, а не от его части?	<ul style="list-style-type: none"> 1. 1НФ 2. 2НФ 3. 3НФ 4. 4НФ
5	В каком порядке в СУБД выполняются операторы SELECT, FROM, GROUP BY?	<ul style="list-style-type: none"> 1. Сначала SELECT, потом FROM и только потом GROUP BY 2. Сначала GROUP BY, потом SELECT и только потом FROM 3. Сначала FROM, потом SELECT и только потом GROUP BY 4. Сначала FROM, потом GROUP BY и только потом SELECT

6	Чем отличается оператор WHERE от HAVING	<p>1. Оператор WHERE применяется для фильтрации групп, а HAVING - для фильтрации отдельных строк</p> <p>2. Оператор HAVING применяется для фильтрации групп, а WHERE - для фильтрации отдельных строк</p> <p>3. HAVING работает только с агрегатными функциями, а WHERE может работать с любыми типами выражений</p> <p>4. WHERE может использоваться для фильтрации по любому полю или выражению, а HAVING - только для фильтрации по выражению в списке выбора или агрегатной функции</p> <p>5. HAVING всегда используется после GROUP BY, а WHERE может использоваться до или после GROUP BY</p>
7	Какой результат покажет выполнение операторов SELECT COUNT (*)?	<p>1. Число строк таблицы, указанной во FROM, не включая значение NULL</p> <p>2. Число строк таблицы, указанной во FROM, где ячейка содержит символ</p> <p>3. Число строк таблицы, указанной во FROM, включая значение NULL</p> <p>4. Сумма строк таблицы, указанной во FROM, где ячейка содержит символ</p>
8	В таблице «Animals» базы данных зоопарка содержится информация обо всех обитающих там животных, в том числе о лисах: red fox, grey fox, little fox. Напишите запрос, возвращающий информацию о возрасте лис	<p>1. SELECT age FROM Animals WHERE Animal LIKE "%fox"</p> <p>2. SELECT age FROM %Fox.Animals</p> <p>3. SELECT age FROM Animals WHERE Animal = fox</p> <p>4. SELECT %fox age FROM Animals</p>

9	<p>Чем отличается DELETE от TRUNCATE?</p>	<p>1. DELETE и TRUNCATE - это одно и то же 2. DELETE используется для удаления одной или нескольких строк из таблицы, а TRUNCATE используется для удаления всех строк из таблицы 3. DELETE может использовать условие WHERE, а TRUNCATE всегда удаляет все записи из таблицы 4. DELETE удаляет данные из таблицы, а TRUNCATE удаляет саму таблицу</p>					
10	<p>Дана таблица:</p> <table><tr><td>COLOR</td></tr><tr><td>BLUE</td></tr><tr><td>RED</td></tr><tr><td>null</td></tr><tr><td>RED</td></tr></table> <p>Каким будет результат запроса? SELECT COUNT (DISTINCT color) FROM Table</p>	COLOR	BLUE	RED	null	RED	<p>1. BLUE,RED,NULL 2. 3 3. 1,2,4 4. 2</p>
COLOR							
BLUE							
RED							
null							
RED							

2. Базы данных - ER

В базе данных есть таблица заказов - orders. В ней есть поля: id (идентификатор заказа), name (название заказа), town (адрес доставки заказа), price (цена заказа), customer_id (идентификатор покупателя). Также есть таблицы: towns (справочник адресов), items (товары), customers (покупатели). Известно, что между orders и items предполагается связь многие-ко-многим.

Что нужно сделать:

Спроектируйте ER-диаграмму с учетом этих вводных. Состав полей таблиц укажите на свое усмотрение с учетом условия, но для таблицы orders учтите те поля, что указаны.

3. Интеграции

Представим, что Вы работаете аналитиком и проектируете работу приложения интернет-магазина. Вам нужно заложить и спроектировать следующий сценарий: отображение витрины товаров (список товаров с кратким описанием), переход с витрины на экран с детальным описанием конкретного товара, добавление товара в корзину.

Что нужно сделать:

1. Спроектируйте REST API, которые нужны для реализации описанного сценария. Решение должно включать описание запросов и описание или пример ответа для каждого из запросов в формате JSON. Способ описания - на ваше усмотрение. По составу полей товара можете ориентироваться на любой известный интернет-магазин.
2. Постройте Sequence UML диаграмму для этого сценария.

4. Алгоритмическое мышление

Возьмем в качестве примера банковское мобильное приложение.

Исходные условия: у вас в руке смартфон, на котором установлено банковское приложение (телефон выключен).

Что нужно сделать:

Используя любую нотацию, опишите в виде диаграммы процесс, в рамках которого Вы, используя банковское приложение, пополните баланс своего телефона на 100Р. В качестве примера можете использовать любое мобильное банковское приложение, что у Вас есть.