**Тестовое задание для аналитиков**

Тестовое задание SQL (только один вариант ответа):

1. Какой оператор используется для добавления данных в таблицу?

a) INSERT INTO

b) UPDATE

c) DELETE FROM

d) SELECT

Ответ: INSERT INTO

1. Каким оператором можно выбрать данные из нескольких таблиц?

a) UNION ALL

b) JOIN

c) INSERT

d) SELECT

Ответ: UNION ALL

1. Каким образом можно удалить таблицу?

a) DELETE

b) UPDATE

c) TRUNCATE

d) DROP

Ответ: DROP

1. Что вернет следующий SQL запрос SELECT product\_id, COUNT(order\_id) FROM order\_details GROUP BY product\_id?  
   a) Количество заказов для каждого клиента в базе данных  
   b) Количество продуктов в каждом заказе в базе данных  
   c) Количество заказов для каждого продукта в базе данных

d) Суммарное количество заказов и продуктов

Ответ: с) Количество заказов для каждого продукта в базе данных

1. Какой оператор используется для изменения структуры существующей таблицы в SQL?  
   a) ALTER TABLE  
   b) UPDATE TABLE  
   c) MODIFY TABLE  
   d) CHANGE TABLE

Ответ: ALTER TABLE

1. Что такое реляционные базы данных?
2. База данных, в которой данные хранятся в виде двумерных таблиц, связанных между собой
3. Хранит данные в формате ключ-значение
4. Любая база данных – реляционная
5. Совокупность не связанных данных

Ответ: База данных, в которой данные хранятся в виде двумерных таблиц, связанных между собой

1. Как выглядит вопрос для вывода всех значений из таблицы Orders?
   1. select ALL from Orders  
      b) select \* from Orders  
      c) select % from Orders  
      d) select orders from Orders

Ответ: b) select \* from Orders

8. Как использовать функцию COUNT() в SQL?  
a) SELECT COUNT(\*) FROM table\_name  
b) SELECT COUNT(column\_name) FROM table\_name  
c) SELECT COUNT(column\_name) AS count\_name FROM table\_name  
d) Все вышеперечисленные варианты

Ответ: d) Все вышеперечисленные варианты

9. Какое ключевое слово используется в SQL для фильтрации данных в таблице?

a) FILTER;

b) SELECT;

c) WHERE;

d) HAVING.

Ответ: c) WHERE;

10. Как сортировать данные в таблице в порядке возрастания?

a) ORDER BY ASC;

b) ORDER BY DESC;

c) ORDER BY;

d) GROUP BY.

Ответ: a) ORDER BY ASC;

**Кейс:**

Крупная компания планирует создать Систему для бронирования отелей. Система позволяет бронировать номера в отелях по всем странам. В системе можно бронировать номера в отелях, с возможностью предоплаты и без.

**Список функциональных требований:**  
1. Регистрация пользователей и создание профилей как для пользователей, так и для владельцев отелей;

2. Поиск отелей по странам и городам;

3. Просмотр информации об отеле, номерах и ценах для бронирования;

4. Бронирование номеров с возможностью предоплаты или без;

5. Роли в системе: пользователь, админ отеля;

6. Возможность отказаться от предоплаты и оплатить номер при заселении;

7. Оплата комиссии системе за бронирование;

8. Получение электронного подтверждения бронирования;

9. Интеграция с системами платежных сервисов;

10. Отчетность для администраторов, описывающая количество забронированных номеров, частоту бронирований и т.д.

**Задание:**

1. Описать систему с помощью UML-диаграмм как: Диаграмма классов, диаграмма последовательностей, ER-диаграмма.
2. Описать API (в REST) получения всех отелей, номеров в отелях, и бронирования. Должно быть включены: примеры запроса/ответа сервера, входные/выходные данные и тд.
3. Описать словами, как бы вы построили работу команды с точки зрения подхода к разработке. Вопрос не обязателен, но желателен.

Для описания системы бронирования отелей можно использовать следующие UML-диаграммы:

1. Диаграмма классов: позволяет описать структуру системы и ее компоненты, а также связи между ними. На диаграмме классов можно отобразить классы, свойства и методы, а также их взаимодействие.



1. Диаграмма последовательностей: позволяет описать последовательность действий, которые выполняются в системе при выполнении определенной операции. На диаграмме последовательностей можно отобразить объекты, которые участвуют в операции, и сообщения, которые они обмениваются.



менеджер

отчет

счет

бухгалтер



номер

номер

ресепшен

ресепшен

клиент

клиент

ER-диаграмма: позволяет описать структуру базы данных, которая используется в системе. На ER-диаграмме можно отобразить таблицы, связи между ними и атрибуты таблиц.

Пример API для системы бронирования отелей

GET /hotels

* Возвращает список всех отелей в системе.
* Параметры запроса: нет.
* Ответ: массив объектов отелей, каждый из которых содержит информацию об отеле (id, name, city, country).

GET /hotels/{hotel\_id}/rooms

* Возвращает список всех номеров в отеле с указанным id.
* Параметры запроса: hotel\_id - id отеля.
* Ответ: массив объектов номеров, каждый из которых содержит информацию о номере (id, name, price).

POST /bookings

* Создает новое бронирование номера в отеле.
* Параметры запроса: user\_id - id пользователя, hotel\_id - id отеля, room\_id - id номера, date - дата бронирования, prepayment - флаг предоплаты.
* Ответ: объект бронирования, содержащий информацию о бронировании (id, user\_id, hotel\_id, room\_id, date, status).

PUT /bookings/{booking\_id}/cancel

* Отменяет бронирование с указанным id.
* Параметры запроса: booking\_id - id бронирования.
* Ответ: объект бронирования, содержащий информацию о бронировании (id, user\_id, hotel\_id, room\_id, date, status).

POST /payments

* Создает новый платеж за бронирование номера в отеле.
* Параметры запроса: user\_id - id пользователя, booking\_id - id бронирования, amount - сумма платежа.
* Ответ: объект платежа, содержащий информацию о платеже (id, user\_id, booking\_id, date, amount).

Построение работы команды с точки зрения подхода к разработке

Для разработки системы бронирования отелей можно использовать Agile-методологию, такую как Scrum. Команда должна состоять из разработчиков, тестировщиков, дизайнеров и менеджеров проекта.

1. Планирование: команда должна определить требования к системе и составить план разработки, включающий в себя список задач, сроки и ответственных за выполнение каждой задачи.
2. Создание прототипа: команда должна создать прототип системы, который позволит проверить работоспособность и функциональность системы.
3. Разработка: команда должна разработать систему, используя Agile-подход и методологию Scrum. Разработчики должны работать в спринтах, выполняя задачи из списка задач.
4. Тестирование: команда должна провести тестирование системы, чтобы убедиться в ее работоспособности и соответствии требованиям.
5. Внедрение: команда должна внедрить систему и обеспечить ее поддержку и сопровождение.