**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

1. Каким оператором можно выбрать данные из нескольких таблиц?

a) UNION ALL

b) JOIN

c) INSERT

d) SELECT

1. Каким образом можно удалить таблицу?

a) DELETE

b) UPDATE

c) TRUNCATE

d) 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

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

1. Как выглядит вопрос для вывода всех значений из таблицы Orders?

a) select ALL from Orders  
b) select \* from Orders  
c) select % from Orders  
d) select orders 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

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

a) FILTER;

b) SELECT;

c) WHERE;

d) HAVING.

10. Как сортировать данные в таблице в порядке возрастания?

a) ORDER BY ASC;

b) ORDER BY DESC;

c) ORDER BY;

d) GROUP BY.

**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.

Sequence diagram

Изображение выглядит как снимок экрана, текст, Параллельный, линия

Автоматически созданное описание

ER-диаграммы

Изображение выглядит как снимок экрана, дизайн

Автоматически созданное описание

Изображение выглядит как снимок экрана, текст, дизайн

Автоматически созданное описание

Описание API для получения всех отелей, номеров и бронирования (в формате REST):

1. Получить список всех отелей:

- Запрос: GET /hotels

- Ответ:

```json

[

{

"id": 1,

"название": "Отель 1",

"город": "Город 1",

"страна": "Страна 1"

},

{

"id": 2,

"название": "Отель 2",

"город": "Город 2",

"страна": "Страна 2"

},

...

]

```

2. Получить список всех номеров в отеле:

- Запрос: GET /hotels/{hotel\_id}/rooms

- Ответ:

```json

[

{

"id": 1,

"название": "Номер 1",

"цена": 100,

"доступность": true

},

{

"id": 2,

"название": "Номер 2",

"цена": 150,

"доступность": false

},

...

]

```

3. Создать новую бронь:

- Запрос: POST /bookings

- Тело запроса:

```json

{

"hotel\_id": 1,

"room\_id": 2,

"дата": "2022-08-01",

"предоплата": true,

"пользователь": {

"имя": "Имя пользователя",

"email": "user@example.com"

}

}

```

- Ответ:

```json

{

"id": 1,

"дата": "2022-08-01",

"предоплата": true,

"пользователь": {

"имя": "Имя пользователя",

"email": "user@example.com"

}

}

```

4. Получить список бронирований пользователя:

- Запрос: GET /bookings?user\_email=user@example.com

- Ответ:

```json

[

{

"id": 1,

"дата": "2022-08-01",

"предоплата": true,

"пользователь": {

"имя": "Имя пользователя",

"email": "user@example.com"

}

},

{

"id": 2,

"дата": "2022-08-05",

"предоплата": false,

"пользователь": {

"имя": "Имя пользователя",

"email": "user@example.com"

}

},

...

]

```

Ответ на вопрос № 3

Разработка может осуществляться по методологии итерации, где каждая итерация (спринт) имеет определенный набор задач, которые команда выполняет за фиксированное время (например, 2 недели). На ежедневной встрече команда обсуждает прогресс, препятствия, результаты и планы.

С точки зрения подхода к разработке, для этого проекта можно применить Agile-методологию разработки, такую как Scrumban, которая комбинирует преимущества Scrum и Kanban. Команда разработки может быть организована по следующим ролям:

- Product Owner: отвечает за управление требованиями, планирование и приоритезацию задач, а также представляет интересы заказчика.

- Scrum Master: обеспечивает процесс разработки, управляет командой и помогает ей стать самоорганизующейся и достигать поставленных целей.

- Разработчики: команда разработчиков, состоящая из аналитиков, разработчиков, тестировщиков и дизайнеров, отвечает за создание функциональных требований, разработку, тестирование и дизайн системы.

Важно обеспечить постоянное взаимодействие и обратную связь с заказчиком, чтобы правильно понимать и удовлетворять его потребности. Регулярные демонстрации продукта и ретроспективы помогут команде повысить качество и эффективность работы.