Задачи для собеседования в офисе:

Хранимая процедура

/\*

Задача.

Есть база данных в банке.

В базе данных есть таблица со счетами (назовем её для простоты T).

В таблице T есть среди прочих поля:

N - Номер счета

S - сумма на счете

Требуется:

Написать хранимую процедуру на языке TSQL, которая:

1. Принимает в качестве аргументов параметры:

@N1 - номер первого счета

@N2 - номер второго счета

@S - сумма денежных средств

2. Переводит сумму @S с первого на второй счет, при это проверяя, достаточно ли средств на первом счете

3. Использует транзакцию при переводе с первого на второй счет (показать, как работают транзакции).

PS

Синтаксис можно строго не соблюдать, важно понимание

\*/

Ответ:

USE Bank;

GO

CREATE PROCEDURE MoneyTransfer

@N1 int,

@N2 int,

@S decimal(18, 2)

AS

BEGIN

BEGIN TRANSACTION;

if (select amount from T

where id\_account = @N1) >= @S

BEGIN

UPDATE T

SET amount -= @S

WHERE id\_account = @N1

UPDATE T

SET amount += @S

WHERE id\_account = @N2

END

else

THROW 50000, 'no amount due', 0;

COMMIT;

END

Задача на вывод в XML.

Пусть есть таблица Purchase со следующим видом и содержанием. Что вернет SQL запрос?

|  |  |  |  |
| --- | --- | --- | --- |
| **Id** | **Code** | **Name** | **StatusId** |
| 1 | SBR003-202001 | Конкурс СМСП | 45 |
| 2 | SBR003-202002 | Запрос предложений | 2 |

SELECT Id, Code, Name, StatusId FROM Purchase FOR XML path('row'), ROOT('data')

Ответ:

<data>

<row>

<Id>1</Id>

<Code>SBR003-202001</Code>

<Name>Конкурс СМСП</Name>

<StatusId>45</StatusId>

</row>

<row>

<Id>2</Id>

<Code>SBR003-202002</Code>

<Name>Запрос предложений</Name>

<StatusId>2</StatusId>

</row>

</data>

Задача на валидность XML

Перед вами пример XML файла, но он не пройдет валидацию. Найдите все ошибки.



Ответ:

1. Нет корневого элемента
2. < 500 000 не пройдёт надо ‘<’ экранировать &lt;
3. TypeInfo, TypeName – неправильный порядок закрытия
4. BidCurrency – не закрыт тег
5. Значение атрибута должно быть в кавычках

Задача на логику

Есть автоматическое задание (task), которое выполняется каждую минуту. Таск регистрирует электронный документ «Результаты торгов». При регистрации документа возможно изменить атрибуты записи в БД. Таск также определяется SQL запросом, который собственно и выбирает лоты, по которым надо сформировать этот документ. Вот пример этого SQL:

SELECT Bid.Id, Purchase.OrgBuId, Purchase.TypeId

FROM Bid JOIN Purchase ON Purchase.Id = Bid.PurchaseId

WHERE Bid.StatusId = 6

*Ситуация*: вечером был перенос этого таска с тестового на боевой сервер. На следующий день вы приходите с утра и видите, что по всем подходящим лотам всю ночь каждую минуту формировался этот документ.

*Вопрос:* почему так произошло и что надо исправить?

Ответ: Bid.StatusId не менялся, соответственно надо в task занести изменения StatusId на другое число в зависимости от логики работы приложения.