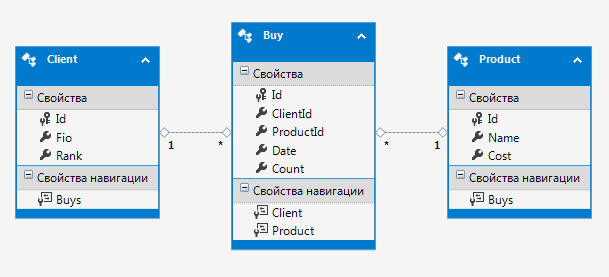
**Тестовые задания**

Есть структура данных:



**Client – клиенты:**

Fio – Фио клиента;

Rank – Ранг клиента.

**Product – товары:**

Name – наименование;

Cost – цена товара.

**Buy - покупки:**

ClientId – кто купил товар;

ProductId – какой товар был куплен;

Date – дата покупки;

Count – количество купленного товара.

Скрипт для создания структуры, и заполнения ее тестовыми данными находится в файле Init.sql.

**Для проверки заданий необходимо прислать исходные файлы проекта с инструкциями для сборки и запуска. Базу MSSQL – в виде sql скриптов.**

**Задание 1:**

Требуется написать одну хранимую процедуру в БД с передачей необходимых параметров для отображения результатов одного из следующих запросов:

* 1. Популярные – отобразить товары, которые покупались больше 3 раз клиентами с рангом больше 5.
  2. Актуальные – отобразить товары, которые покупались одним и тем же клиентом больше двух раз.
  3. Топ 5 – отобразить топ 5 товаров, в порядке убывания общей суммы покупок за всё время.

Все результаты запросов должны быть в одном виде с одинаковым набором столбцов.

Пример запроса для вывода всех товаров:

SELECT

p.[Id] AS 'Id товара'

, p.[Name] AS 'Наименование товара'

, p.[Cost] AS 'Цена'

, (SELECT

COUNT(\*)

FROM Buy b

WHERE

b.[ProductId] = p.[Id]) AS 'Количество покупок'

FROM Product p

|  |  |  |  |
| --- | --- | --- | --- |
| **Id товара** | **Наименование товара** | **Цена** | **Количество покупок** |
| 2 | Товар 1 | 255,00 | 4 |
| 3 | Товар 2 | 10,00 | 3 |
| 4 | Товар 3 | 60,00 | 13 |
| 5 | Товар 4 | 500,00 | 6 |
| 6 | Товар 5 | 1000,00 | 1 |
| … |  |  |  |

Необходимо учитывать оптимизацию запросов и добавить необходимые индексы.

**Задание 2:**

Необходимо разработать SPA (Single Page Application) приложение на React.JS (или на любой другой JS библиотеке или фреймворке), используя в качестве WEB-сервера ASP.NET. Можно реализовать Desktop приложение на C# (WPF, WinForms), что будет менее приоритетно при оценке задания.

Реализовать систему распределения ролей на стороне БД с применением хранимых процедур и дополнительных таблиц. На стороне приложения можно только включать/выключать роли, получать текущие роли (доступный функционал согласно ролям) из БД, а также отображать на странице кнопки меню только для доступных ролей. После перезагрузки страницы выбранные роли должны сохраняться.

Возможные роли:

* Аналитик – присутствует кнопка меню «Популярные».
* Менеджер – присутствует кнопка меню «Актуальные».
* Руководитель – присутствует кнопка меню «Топ 5».

На стороне клиента на странице должен быть блок включения и отключения ролей (одновременно может быть несколько ролей), блок меню из трех кнопок и блок данных.

Роли:

* Аналитик
* Менеджер
* Руководитель

Актуальные

Топ 5

Популярные

Блок данных

Так как строим SPA, то при переходе по пунктам меню не должно быть перезагрузки страницы. Ссылки перехода для кнопок:

1. Популярные - /#popular
2. Актуальные - /#actual
3. Топ 5 - /#top

При указании в адресной строке любого из вышеперечисленных URL в блоке «Блок данных» должны отобразиться соответствующие данные из **задания 1**.