Тестовое задание

1. **Навыки работы с платформой .Net и C#**

**Проблема бизнеса:**

Существует клиентское приложение, которому требуются данные о продажах в разрезе клиентов. Должна быть возможность получать как всю информацию, так и с фильтрацией по датам продаж, клиентам.

**Постановка задачи:**

Разработать web api, реализующий функционал получения продаж в разрезе клиентов. API поддерживает получение всей доступной информации по всем клиентам, по конкретному клиенту, по датам продаж.

Требования к реализации:

1. Onion architecture
2. REST модель
3. HTTP
4. C#, .Net Core 2.2.4
5. Следование соглашениям о наименовании для C#.
6. Хранить данные в БД. СУБД – SQL Server.
7. Проект должен использовать технологию внедрения зависимостей.
8. Unit тесты. Использовать NUnit либо XUnit

Требования к системе контроля версий: GIT

**Учебные материалы:**

C#: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tutorials/>, <https://metanit.com/sharp/>

Onion architecture: <https://docs.microsoft.com/ru-ru/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures>

REST: <https://docs.microsoft.com/ru-ru/azure/architecture/best-practices/api-design>

Net Core installing: <https://dotnet.microsoft.com/download/dotnet-core/2.2>

Следование соглашениям о наименовании для C#: <https://docs.microsoft.com/ru-ru/dotnet/standard/design-guidelines/general-naming-conventions>

Антипаттерны: [https://web.archive.org/web/20161231130810/http://wiki.c2.com/?AntiPatternsCatalog](https://web.archive.org/web/20161231130810/http:/wiki.c2.com/?AntiPatternsCatalog)

Паттерны: <https://martinfowler.com/eaaCatalog/>

Рефакторинг: <https://www.refactoring.com/catalog/>

Паттерны-рефакторинг на русском: <https://refactoring.guru/ru>