Тестовое задание для соискателя на позицию C# разработчика

**Описание задания:**

Требуется разработать библиотеку классов (коннектор), которая будет получать для IDM (система управления учетными данными) кадровые данные из удаленного источника (REST API). Для этого нужно реализовать интерфейс **IConnector** (см. приложение «Connector.cs»), там же будет указано правило формирование URL’ов.

До того, как IDM запросит первые данные по любому из методов, данные уже должны быть получены и провалидированы. При успешной валидации данные должны быть записаны в архив на сервере.

**Требования к реализации:**

1. На вход коннектор принимает базовый URL кадрового источника, логин и пароль (вымышленный URL, так как тестовые данные предоставлены в файлах \*.json; логин и пароль - тоже). Тип авторизации: Basic.
2. После получения, данные должны быть преобразованы по структуре IDM (все классы есть в приложении к заданию «Connector.cs») и провалидированы по следующим пунктам:

* если у сотрудника («employees.json») указано несуществующее подразделение («units.json»), то остановить работу коннектора (путем выброса ошибки) и записать в лог информацию об этой ошибке
* если нарушена иерархия подразделений, т.е. отсутствует какое-либо родительское подразделение, то остановить работу коннектора и записать в лог информацию
* если у сотрудника отсутствует дата приема на работу – работа коннектора не останавливается, но данный сотрудник не должен попасть в выходные данные для IDM, информация о таком сотруднике должна быть записана в лог

1. Работа с API должна учитывать работу с нетипичными для формата json данными. Например, пустую строку стоит принимать за null.
2. Методы интерфейса намеренно синхронные, это нужно учитывать, а не изменять.
3. На вход коннектор также принимает полный путь к папке, куда нужно создать архив с валидными данными. Архив должен состоять из 3-х файлов: employees.json, units.json, positions.json и содержать в себе соответствующие объекты. Название архива должно содержать текущую дату и время (ISO 8601) в часовом поясе Москвы (GMT+3).

**Требования к результату тестового задания:**

* срок реализации тестового задания – до 16 рабочих часов
* код размещен в github в приватном репозитории. Предоставить доступ для: [alex.d.tech@gmail.com](mailto:alex.d.tech@gmail.com), [tm.shuraleva@gmail.com](mailto:tm.shuraleva@gmail.com)
* код должен быть снабжен документацией в формате \*.md
* функции и классы должны быть снабжены комментариями, описывающими их поведение
* при разработке должны учитываться принципы чистого кода, а также классические паттерны проектирования и разработки
* реализуемый код должен быть спроектирован с учетом возможного расширения функционала, например, переключения на другой источник данных