

Техническое задание (ТЗ) для разработки проекта по базы данных о посетителях Фитнес-клуба

1. Общие сведения

Проект представляет собой консольное приложение для управления данными о посетителях Фитнес-клуба. Приложение позволяет генерировать случайные данные о посетителях фитнес-клуба и сохранять их в текстовый файл. Также присутствует система мониторинга посещений в клуб, которая записывает время и дату прихода и ухода.

2. Цель проекта

Создать приложение, которое будет:

1. Генерировать случайный абонемент в фитнес-клуб.
2. Мониторить данные о посещениях в фитнес-клуб.
3. Записывать данные о пользователях и посещениях в текстовые файлы.

3. Требования к функционалу

3.1 Классы

- **Класс Visitor:**
 - **Поля:**
 - id – уникальный идентификатор посетителя
 - countId – статистическая переменная отвечающая за генерацию уникальных идентификаторов
 - name – Имя посетителя.
 - abonement – количество дней на которые действует абонемент.
 - surname – Фамилия посетителя.
 - names – статистический вектор, отвечающий за генерацию имен.
 - surnames – статистический вектор, отвечающий за генерацию фамилий.

- abonements – статистический вектор, отвечающий за генерацию дней действия абонемента.
- **Методы:**
 - Конструктор:
 - Visitor(string name, string surname, string abonement) – конструктор, создающий объект посетителя фитнес-клуба
 - getId() – метод получения идентификатора продукта.
 - generateVisitor() – метод для генерации случайного посетителя.
 - toString() – метод, который возвращает значения данных пользователя.
- **Класс Database:**
 - **Поля:**
 - path – путь к файлу для хранения данных.
 - **Методы:**
 - Конструктор:
 - Database(string path) – конструктор, создающий объект базы данных, для записи информации.
 - addLine(string line) – метод добавления строки в файл.
 - removeLine(int id) – метод для удаления строки из файла по списку идентификаторов.
 - showDoc() – метод, показывающий содержимое файла.
 - splitStr(string str, char del) – метод для разделения строки по заданному разделителю.

- **Класс Timemanager:**

- **Поля:**

- Id – уникальный идентификатор.
 - idVisitor – уникальный идентификатор посетителя.
 - localDateTime – местное время и дата.
 - status – статус, определяющий приход или уход.
 - int countId – количество идентификаторов.

- **Методы:**

- **Конструктор:**

- Timemanager(int idVisitor, bool status) – конструктор, создающий объект посетителя фитнес-клуба
 - toString() – метод, который возвращает значения данных пользователя.

- **Класс Company:**

- **Поля:**

- visitors – уникальный идентификатор посетителя.
 - tmDb – База данных о времени и дате прихода и ухода в фитнес-клуб.

- **Методы:**

- incomeVisitor(Visitor* visitor) – метод для сбора информации о приходе посетителей.
 - outcomeVisitor(int id) – метод для сбора информации о уходе посетителей.

3.2 Основные функции

- Генерация случайных пользователей с помощью метода `generateVisitor()` и запись данных в файл с помощью метода `addLine(visitor->toString())`.
- Функции `inclomeVisitor(Visitor* visitor)` и `outcomeVisitor(int id)`, которые записывают данные о приходе и уходе посетителей в отдельный файл.