МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

КУРСОВОЙ ПРОЕКТ

ЗАЩИЩЕН С ОЦЕНКОЙ

РУКОВОДИТЕЛЬ

Ст.преподаватель Е.О. Шумова

должность, уч. степень, звание подпись, дата инициалы, фамилия

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К КУРСОВОМУ ПРОЕКТУ

Разработка приложения для организации взаимодействия объектов при заданных критериях

по дисциплине: ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. Z1431 М.Д. Быстров

подпись, дата инициалы, фамилия

Санкт-Петербург 2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

**Задание**

**на курсовой проект по дисциплине**

**«Объектно-ориентированное программирование»**

Студенту группы\_\_\_\_\_Z1431\_\_\_\_ **\_\_\_**Быстров М.Д.**\_\_**

№ группы Ф.И.О.

Тема «Разработка приложения для организации взаимодействия объектов при заданных критериях»

Исходные данные: Разработка иерархии классов для обеспечения работы гостиницы

Проект должен содержать:

* анализ предметной области
* разработку классов
* разработку тестового приложения
* оформление пояснительной записки по результатам выполнения проекта
* создание презентации к проекту

Срок сдачи законченного проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ст.преп. Е.О.Шумова

Дата выдачи задания 01.09.2023 г.

# Введение

Предметная область курсового проекта – обеспечение работы гостиницы. В ходе выполнения проекта должно быть спроектировано и реализовано приложение для учета и распределения по свободных номерам приезжих гостей.

Приложение должно отвечать следующим требованиям: работа с базой данных, графический интерфейс, использование концепции ООП и паттернов проектирования.

В разделах «Постановка задачи», «Проектирование классов» настоящей пояснительной записки содержатся: описание определенных на основе анализа предметной области сущностей, информация о выбранных технологиях и инструментах разработки, диаграммы разработанных иерархий классов, перечисление используемых паттернов проектирования.

Далее, в разделе «Разработка приложения» описаны детали реализации приложения, приводятся изображения, содержащие пользовательский интерфейс тестового приложения.

В разделе «Тестирование» продемонстрирована работа приложения с использованием различных наборов тестовых данных.

В Приложении 1 размещен полный исходный код реализованной программы.

# Постановка задачи

## Анализ предметной области

Предметной областью курсового проекта является работа гостиницы в части регистрации и заселения клиентов. Процесс регистрации начинается при первом получении информации о клиенте, далее регистрируется факт заселения и выселения клиента.

По итогам анализа предметной области выделены следующие основные сущности:

1.1.1 Основные сущности

* Клиент (постоялец, гость) гостиницы
* Номер гостиницы
* Спальня номера
* Кровать в спальне

1.1.2 Вспомогательные сущности

* Поставщик данных БД
* Требования к номеру
  + Требование к вместимости
  + Требование к спальным местам
  + Требование к количеству комнат
  + Требование к количеству ванных комнат
  + Требование к этажу номера
  + Требование к площади номера
  + Требования к классу номера
* Набор требований к номеру
* Строитель набора требований к номеру
* Поставщик набора номеров в соответствии с требованиями к номеру

## Формулировка технического задания

Спроектировать и реализовать программу для учета гостей гостиницы и данных об их проживании.

Программа должна иметь графический пользовательский интерфейс (GUI), выполняться в операционной системе MS Windows.

Язык программирования – C#. Среда разработки – Visual Studio.

Технология для создания пользовательского графического интерфейса – Windows Forms.

# Проектирование классов

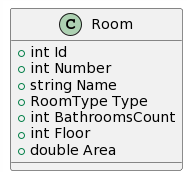
## 2.1 Классы сущностей

2.1.1 Класс «Room»

Класс «Room» описывает номер – сущность предметной области. Каждый экземпляр класса описывает один номер гостиницы.

Описание полей:

* Id – уникальный идентификатор
* Number – числовой код номера, в соответствии с распределением номеров в гостинице
* Name – название номера
* Type – тип (класс) номера
* BathroomsCount – количество ванных комнат в номере
* Floor – номер этажа, на котором располагается номер
* Area – площадь номера (м2)



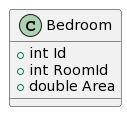
Все поля являются свойствами языка C#, т.е. имеют неявные методы доступа (getter, setter), инкапсулирующие поля. Далее во всех описаниях классов подразумевается, что доступ к данным, содержащимся в полях экземпляров классов, происходит посредством неявных вызовов методов доступа.

2.1.2 Класс «Bedroom»

Класс «Bedroom» описывает спальню, принадлежащую номеру.

Описание полей:

* Id – уникальный идентификатор комнаты
* RoomId – идентификатор комнаты, которой принадлежит спальня
* Area – площадь спальни

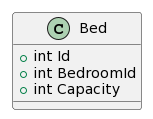


2.1.3 Класс «Bed»

Класс «Bed» описывает кровать, расположенную в спальне номера.

Описание полей:

* Id – уникальный идентификатор кровати
* BedroomId – идентификатор спальни, в которой расположена кровать
* Capacity – вместимость кровати (количество человек)

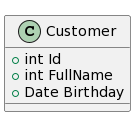


2.1.4 Класс «Customer»

Класс «Customer» описывает гостя гостиницы (постоялец, заказчик).

Описание полей:

* Id – уникальный идентификатор гостя
* FullName – ФИО
* Birthday – дата рождения

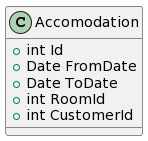


2.1.5 Класс «Accomodation»

Класс «Accomodation» описывает заселение постояльца в номер.

Описание полей:

* Id – уникальный идентификатор заселения
* FromDate – дата заселения
* ToDate – дата выезда
* RoomId – ИД комнаты
* CustomerId – ИД постояльца



2.1.6 Диаграмма классов

Поскольку сущности предметной области не связаны между собой отношениями, отражаемыми диаграммой классов, диаграмма классов для классов, описывающих сущности предметной области, приведена не будет.

## Классы хранения данных

* + 1. Класс «BaseDbContext»

Описание полей:

* \_databaseFullFileName – путь к файлу базы данных

Описание методов:

* setDatabaseFullFileName – установить путь к файлу базы данных
* getDatabaseFullFileName() – получить путь к файлу базы данных
* selectDatabaseFile() – предоставить пользователю выбрать файл базы данных (либо имя нового файла для хранения данных)(статический)
* getRooms – получить номера
* getRoom – получить номер по идентификатору
* addRoom – добавить новый номер
* updateRoom – обновить существующий номер
* deleteRoom – удалить номер
* getAccomodations – получить все данные заселений
* getAccomodation – получить заселение по идентификатору
* addAccomodation – создать новое заселение
* updateAccomodation – обновить заселение
* deleteAccomodation – удалить заселение
* getBedrooms – получить спальни
* getRoomBedrooms – получить спальни по идентификатору номера
* getBedroom – получить спальню по идентификатору
* addBedroom – добавить спальню
* updateBedroom – обновить спальню
* deleteBedroom – удалить спальню
* getBeds – получить кровати
* getBedroomBeds – получить кровати по идентификатору спальни
* getBed – получить кровать по идентификатору
* addBed – добавить кровать
* updateBed – обновить кровать
* deleteBed – удалить кровать
* getCustomers – получить постояльцев
* getCustomer – получить постояльца по идентификатору
* addCustomer – добавить постояльца
* updateCustomer – обновить постояльца
* deleteCustomer – удалить постояльца
* clearDatabase – очистить базу данных
  + 1. Класс «HostelDbContext»

Описание полей:

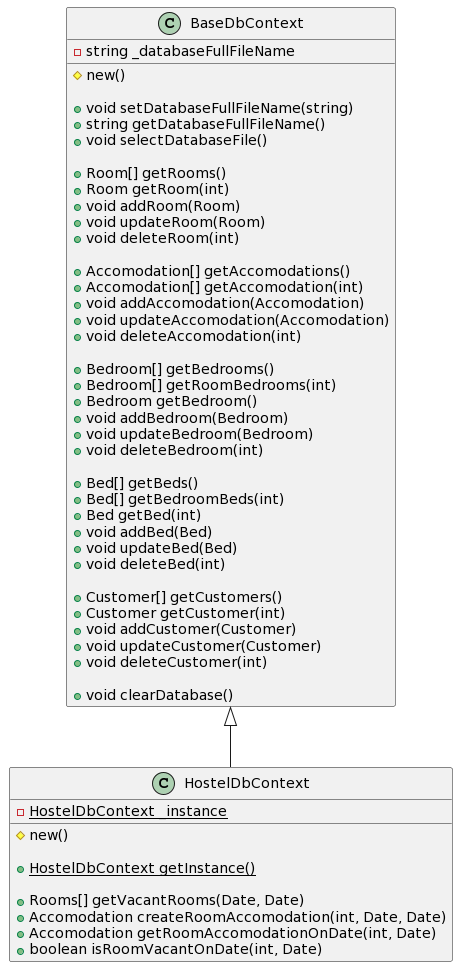
* \_instance – экземпляр контекста (статическое)

Описание методов:

* getInstance – (статическое) – получить экземпляр контекста базы данных
* getVacantRooms – получить доступные комнаты на период дат
* createRoomAccomodation – создать заселение постояльца в комнату
* getRoomAccomodationOnDate – получить заселение в комнату на дату
* isRoomVacantOnDate – проверить, свободна ли комната на дату

В классе HostelDbContext реализован паттерн «Singleton». Назначение и устройство паттерна будет описано в разделе 2.5 «Используемые паттерны проектирования».

* + 1. Диаграмма классов



## Управляющие классы

* + 1. Класс «IRequirement»

Класс «IRequirement»

* + 1. Класс «CapacityRequirement»

Класс «CapacityRequirement»

* + 1. Класс «RoomTypeRequirement»

Класс «RoomTypeRequirement»

* + 1. Класс «BedRequirement»

Класс «BedRequirement»

* + 1. Класс «BedroomRequirement»

Класс «BedroomRequirement»

* + 1. Класс «BathroomRequirement»

Класс «BathroomRequirement»

* + 1. Класс «FloorNumberRequirement»

Класс «FloorNumberRequirement»

* + 1. Класс «AreaRequirement»

Класс «AreaRequirement»

* + 1. Класс «RequirementSet»

Класс «RequirementSet»

* + 1. Класс «RequirementSetBuilder»

Класс «RequirementSetBuilder»

* + 1. Класс «RequirementRoomProvider»

Класс «RequirementRoomProvider»

* + 1. Диаграмма классов