Министерство образования Калининградской области

государственное бюджетное учреждение Калининградской области

профессиональная образовательная организация

«Колледж информационных технологий и строительства»

(ГБУ КО ПОО «КИТиС»)

**Отчет по учебной практике**

УП.01 Разработка модулей программного обеспечения для компьютерных систем

по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Специальность 09.02.07 «Информационные системы и программирование»

Сроки прохождения практики:

с «02» декабря 2020 г. по «22» декабря 2020 г.

Место практики ГБУ КО ПОО «КИТиС»

|  |  |
| --- | --- |
| Выполнил: | студент 3 курса,  группы ИС 18-2к  <Даниелян Гор Артурович>  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |
| Проверила: | Большакова-Стрекалова Анна Викторовна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |

Калининград

2020

# Содержание

[Введение 3](#_Toc59024669)

[1. 1 Описание предметной области 4](#_Toc59024670)

[1. 2 Концептуальная модель предметной области 5](#_Toc59024671)

[1. 3Диаграмма классов (ClassDiagram) 6](#_Toc59024674)

[1.4 Построение ER-диаграммы 8](#_Toc59024675)

[1.5 Диаграмма ER- экземпляров 9](#_Toc59024679)

[1.6 Построение диаграммы деятельности 13](#_Toc59024681)

[Заключение 18](#_Toc59024682)

# Разработка АИС

# Введение

Под автоматизированной информационной системой (АИС) мы понимаем, совокупность программно-аппаратных средств, предназначенных для автоматизации деятельности, связанной с хранением, передачей и обработкой информации.

АИС, основанная на базе данных, служит для сбора, накопления, хранения информации, а также её эффективного использования для различных целей. Информация представляется в виде данных, хранимых в памяти ЭВМ. При проектировании АИС, с одной стороны, решается вопрос о том, какие сведения и для каких целей будут содержаться в системе, с другой – как соответствующие данные будут организованы в памяти ЭВМ, и как они будут обрабатываться при эксплуатации АИС.

АИС может быть определена как комплекс автоматизированных информационных технологий, предназначенных для информационного обслуживания – организованного непрерывного технологического процесса подготовки и выдачи потребителям научной, управленческой и др. информации, используемой для принятия решений, в соответствии с нуждами для поддержания эффективной деятельности.

Основной причиной создания и развития АИС является необходимость ведения учёта информации о состоянии и динамике объекта, новому аэропорту города Калининград которому посвящена система. На основании информационной картины, создаваемой системой, руководители различного звена могут принимать решения об управляющих воздействиях с целью решения текущих проблем.

Целью учебной практики служит создать программный продукт обеспечивающий автоматизацию работы, для удобного и быстрого доступа к информации, характеризующей работу аэропорта города Калининграда.

1. 1 Описание предметной области

Аэропорт – сложная система, основной целью которой является качественное обслуживание клиентов, которое заключается в транспортировке пассажиров и грузов, предоставление услуг, направленных на улучшение удобства перелёта.

Вместе с этим необходимо обеспечить выполнение ряда других задач, таких как выбор, и зарплата персонала, поддержание оборудования в хорошем состоянии, ремонт, обеспечение авиационной безопасности, составление и изменение расписания полётов, продажа и бронирование билетов и др.

Работа современного аэропорта невозможна без автоматизации производственных и бизнес-процессов. Так как аэропорт – большая и сложная система, будем рассматривать упрощённую модель, в которой сущностями будут самолёты, пилоты, пассажиры и багаж, а связями – рейсы и принадлежность багажа.

### 1. 2 Концептуальная модель предметной области

# Концептуальная модель отражает состав и взаимодействие объектов базы данных. Для этой цели разработано несколько систем соглашений о представлении информации, содержащейся в базе данных.

# Одним средством концептуального моделирования данных является универсальный язык моделирования UML. UML предоставляет средства для создания визуальных моделей, которые единообразно понимаются всеми разработчиками, вовлеченными в проект, и являются средством коммуникации в рамках проекта. Диаграмма в UML - это графическое представление набора элементов. Диаграммы рисуют для визуализации системы с разных точек зрения.

Для наглядного отображения оптимально подходит диаграмма вариантов использования. Этот вид диаграмм позволяет создать список операций, которые выполняет система, используется при описании бизнес процессов автоматизируемой предметной области, определении требований к будущей программной системе. Отражает объекты, как системы, так и предметной области и задачи, ими выполняемые.

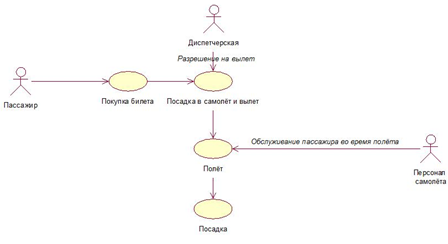


Рисунок 1 – Диаграмма вариантов использования

На рисунке 1 показана диаграмма вариантов использования для моей информационной системы. Она отображает основную концепцию разрабатываемой информационной системы. В данном случае:

Актёр – пассажир – человек который покупал билет самолета и , которому необходим перелёт.

Актёр – диспетчерская – разрешает или запрещает вылет самолёта на намеченный рейс.Это один из главных работников аэропорта потому что он можно сказать регулирует воздушное пространство и ежедневно из его работы зависеть жизнь миллион человека

Актёр – персонал – работники самолёта, обеспечивающие контроль за пассажирами и их обслуживание во время полёта.

Из диаграммы видим, что пассажир приобретает билет и занимает своё место в самолёте, когда все пассажиры на месте, диспетчерская даёт разрешение на вылет. Во время полёта пассажир взаимодействует с персоналом. Следующий и заключительный этап – посадка.

1. 3 Диаграмма классов (ClassDiagram)

Диаграмма классов отражает, в частности, различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывает их внутреннюю структуру и типы отношений.

Диаграмма классов (classdiagram) — диаграмма языка UML, на которой представлена совокупность декларативных или статических элементов модели, таких как классы с атрибутами и операциями, а также связывающие их отношения.

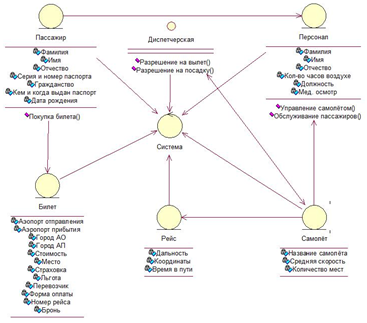


Рисунок 2 – Диаграмма классов

На рисунке 2 представлена диаграмма классов моего проекта.

Система – управляющий класс (controlclass), контролирующий работу всей системы и хранение данных. Диспетчерская – является интерфейсом (interfaceklass) и осуществляет 2 операции. Остальные объекты, представленные на диаграмме, являются классами-сущностями (entityclass): пассажир, персонал, самолёт, рейс и билет. Для каждого из них определены атрибуты, а также операции. На диаграмме видно, что все классы взаимодействуют с системой, а также некоторые друг с другом.

Диаграмма прецедентов — диаграмма, отражающая отношения между [актёрами](http://ru.wikipedia.org/wiki/%D0%90%D0%BA%D1%82%D1%91%D1%80_(UML)) (человеком или системой) и [прецедентами](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82_(UML)) (последовательностью действий) и являющаяся составной частью модели прецедентов.

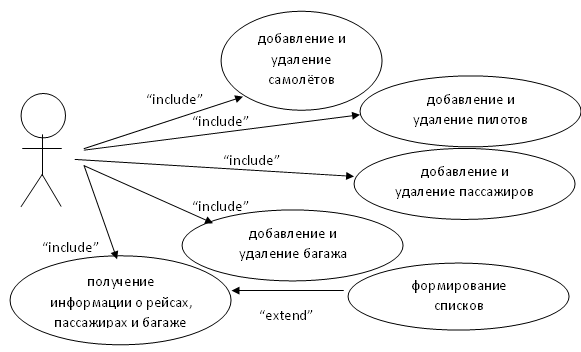


Рисунок 3 – Диаграмма прецедентов

На рисунке 3 описывается функциональное значение системы, то есть что система будет выполнять в процессе своего функционирования.

Пользователь имеет доступ к таким возможностям программы как:

- добавление и удаление самолётов;

- добавление и удаление пилотов;

- добавление и удаление пассажиров;

- добавление и удаление багажа;

- получение информации о рейсах, пассажирах и багаже.

Получение информации о рейсах, пассажирах и багаже может выполняться с помощью формирования списков.

1.4 Построение ER-диаграммы

Рассмотрим предметную область «АИС учет работы редакции аэропорта». Проанализировав ее, выделим часть, которую будем автоматизировать. Список сущностей, которые я выделил из предметной области, будет выглядеть так: самолёты, пилоты, пассажиры и багаж.

Связи между объектами можно сформировать в список следующего вида: рейс (связь: самолёт – пилот – пассажир ), принадлежит (связь: пассажир – багаж ).

Связи между сущностями отображены на ER-диаграмме предметной области "АИС учет работы редакции аэропорта "

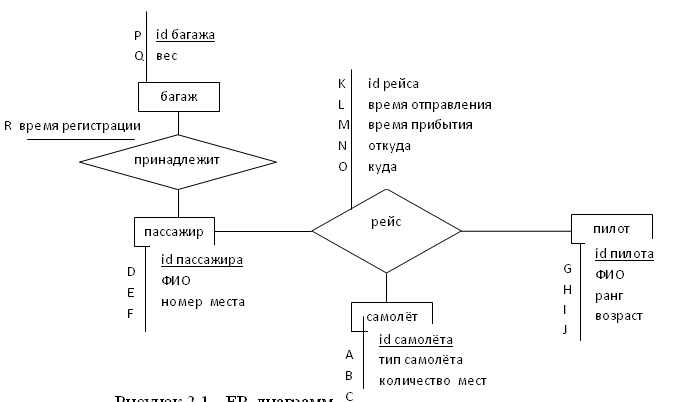


Рисунок 4 – ER диаграмма аэропорта

### 1.5 Диаграмма ER- экземпляров

Для понимания логики работы предметной области, а также установления степени связности и класса принадлежности таблиц разработана диаграмма ER-экземпляров (рис.5)

пассажир 1

багаж 1

пассажир 2

багаж 2

пассажир 3

Рисунок 5 – Диаграмма ER- экземпляров

Построение диаграммы последовательностей

Основными элементами диаграммы последовательности являются обозначения объектов (прямоугольники с названиями объектов), вертикальные "линии жизни" ("Пассажир", "Сотрудник аэропорта", "Инспектор по досмотру", "Транспортировка багажа"), отображающие течение времени, прямоугольники, отражающие деятельность объекта или исполнение им определенной функции (прямоугольники на пунктирной "линии жизни"), и стрелки, показывающие обмен сигналами или сообщениями между объектами.

Входом в диаграмму последовательностей служит выбор и покупка билета, далее Пассажир, взаимодействуя с системой и Сотрудником аэропорта для дальнейшего прохождения регистрации на рейс. Далее Сотрудник проверяет билет и если есть багаж, то отправляет его на погрузку. После данной процедуры идет выбор сиденья на воздушном судне и выдача посадочного билета.

Потом наступает очередь инспектора, который проверяет билеты и проводит осмотр пассажиров. После удачного прохода всех проверок, пассажир идет на посадку в самолет, как это показано на рисунке 6.

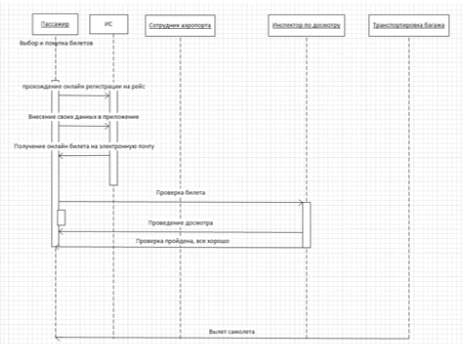


Рисунок 6 – Диаграмма последовательностей для online регистрации пассажира и проведения осмотра перед полетом

Как мы можем увидеть из диаграмм, действие сотрудников аэропорта в регистрации пассажиров не требуется и, соответственно, экономятся средства и время.

Также давайте распишем спецификацию для каждого отдельного варианта использования и сделаем диаграмму последовательностей (рис.7), (рис.8).

Вход в систему.

Наименование: Вход в систему.

Actor, взаимодействующие с прецедентом: Клиент/Пассажир.

Описание: Вход в систему служит для того, чтобы клиент мог в своем личном кабинете осуществлять регистрацию на предстоящий рейс по прибытии в аэропорт.

Приоритетный маршрут: Пользователь не может быть зарегистрирован на рейс без ввода данных билета и данных самого пассажира.



Рисунок 7 – Диаграмма последовательностей для прецедента "Вход в систему"

Регистрация.

Наименование: Регистрация.

Actor, взаимодействующие с перцедентом: Клиент/Пассажир.

Описание: Регистрация на рейс проходит в процессе внесения данных о пассажире

Предположение: Регистрация обязательна для дальнейшего прохождения контроля перед полетом.

Постусловия: После успешной регистрации, пассажир проходит на досмотр, а позже на посадку в самолет.

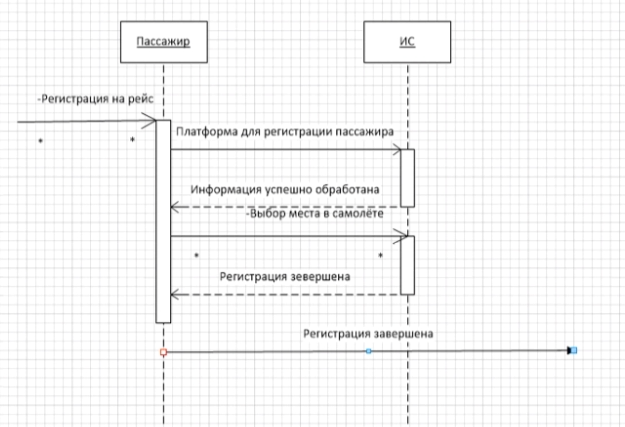


Рисунок 8 – Диаграмма последовательностей для прецедента "Регистрация

Построение диаграммы состояний

Каждая диаграмма состояний в UML описывает все возможные состояния одного экземпляра определенного класса и возможные последовательности его переходов из одного состояния в другое, то есть моделирует все изменения состояний объекта как его реакцию на внешние воздействия.

Диаграммы состояний чаще всего используются для описания поведения отдельных объектов, но также могут быть применены для спецификации функциональности других компонентов моделей, таких как варианты использования, актеры, подсистемы, операции и методы. Так была разработана диаграмма состояний (рис.9).

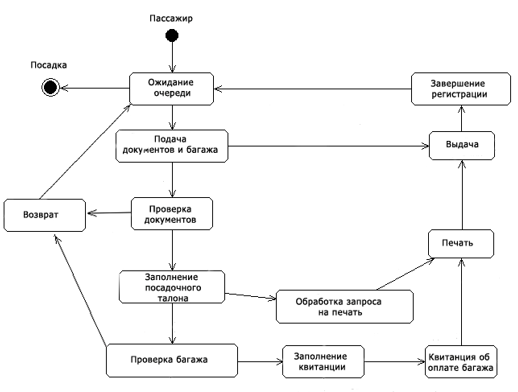


Рисунок 9 – Диаграмма состояний Пассажира

1.6 Построение диаграммы деятельности

Каждая диаграмма деятельности акцентирует внимание на последовательности выполнения определенных действий, которые в совокупности приводят к получению желаемого результата. Диаграмма деятельности – UML диаграмма, на которой показаны действия, состояния которых описано на диаграмме состояний. Так была построена диаграмма деятельности (рис.10).

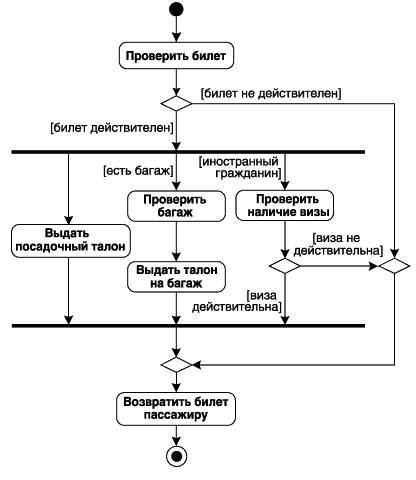


Рисунок 10 – Диаграмма регистрации пассажиров в аэропорту

Первоначально выполняется деятельность по проверке билета. В случае если билет не действителен, он возвращается пассажиру, при этом никаких дополнительных действий не выполняется.

Если же билет действителен, то пассажиру выдается посадочный талон. В дополнение к этому проверяется гражданство и наличие багажа у пассажира. Если есть багаж, то его проверка может быть выполнена параллельно, по результатам которой пассажиру выдается талон на багаж.

Если пассажир является иностранным гражданином, то дополнительно проверяется наличие у него визы. Если виза действительна, то проверка завершается успешно, и пассажир с возвращенным ему билетом может проследовать на посадку.

Если же виза окажется не действительной, то для этого пассажира посадка на данный рейс оказывается невозможной. В этом случае ему не выдается посадочный талон и талон на багаж, в случае его наличия, поскольку происходит прекращение всех выполняемых сотрудниками аэропорта действий. Была создана БД Аirport Sql в нем 12 основных таблицы

* AirCompany
* AirPort
* Buyer
* City
* Country
* Events
* Flight
* Passengers
* Payment
* Plane
* Tickets

Вот данные из них

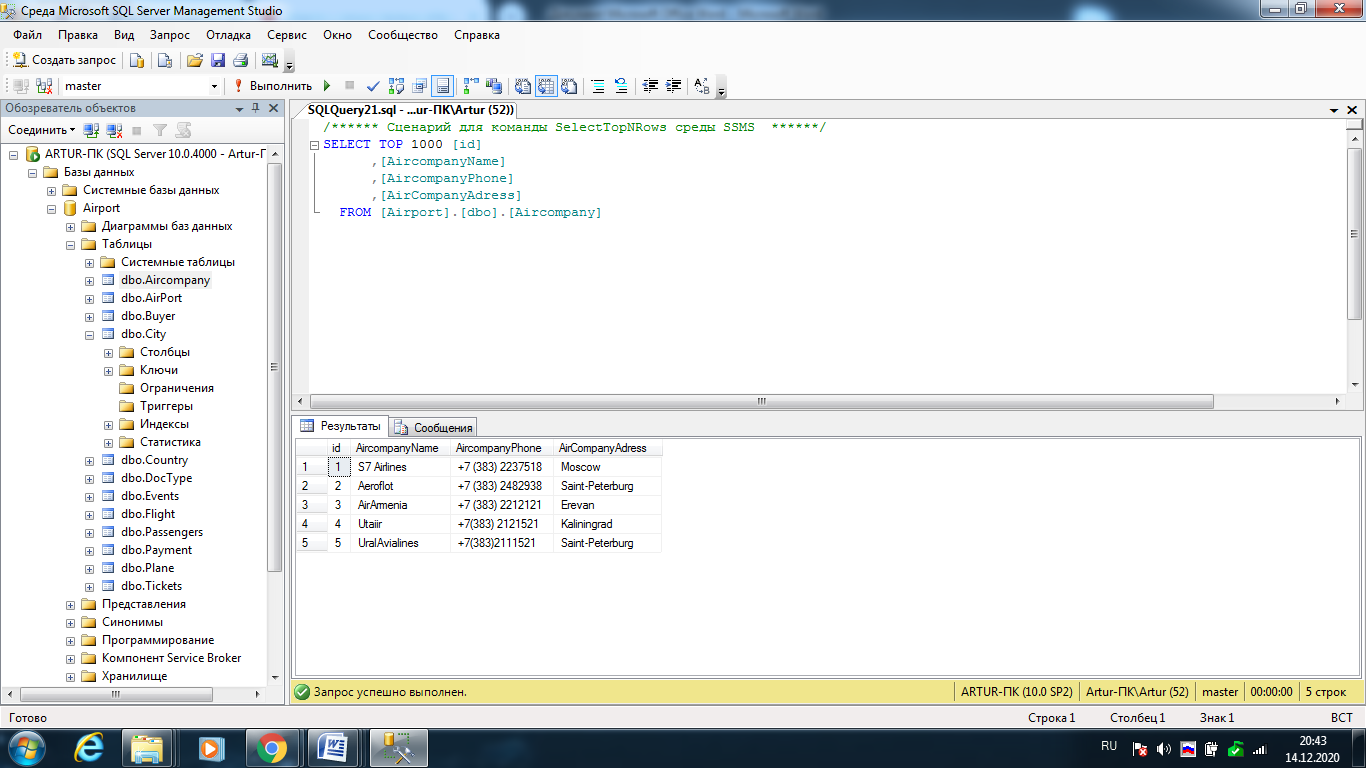


Рисунок 11 – Таблица AirCompany

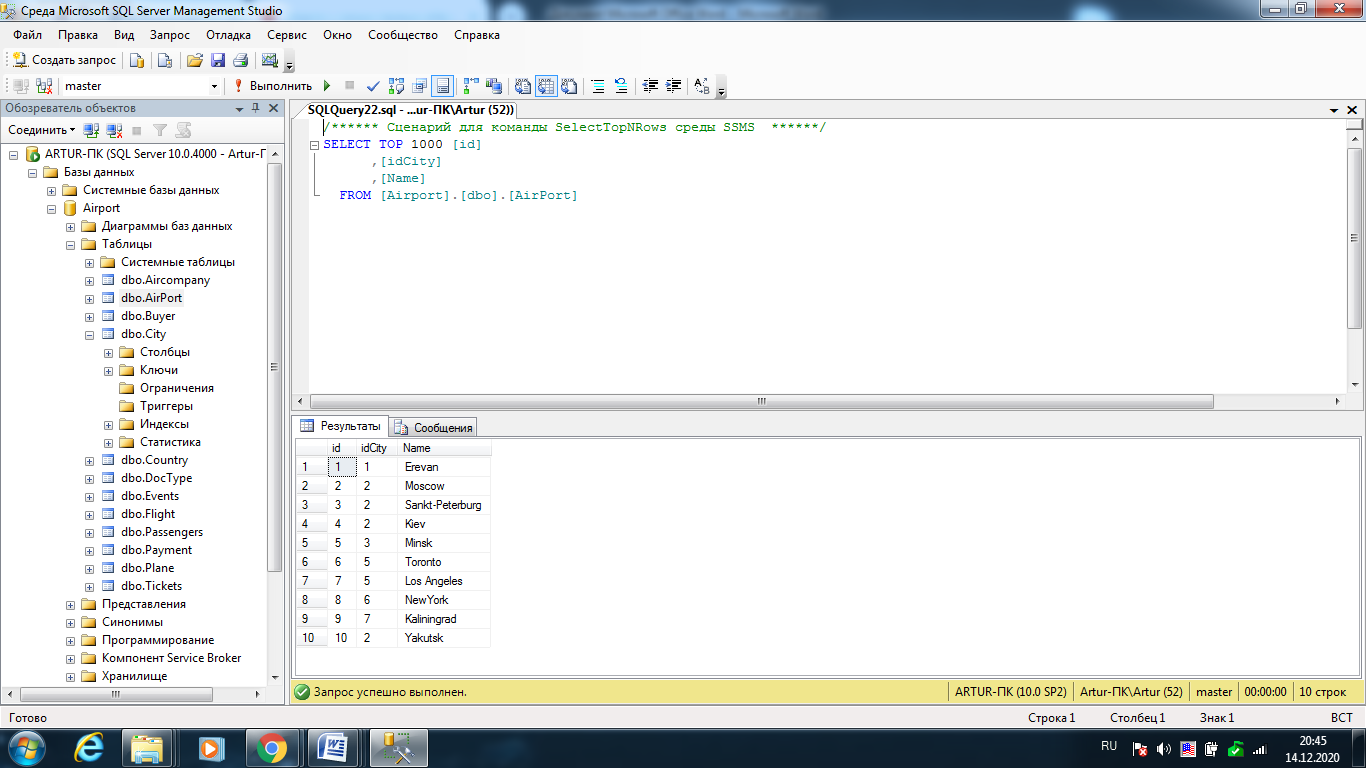


Рисунок 12 – Таблица AirPort

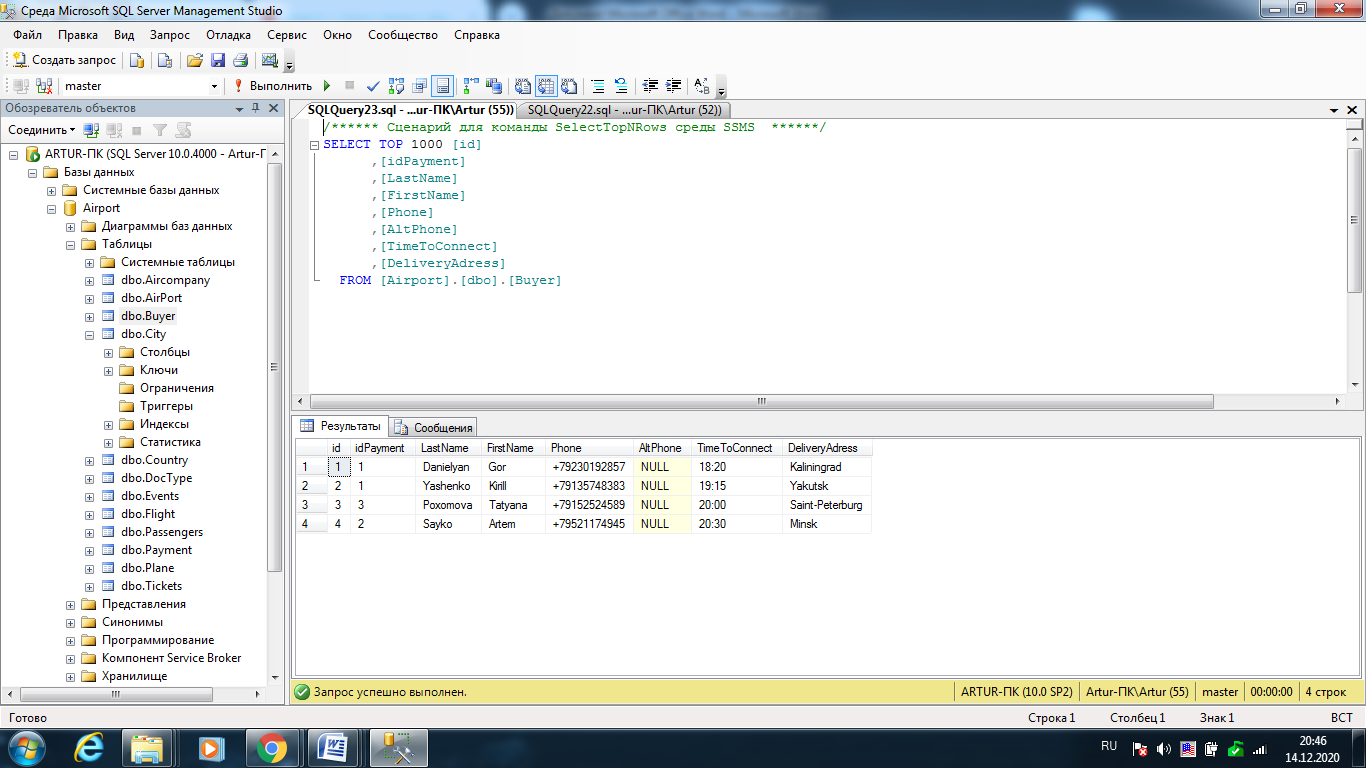


Рисунок 13 – Таблица Buyer

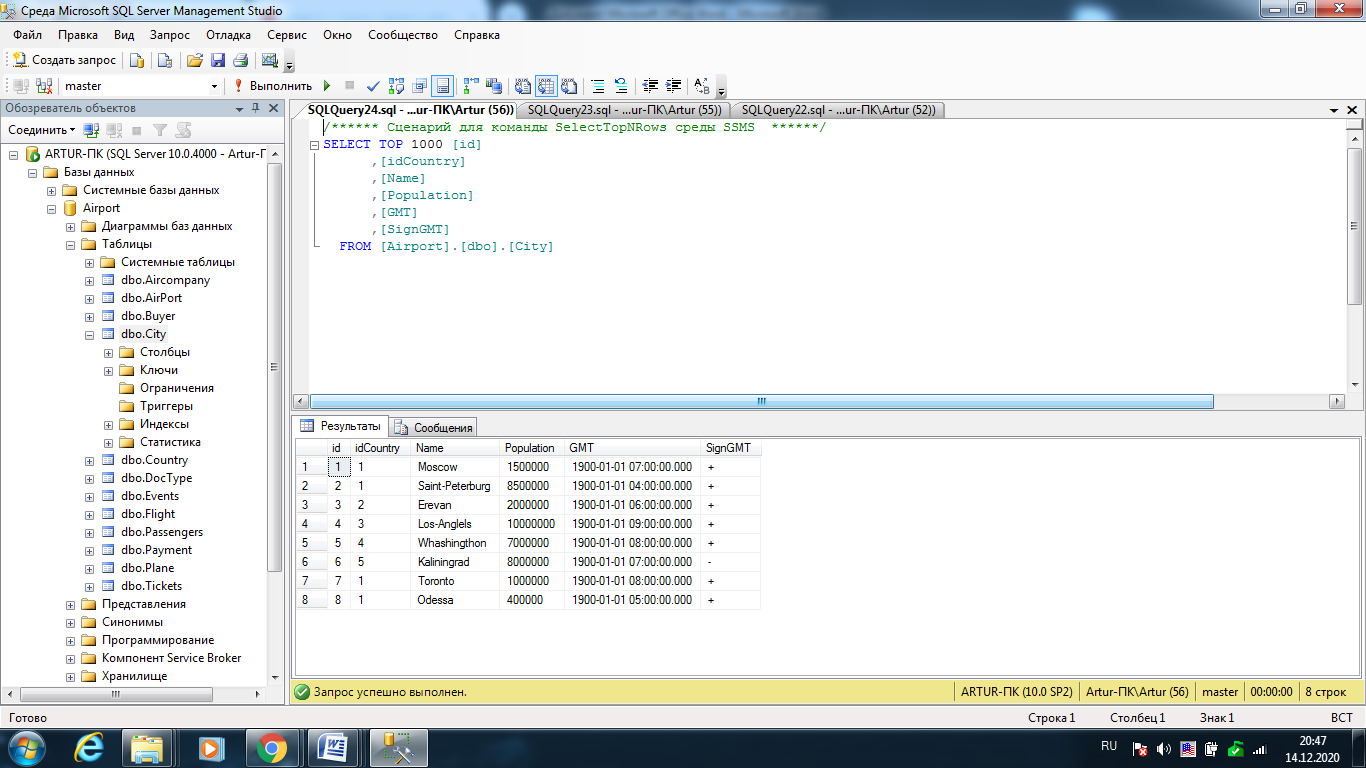


Рисунок 14 – Таблица City

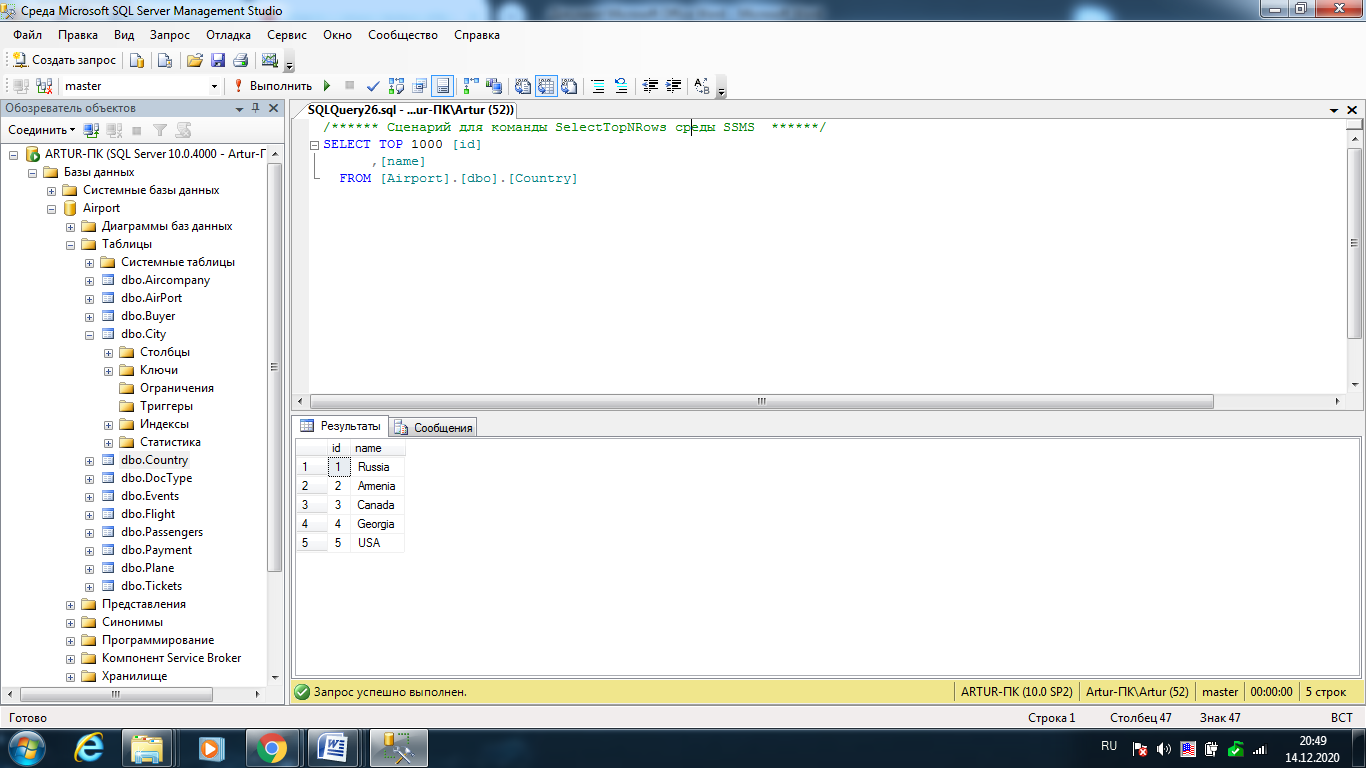


Рисунок 14 – Таблица Country

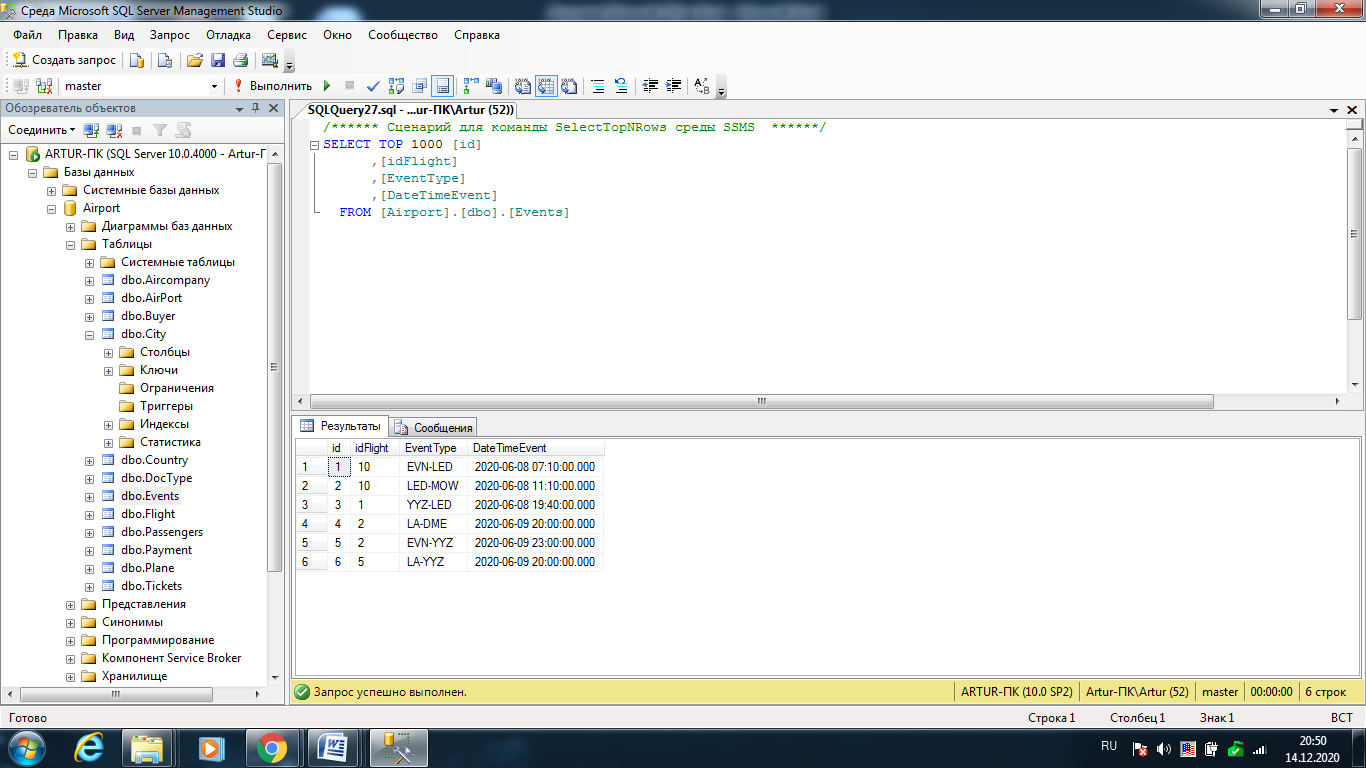


Рисунок 15 – Таблица Events

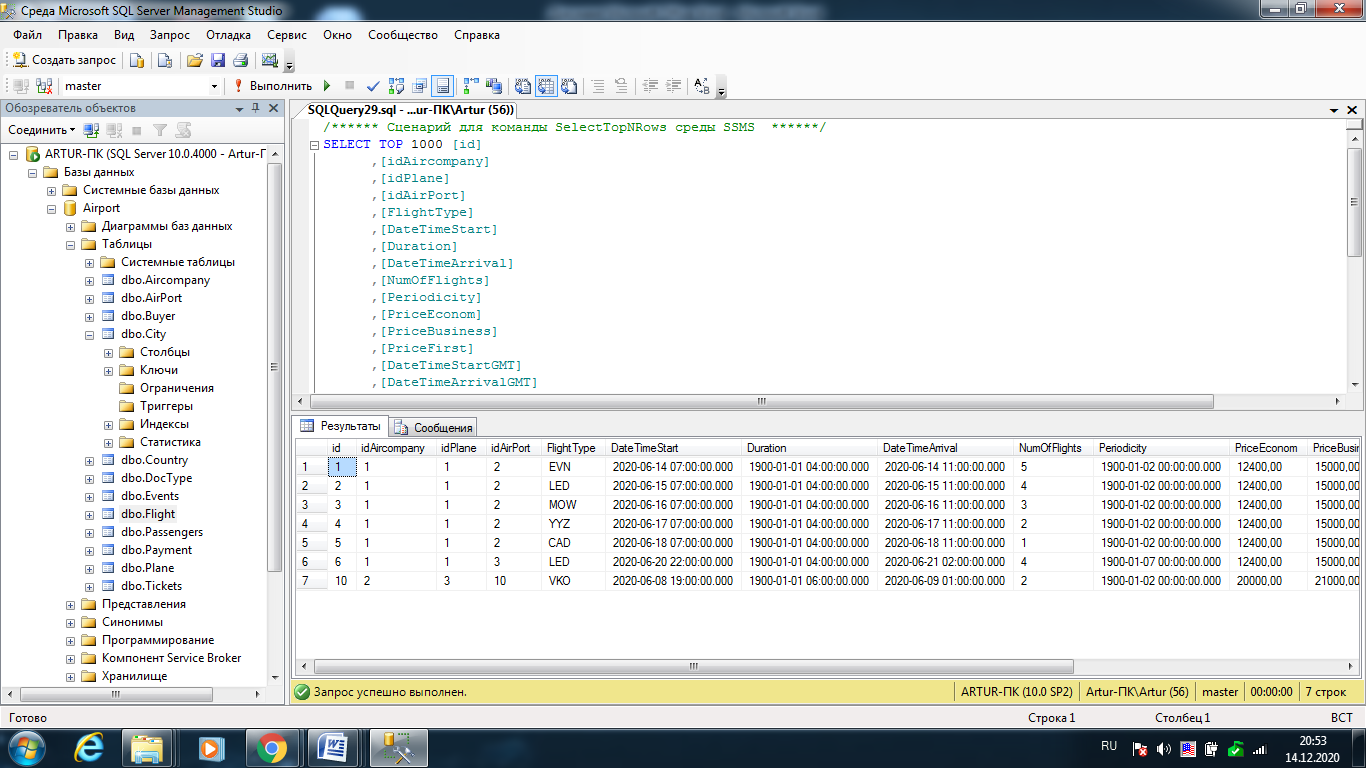


Рисунок 16 – Таблица Fights

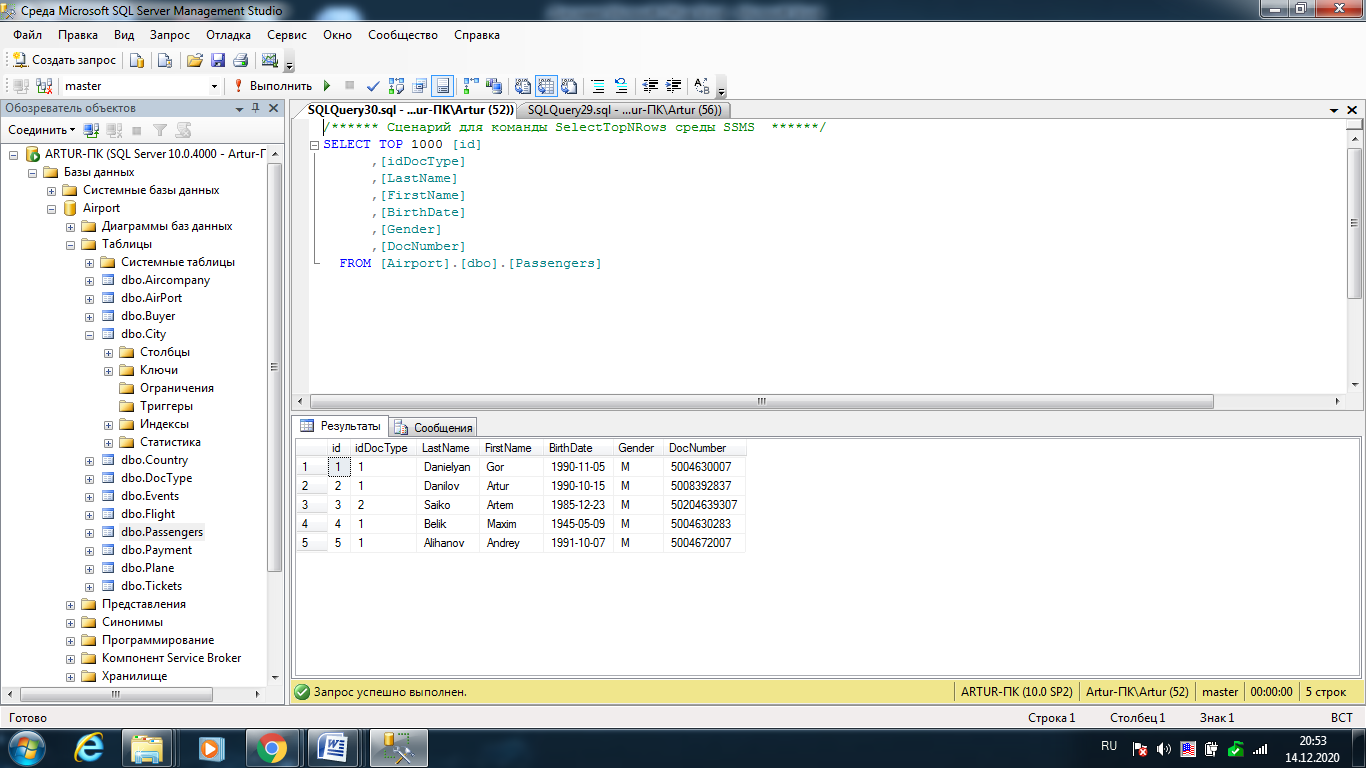


Рисунок 17 – Таблица Passengers

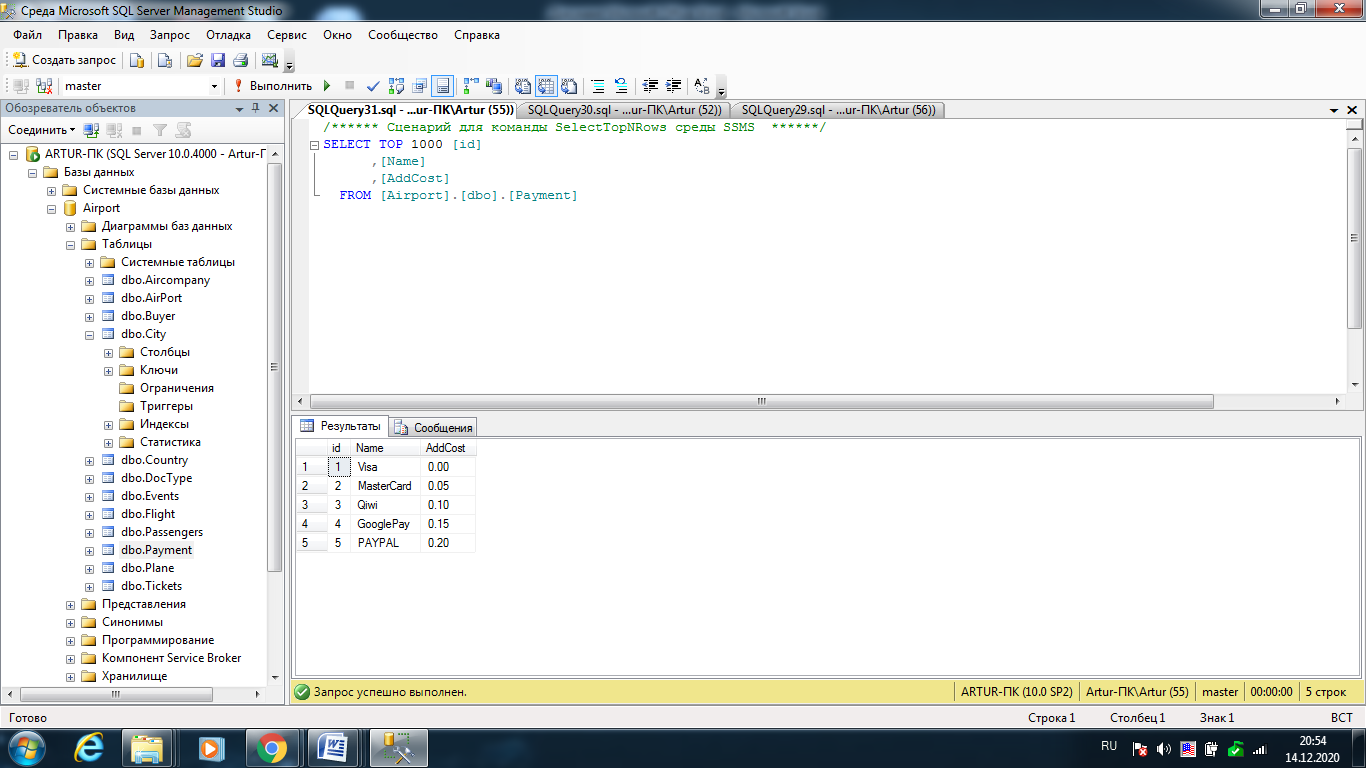


Рисунок 18 – Таблица Payment

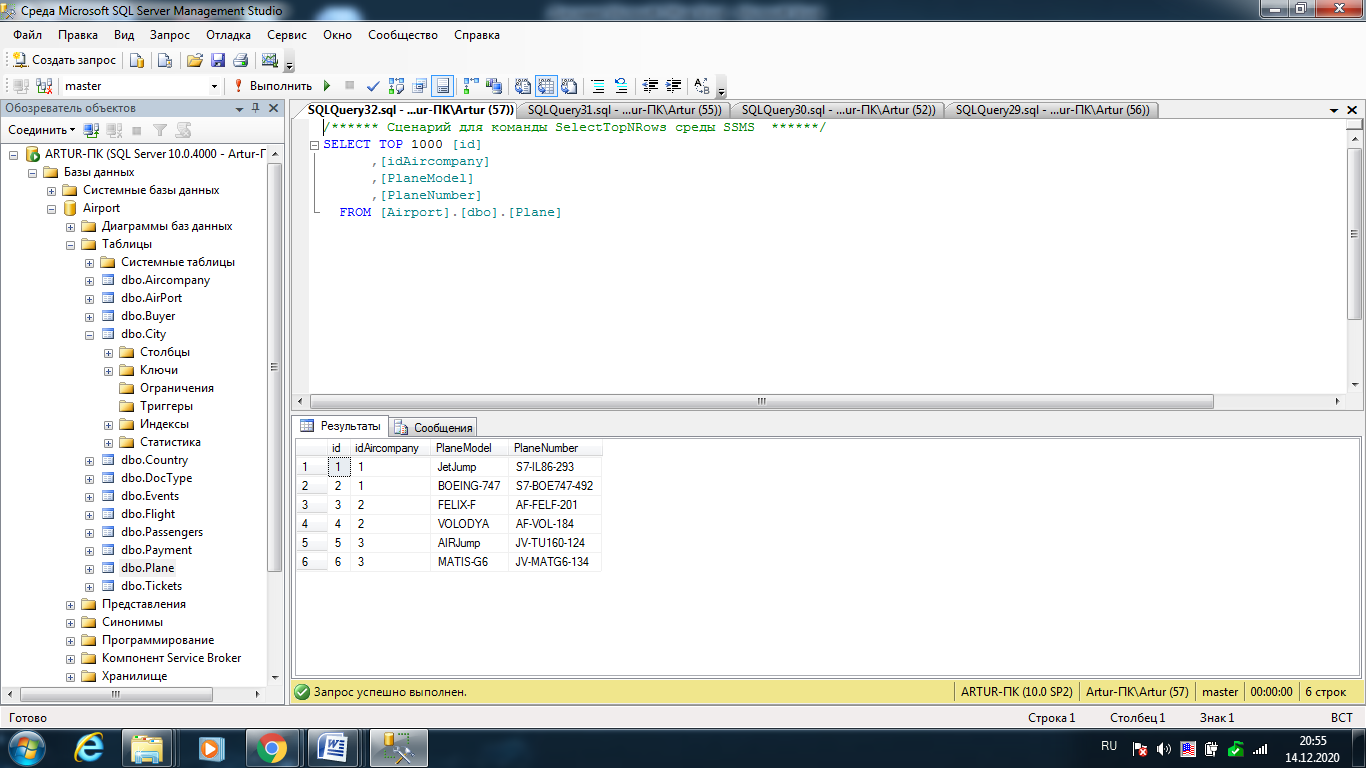


Рисунок 19 – Таблица Plane

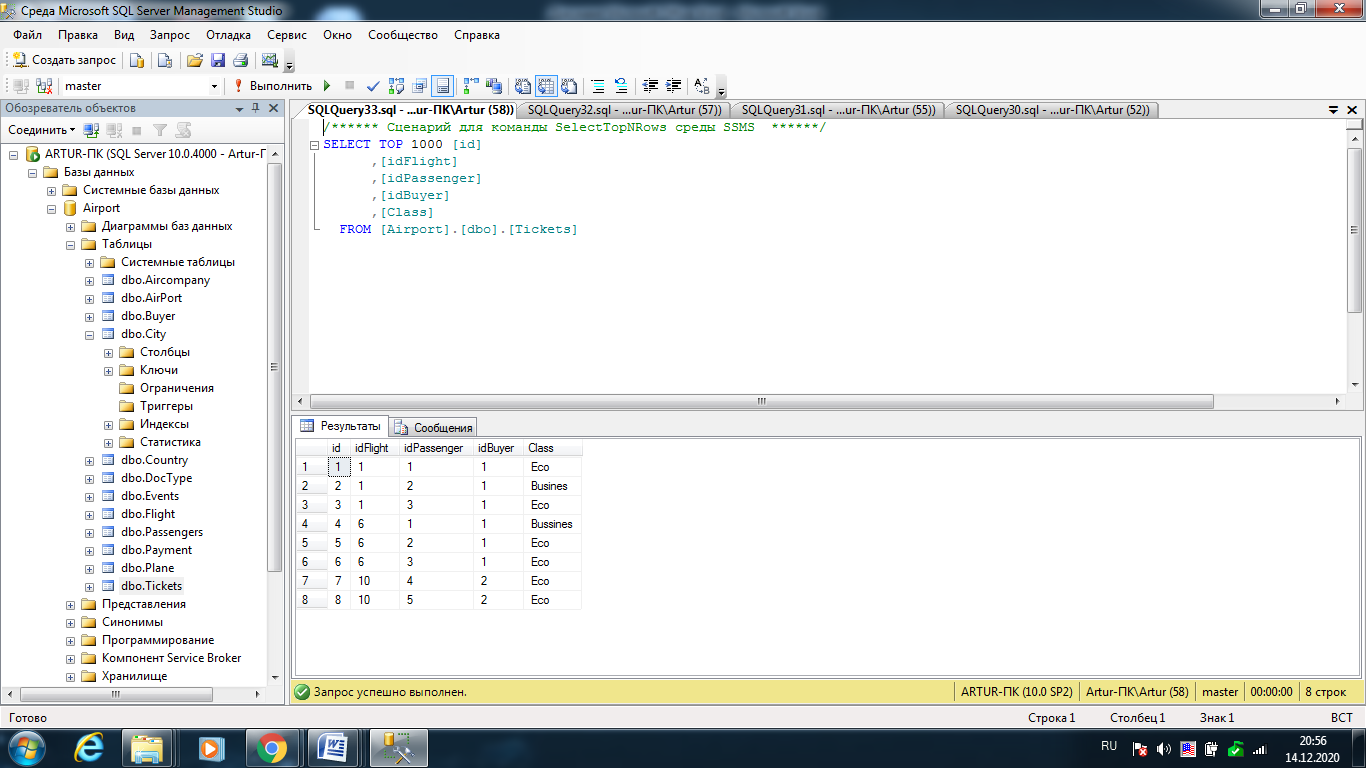


Рисунок 19 – Таблица Tickets

# Заключение

Была сформулирована и создана АИС Аэропорта на основе БД sql АИС, основанная на базе данных, служит для сбора, накопления, хранения информации. АИС может быть определена как комплекс автоматизированных информационных технологий, предназначенных для информационного обслуживания – пользователя.