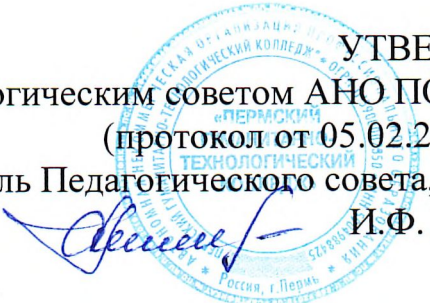


**Автономная некоммерческая организация профессионального образования
«ПЕРМСКИЙ ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКИЙ КОЛЛЕДЖ»
(АНО ПО «ПГТК»)**

УТВЕРЖДЕНА
Педагогическим советом АНО ПО «ПГТК»
(протокол от 05.02.2026 № 01)
Председатель Педагогического совета, директор
И.Ф. Никитина



**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
МЕЖДИСЦИПЛИНАРНОГО КУРСА**

**МДК 01.01. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БАЗ
ДАННЫХ**

для специальности

09.02.11 Разработка и управление программным обеспечением
(код и наименование специальности)

Квалификация выпускника
Программист

Форма обучения
Очная

Пермь 2026

Фонд оценочных средств междисциплинарного курса МДК 01.01. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БАЗ ДАННЫХ составлен в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением (утвержден приказом Министерства Просвещения Российской Федерации от 24 февраля 2025 г. N 138).

Программа предназначена для студентов и преподавателей АНО ПО «ПГТК».

Автор – составитель: Могильникова Н.С., старший преподаватель.

Фонд оценочных средств междисциплинарного курса рассмотрена и одобрена на заседании кафедры математических и естественно-научных дисциплин, протокол, № 01 от 04.02.2026.

Содержание ФОС УД

1. Паспорт фонда оценочных средств
 - 1.1. Область применения фонда оценочных средств
 - 1.2. Организация текущего контроля успеваемости и промежуточной аттестации по итогам освоения междисциплинарного курса
2. Контроль и оценка достижения запланированных результатов обучения
 - 2.1. Перечень вопросов и заданий для текущего контроля знаний
 - 2.2. Перечень вопросов и заданий для промежуточной аттестации
 - 2.3. Критерии оценивания ПА

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

1.1 Область применения ФОС

Фонд оценочных средств (ФОС) представляет собой комплект материалов для проведения промежуточной аттестации и текущего контроля.

Результаты обучения - это усвоенные знания и освоенные умения по дисциплине в целях овладения предусмотренных стандартом общих и профессиональных компетенций, а также для оценки достижения обучающимися личностных результатов.

Фонд оценочных средств позволяет оценивать:

Код ОК, ПК	Уметь	Знать	Владеть навыками
ОК 01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части определять этапы решения задачи, составлять план действия, реализовывать составленный план, определять необходимые ресурсы выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы владеть актуальными методами работы в профессиональной и смежных сферах оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)	актуальный профессиональный и социальный контекст, в котором приходится работать и жить структура плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях основные источники информации и ресурсы для решения задач и/или проблем в профессиональном и/или социальном контексте методы работы в профессиональной и смежных сферах порядок оценки результатов решения задач профессиональной деятельности	
ОК 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	определять задачи для поиска информации, планировать процесс поиска, выбирать необходимые источники информации выделять наиболее значимое в перечне информации, структурировать получаемую информацию, оформлять результаты поиска оценивать практическую значимость результатов поиска применять средства информационных технологий для решения профессиональных задач использовать современное	номенклатура информационных источников, применяемых в профессиональной деятельности приемы структурирования информации формат оформления результатов поиска информации современные средства и устройства информатизации, порядок их применения программное обеспечение в профессиональной деятельности, в том числе цифровые средства психологические основы	

	программное обеспечение в профессиональной деятельности использовать различные цифровые средства для решения профессиональных задач	деятельности коллектива	
ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке проявлять толерантность в рабочем коллективе	правила оформления документов правила построения устных сообщений особенности социального и культурного контекста	
ОК 09 Пользоваться профессиональной документацией на государственном и иностранном языках	понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы участвовать в диалогах на знакомые общие и профессиональные темы строить простые высказывания о себе и о своей профессиональной деятельности кратко обосновывать и объяснять свои действия (текущие и планируемые) писать простые связные сообщения на знакомые или интересующие профессиональные темы	правила построения простых и сложных предложений на профессиональные темы основные общеупотребительные глаголы (бытовая и профессиональная лексика) лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности особенности произношения правила чтения текстов профессиональной направленности	
ПК.1.1 Проектировать базы данных	анализировать предметную область и выделять основные сущности; определять требования к базе данных; разрабатывать концептуальную, логическую и физическую модели баз данных; проектировать схему базы данных; работать с современными case-средствами проектирования баз данных;	основные положения теории баз данных, хранилищ данных, баз знаний; основные принципы структуризации и нормализации базы данных; основные принципы построения концептуальной, логической и физической модели данных; методы описания схем баз данных в	разработки концептуальной модели базы данных; разработки инфологической модели базы данных; разработки физической модели базы данных; разработки требований к базе данных нормализация структуры базы данных документирования схемы базы данных, включая диаграммы ER и

	<p>определять связи между таблицами;</p> <p>определять типы данных для полей таблиц;</p> <p>оформление документации на спроектированную базу данных</p> <p>разработки схемы базы данных, используя NoSQL модели данных, такие как документо-ориентированные, ключ-значение, колоночные и др.;</p>	<p>современных системах управления базами данных;</p> <p>структуры данных систем управления базами данных, основные понятия и принципы проектирования баз данных;</p> <p>структура реляционной базы данных;</p> <p>язык SQL и особенности его реализации в различных системах управления базами данных;</p> <p>оптимизация производительности баз данных</p> <p>принципы безопасности хранения данных</p>	<p>описания таблиц;</p> <p>документирования прав доступа и безопасности базы данных, включая учетные записи пользователей и их роли</p>
<p>ПК.1.2</p> <p>Разрабатывать объекты баз данных в соответствии с результатами анализа предметной области</p>	<p>разрабатывать объекты баз данных</p> <p>создавать таблицы, индексы, ограничения и другие объекты базы данных</p> <p>оптимизировать запросы к базе данных для повышения производительности</p> <p>разрабатывать хранимые процедуры и триггеры для баз данных;</p> <p>разрабатывать необходимые для различных групп пользователей представления</p>	<p>основы реляционной модели данных</p> <p>язык SQL и его основные команды</p> <p>принципы нормализации баз данных</p> <p>принципы работы с различными СУБД</p> <p>общий подход к организации представлений, таблиц, индексов и кластеров;</p> <p>методы организации целостности данных;</p> <p>способы контроля доступа к данным и управления привилегиями</p>	<p>работы с различными объектами базы данных</p>
<p>ПК.1.3</p> <p>Реализовывать базу данных в конкретной системе управления базами данных</p>	<p>разрабатывать объекты базы данных, такие как таблицы, индексы и связи между ними;</p> <p>программировать и создавать хранимые процедуры, функции и триггеры для обработки данных;</p> <p>управлять данными в базе данных, включая ввод, обновление и удаление данных;</p> <p>оптимизировать запросы и проводить мониторинг производительности базы данных;</p> <p>работать с NoSQL базами</p>	<p>основные принципы создания объектов базы данных;</p> <p>синтаксис и основные приемы работы с SQL;</p> <p>методы оптимизации запросов и повышения производительности базы данных;</p> <p>основные принципы управления данными и обслуживания базы данных;</p> <p>основные принципы работы NoSQL баз данных и их моделей данных;</p> <p>преимущества и</p>	<p>создания таблиц базы данных с определением структуры и типов данных для каждого атрибута;</p> <p>определения первичных и внешних ключей для установления связей между таблицами;</p> <p>создания индексов для оптимизации запросов и повышения производительности;</p> <p>разработки хранимых процедур, функций и триггеров для обработки данных и поддержки бизнес-логики;</p>

	<p>данных; использовать запросы для работы с данными в NoSQL базах данных; оптимизировать производительность NoSQL баз данных.</p>	<p>недостатки NoSQL технологий по сравнению с реляционными базами данных; методы оптимизации производительности NoSQL баз данных; основные принципы управления данными и обслуживания NoSQL баз данных.</p>	<p>ввода, обновления и удаления данных в соответствии с требованиями бизнес-процессов; оптимизации запросов для повышения производительности системы; создания баз данных на основе NoSQL технологий создания запросов для работы с данными в NoSQL базах данных; оптимизации производительности NoSQL баз данных, используя индексы и другие техники;</p>
<p>ПК.1.4 Администрировать базы данных</p>	<p>устанавливать и настраивать СУБД; создавать и удалять базы данных; создавать пользователей и назначать права доступа; оптимизировать запросы к базе данных; обеспечивать безопасность баз данных; создавать и настраивать базы данных в соответствии с требованиями бизнеса; управлять транзакциями и контролировать целостность данных; обеспечивать безопасность и управлять доступом к данным; создавать и восстанавливать резервные копии данных работать с индексами и оптимизировать производительность запросов нормализовать базы данных и проектировать эффективные структуры данных мониторить и анализировать производительность баз данных работать с нереляционными базами</p>	<p>архитектура СУБД основные принципы администрирования баз данных методы мониторинга и оптимизации работы баз данных принципы резервного копирования и восстановления баз данных методы защиты баз данных от внешних угроз особенности работы с различными СУБД Язык SQL (Structured Query Language) управление транзакциями и контроль целостности данных управление доступом и безопасностью баз данных резервное копирование и восстановление данных оптимизация производительности баз данных работа с индексами и оптимизация запросов мониторинг и анализ производительности принципы работы с реляционными базами данных принципы работы с нереляционными базами</p>	<p>установки и настройки СУБД; создания и удаления баз данных; восстановления баз данных; резервного копирования баз данных; создания пользователей и назначения прав доступа; оптимизации запросов к базе данных мониторинга и обслуживания NoSQL баз данных, включая резервное копирование и восстановление данных.</p>

	данных и выбирать наиболее подходящий тип базы данных для конкретной задачи	данных	
ПК.1.5 Защищать информацию в базе данных с использованием технологии защиты информации	разрабатывать и внедрять системы защиты баз данных от несанкционированного доступа разрабатывать и внедрять системы резервного копирования и восстановления баз данных проводить аудит безопасности баз данных устанавливать и настраивать механизмы аутентификации и авторизации пользователей создавать и управлять ролями и правами доступа к данным шифровать данные и обеспечивать их конфиденциальность контролировать целостность данных и обнаруживать изменения использовать механизмы аудита для отслеживания доступа к данным использовать механизмы мониторинга для обнаружения угроз безопасности создавать и управлять защищенными соединениями с базой данных использовать механизмы защиты от SQL-инъекций и других видов атак создавать и управлять бэкапами и резервными копиями данных обеспечивать безопасность базы данных при использовании облачных сервисов.	методы защиты баз данных от несанкционированного доступа методы создания и восстановления резервных копий баз данных особенности работы с различными типами СУБД методы проведения аудита безопасности баз данных принципы криптографии и методов шифрования данных стандарты и протоколы безопасности, таких как SSL/TLS, SSH, Kerberos и др. методы аутентификации и авторизации пользователей, включая использование паролей, сертификатов и биометрических данных методы контроля доступа, включая создание ролей и групп пользователей, управление правами доступа и аудит доступа к данным методы обнаружения и предотвращения атак, включая защиту от SQL-инъекций, DoS/DDoS-атак и других угроз безопасности методы мониторинга и анализа журналов событий для обнаружения угроз безопасности и анализа производительности базы данных методы создания и управления защищенными соединениями с базой данных, включая VPN-туннели и SSL-шифрование	использования стандартных методов защиты объектов базы данных; разработки и внедрения систем защиты баз данных от несанкционированного доступа разработки и внедрения систем резервного копирования и восстановления баз данных аудита безопасности баз данных

		<p>методы создания и управления бэкапами и резервными копиями данных, включая использование инкрементальных и дифференциальных бэкапов</p> <p>методы обеспечения безопасности базы данных при использовании облачных сервисов, включая защиту от утечки данных и управление доступом к облачным ресурсам</p> <p>законодательство и стандарты безопасности, такие как GDPR, HIPAA, PCI DSS и др.</p>	
--	--	---	--

1.2. Организация текущего контроля успеваемости и промежуточной аттестации по итогам освоения программы междисциплинарного курса

В период обучения по образовательной программе СПО осуществляется текущий контроль успеваемости студентов, промежуточная аттестация по учебным дисциплинам и МДК.

Текущий контроль осуществляется в пределах учебного времени, отведенного на учебную дисциплину, оценивается по пятибалльной шкале. Текущий контроль проводится с целью объективной оценки качества освоения программы дисциплины, а также стимулирования учебной деятельности студентов, подготовки к промежуточной аттестации и обеспечения максимальной эффективности учебного процесса. Для оценки качества подготовки используются различные формы и методы контроля. Текущий контроль дисциплины осуществляется в форме устного опроса; защиты практических заданий, реферата, творческих работ; выполнения контрольных и тестовых заданий; решения ситуационных задач и других форм контроля, предусмотренных программой дисциплины.

Промежуточная аттестация проводится в форме, предусмотренной планом учебного процесса: экзамена, дифференцированного зачета, зачета.

В период сложной санитарно-эпидемиологической обстановки или других ситуациях невозможности очного обучения и проведения аттестации студентов колледж реализует образовательные программы или их части с применением электронного обучения, дистанционных образовательных технологий в предусмотренных законодательством формах обучения или при их сочетании, при проведении учебных занятий, практик, текущего контроля успеваемости, промежуточной аттестации обучающихся.

Форма промежуточной аттестации по дисциплине МДК 01.01. Проектирование и разработка баз данных – экзамен.

2. КОНТРОЛЬ И ОЦЕНКА ОСВОЕНИЯ ПРОГРАММЫ МЕЖДИСЦИПЛИНАРНОГО КУРСА

2.1. Перечень вопросов и заданий для текущего контроля

В результате текущей аттестации по МДК 01.01. Проектирование и разработка баз данных осуществляется проверка сформированности умений и знаний, направленных на формирование соответствующих ФГОС СПО общих и профессиональных компетенций.

Перечень практических занятий:

1. Нормализация реляционной БД, освоение принципов проектирования БД
2. Преобразование реляционной БД в сущности и связи.
3. Проектирование реляционной БД. Нормализация таблиц.
4. Задание ключей. Создание основных объектов БД
5. Создание проекта БД. Создание БД. Редактирование и модификация таблиц
6. Редактирование, добавление и удаление записей в таблице. Применение логических условий к записям. Открытие, редактирование и пополнение табличного файла.
7. Создание ключевых полей. Задание индексов. Установление и удаление связей между таблицами.
8. Проведение сортировки и фильтрации данных. Поиск данных по одному и нескольким полям. Поиск данных в таблице.
9. Работа с переменными. Написание программного файла и работа с табличными файлами. Заполнение массива из табличного файла. Заполнение табличного файла из массива.
10. Добавление записей в табличный файл из двумерного массива. Работа с командами ввода-вывода. Использование функций для работы с массивами.
11. Создание меню различных видов. Модификация и управление меню.
12. Создание рабочих и системных окон. Добавление элементов управления рабочим окном
13. Создание файла проекта базы данных. Создание интерфейса входной формы. Использование исполняемого файла проекта БД, приемы создания и управления.
14. Создание формы. Управление внешним видом формы.
15. Задание значений и ограничений поля. Проверка введенного в поле значения. Отображение данных числового типа и типа дата
16. Создание и модификация таблиц БД. Выборка данных из БД. Модификация содержимого БД.
17. Обработка транзакций. Использование функций защиты для БД.

Промежуточный тест

1 Предметная область - это:

- 1) совокупность таблиц, состоящих из записей и полей; информации об индексах и связях; хранимых процедур;
- 2) совокупности таблиц, объединенных связями; экранных форм, отчетов, запросов
- 3) некоторая часть реально существующей системы, функционирующая как самостоятельная единица;
- 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
- 5) набор правил, обеспечивающих соответствие ключевых значений в связанных таблицах.

2 Система управления базой данных (СУБД) - это:

- 1) регулярная структура, состоящая из однотипных записей, разбитых на поля;
- 2) комплекс программных и языковых средств, необходимых для создания и модификации базы данных;
- 3) поименованная совокупность структурированных данных, относящихся к определенной предметной области;

- 4) служебная информация, содержащая упорядоченные сведения о ключевых значениях;
 - 5) программно-аппаратный комплекс, предназначенный для хранения и обработки информации какой-либо предметной области.
3. База данных - это:
- 1) комплекс программных и языковых средств, необходимых для добавления, модификации, удаления, поиска и отбора информации;
 - 2) совокупности таблиц, объединенных связями; экранных форм, отчетов, запросов;
 - 3) некоторая часть реально существующей системы, функционирующая как самостоятельная единица;
 - 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
 - 5) программно-аппаратный комплекс, предназначенный для хранения и обработки информации какой-либо предметной области.
4. Реляционная модель базы - это:
- 1) совокупность таблиц, состоящих из записей и полей; информации об индексах и связях; хранимых процедур;
 - 2) совокупности таблиц, объединенных связями; экранных форм, отчетов, запросов;
 - 3) некоторая часть реально существующей системы, функционирующая как самостоятельная единица;
 - 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
 - 5) набор правил программно-аппаратный комплекс, предназначенный для хранения и обработки информации какой-либо предметной области.
5. Таблица базы данных - это:
- 1) регулярная структура, состоящая из однотипных записей, разбитых на поля;
 - 2) комплекс программных и языковых средств, необходимых для создания и модификации базы данных;
 - 3) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
 - 4) служебная информация, содержащая упорядоченные сведения о ключевых значениях;
 - 5) функциональная зависимость между объектами.
6. Ключ таблицы базы данных - это:
- 1) поле или строковое выражение, образованное из значений нескольких полей, по которому можно определить значения других полей для одной или нескольких записей таблицы;
 - 2) поле или строковое выражение, образованное из значений нескольких полей, по которому можно однозначно идентифицировать строку в таблице;
 - 3) программный модуль, сохраняемый в базе данных для выполнения определенных операций с информацией базы;
 - 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
 - 5) набор правил, обеспечивающих связи между таблицами в базе данных.
7. Отношение в теории реляционных баз данных - это:
- 1) основной объект базы данных, состоящий из кортежей и имеющий определенный набор свойств – атрибутов;
 - 2) набор всех допустимых значений, которые может содержать атрибут;
 - 3) формальный метод анализа отношений на основе их первичного ключа и существующих функциональных зависимостей;
 - 4) функциональная зависимость между объектами;
 - 5) математические принципы, вытекающие из теории множеств и логики предикатов/

8. Связи между ключевыми значениями в реляционной модели бывают:
- 1) "один к одному", "один ко многим", "многие ко многим";
 - 2) только "один к одному";
 - 3) только "один ко многим";
 - 4) только "многие ко многим".
9. Сетевая модель данных состоит из:
- 1) набора экземпляров одного типа, образующих дерево с одним корневым объектом;
 - 2) набора записей и набора связей с любым числом других записей;
 - 3) совокупности таблиц со связями по ключевым значениям;
 - 4) многомерных таблиц, созданных с использованием объектно-ориентированных методов;
 - 5) множества баз данных, управляемых одной СУБД.
10. Реляционная модель данных состоит из:
- 1) набора экземпляров одного типа, образующих дерево с одним корневым объектом;
 - 2) набора записей и набора связей с любым числом других записей;
 - 3) совокупности таблиц со связями по ключевым значениям;
 - 4) многомерных таблиц, созданных с использованием объектно-ориентированных методов;
 - 5) множества баз данных, управляемых одной СУБД.
11. Иерархическая модель данных состоит из:
- 1) набора экземпляров одного типа, образующих дерево с одним корневым объектом;
 - 2) набора записей и набора связей с любым числом других записей;
 - 3) совокупности таблиц со связями по ключевым значениям;
 - 4) многомерных таблиц, созданных с использованием объектно-ориентированных методов;
 - 5) множества баз данных, управляемых одной СУБД.
12. Использование каких моделей данных наиболее эффективно в системах класса OLAP:
- 1) Реляционных;
 - 2) Объектно-ориентированных;
 - 3) Многомерных;
 - 4) Сетевых.
13. Предметно-ориентированный, интегрированный, неизменяемый и поддерживающий хронологию набор данных, предназначенный для обеспечения принятия управленческих решений, называется:
- 1) Банком данных;
 - 2) Информационным массивом;
 - 3) Хранилищем данных;
 - 4) Информационной системой.
14. Какое из перечисленных высказываний не является верным по отношению к объектно-ориентированным базам данных (ООБД):
- 1) При ссылке на объекты необходимо повторять пользовательские ключи;
 - 2) Все объекты ООБД идентифицируются одинаковым образом;
 - 3) Идентификаторы никогда не изменяются до тех пор, пока существуют объекты, которые они идентифицируют;
 - 4) Идентификаторы не характеризуются излишней сложностью.
15. В случае, если СУБД по отношению к базе данных выполняет не только те действия, которые явно указывает пользователь, но и дополнительные действия в соответствии с правилами, заложенными в саму СУБД, база данных называется:
- 1) Многомерной;
 - 2) Активной;
 - 3) Реляционной;
 - 4) Дедуктивной.

16. Оператор SQL, выполняющий проверку на диапазон значений:
 - 1) FROM...TO;
 - 2) BETWEEN...AND;
 - 3) FROM...AND;
 - 4) BETWEEN...TO.
17. Оператор IN в языке SQL выполняет:
 - 1) Проверку выражения на NULL;
 - 2) Проверку выражения на совпадение с любым из элементов списка;
 - 3) Проверку выражения на совпадение со всеми элементами списка;
 - 4) Логическую импликацию выражений.
18. Какая команда SQL осуществляет выбор пяти первых фамилий студентов, упорядоченных по учебным группам:
 - 1) SELECT Имя, Фамилия FROM Студент[Группа] LIMIT 5;
 - 2) SELECT Имя, Фамилия ORDER BY [Группа] FROM Студент LIMIT 5;
 - 3) SELECT Имя, Фамилия FROM Студент ORDER BY[Группа] LIMIT 5;
 - 4) SELECT Имя, Фамилия ORDER BY[Группа] WHERE Студент LIMIT 5.
19. Чем отличаются подходы, применяемые в реляционной алгебре и реляционном исчислении?
 - 1) Реляционная алгебра использует описательный подход, а реляционное исчисление предписывающий;
 - 2) Оба подхода описательные;
 - 3) Оба подхода предписывающие;
 - 4) Реляционная алгебра использует предписывающий подход, а реляционное исчисление описательный.
20. Какой из перечисленных тестов для СУБД измеряет насколько быстро СУБД может выполнять однотабличный запрос, ответ на который содержит определенную процентную долю строк таблицы:
 - 1) Полное сканирование;
 - 2) Обновление;
 - 3) Чтение с произвольной выборкой;
 - 4) Выборка.

Краткие методические указания.

Промежуточный тест проводится в электронной форме во время последнего в учебном периоде лабораторного занятия. Тест состоит из 20 тестовых заданий. На выполнение теста отводится 20 минут. Во время проведения теста использование литературы и других информационных ресурсов допускается только по предварительному согласованию с преподавателем.

Критерии оценки.

№	Баллы	Описание
5	19–20	Процент правильных ответов от 95% до 100%
4	16–18	Процент правильных ответов от 80 до 94%
3	13–15	Процент правильных ответов от 65 до 79%
2	9–12	Процент правильных ответов от 45 до 64%
1	0–8	Процент правильных ответов менее 45%

Аттестация проводится в форме практической работы для получения практического опыта и умения по разработке программ

- 1) Задание на создание базы данных – таблиц, индексов, установки реляционных

отношений.

2) Формирование логической схемы данных (СУБД MS Access).

3) Задание на разработку объектов базы данных в СУБД SQL SERVER. или СУБД MS Access.

4) Инструкция по выполнению работы.

Критерии оценивания:

«отлично» самостоятельно созданы таблицы, индексы, связи, остальные объекты базы данных в соответствии с заданием, написан и отлажен корректный программный код для выполнения поставленной задачи (СУБД MS Access) за отведенный на работу отрезок времени, выполнен ввод необходимых данных, получены корректные выходные данные

«хорошо» созданы объекты базы данных, разработан корректный программный код для выполнения поставленной задачи, выполнен ввод необходимых данных, получены корректные выходные данные. Но при выполнении задания учащемуся потребовалась незначительная помощь преподавателя

«удовлетворительно» созданы таблицы, индексы, основные объекты в соответствии с изучаемой темой разработаны не в соответствии с заданием (СУБД MS Access) или разработан программный код, не полностью реализующий поставленную задачу (СУБД MS Access), при выполнении задания учащемуся потребовалась значительная помощь преподавателя

«неудовлетворительно» В установленные сроки не разработаны объекты базы данных или программный код, реализующий поставленную задачу.

2.2. Перечень вопросов и заданий для промежуточной аттестации

Форма: экзамен

Список вопросов к экзамену

1. Основные понятия баз данных: объект, сущность, параметр, атрибут, триггер, правило, ограничение, хранимая процедура, ссылочная целостность, нормализация, первичный, альтернативный и внешний ключи.

2. Основные компоненты СУБД и их взаимодействие.

3. Типы моделей данных. Информационная модель предприятия. Информационная модель данных, её состав. Дидактический переход от одной модели данных к другой. Три типа логических моделей: иерархическая, сетевая и реляционная. Понятие логической и физической независимости данных.

4. Основные понятия реляционной модели. Понятие домена, отношения, атрибута и кортежа. Табличное представление отношений. Первичные и внешние ключи отношений, представление связей в реляционной базе данных. Целостность баз данных. Типы связей между отношениями. Понятие целостности.

5. Типы взаимосвязей в модели: «один-к-одному», «один-ко-многим» и «многие-ко-многим». Реляционный подход к построению модели данных. Преобразование взаимосвязи «многие-ко-многим» в таблицу перекрестных связей.

6. Нормальные формы схем отношений. Первая нормальная форма. Вторая нормальная форма. Третья нормальная форма. Л

7. Работа с таблицами и полями. Тип, размер, формат поля. Общие сведения о типах полей. Ввод данных, редактирование, выделение, копирование и вставка в таблице.

8. Схема данных и поддержка целостности данных. Ключи. Схема данных. Создание связи. Подтаблицы. Работа с подтаблицами. Проблемы целостности данных. Поддержка целостности данных. Параметры объединения. Поддержка целостности данных

9. Создание таблиц и ввод исходных данных

10. Работа со связанными таблицами, установка первичных и внешних ключей.

11. Сортировка и фильтрация отношений. Сортировка данных по возрастанию или убыванию.

12. Основные типы запросов. Создание запросов. Запрос на выборку с групповыми операциями.

13. Запрос на создание таблицы. Запрос на обновление. Запрос на добавление. Запрос на удаление. Перекрестный запрос.

14. Характеристика и стандарты языка SQL. Назначение и область применения. Стандарты SQL. Создание представлений в MySQL (views).

15. Характеристика и стандарты языка SQL. Назначение и область применения. Стандарты SQL. Создание пользовательских функций в MySQL (function).

16. Характеристика и стандарты языка SQL. Назначение и область применения. Стандарты SQL. Создание хранимых процедур в MySQL (stored procedure).

17. Характеристика и стандарты языка SQL. Назначение и область применения. Стандарты SQL. Создание триггеров (triggers) в MySQL.

Экзамен проходит: письменно по билетам, посредством написания кода.

В билете три практических задания.

- Место выполнения задания: учебный кабинет (компьютерный класс)
- Максимальное время выполнения задания: 60 мин.

В процессе подготовки к ответу обучающийся может воспользоваться:

- Рукописным конспектом лекций по данной дисциплине.

В процессе подготовки к ответу запрещено:

- Пользоваться интернетом на телефоне/компьютере.
- Заходить на сайт Портала.
- Разговаривать между собой.
- Передавать билет другим людям.

**Автономная некоммерческая организация
профессионального образования
«ПЕРМСКИЙ ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКИЙ КОЛЛЕДЖ»
(АНО ПО «ПГТК»)**

МДК 01.01. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БАЗ ДАННЫХ

Экзаменационный Билет № 1

1. Создать представление (views) для базы данных Логистическая компания. Первое представление отображает заказ одного покупателя по его уникальному идентификатору (id_c) и в отдельный вычисляемый столбец (sum_price) выводит стоимость каждого товара, умноженного на количество:

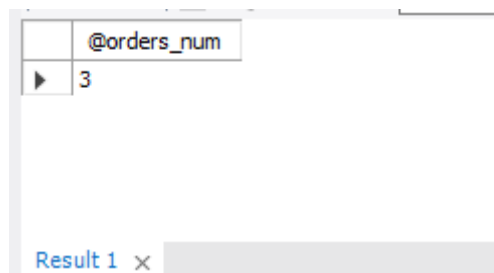
	id_d	date_d	id_c	fio_c	name_it	quantity_d	price	sum_price
▶	6	2023-12-13	3	Миронова Анастасия Игоревна	Крем для лица	2	199	398
	7	2023-12-13	3	Миронова Анастасия Игоревна	Крем для рук	1	150	150

2. Написать триггер для операции AFTER INSERT, который при попытке ввода даты из диапазона от 2023-12-31 до 2024-01-07 выдает сигнал об ошибке с сообщением «Такая дата доставки товара некорректна» (см. рис. 19-20):

14	2023-11-11	10	4	1	8	7
HULL	2024-01-02	2	1	1	1	3

Рисунок 19 – Попытка ввести некорректную дату в таблицу

3. Написать процедуру с входным и выходным параметрами, в результате которой по уникальному идентификатору покупателя customerID_d из таблицы delivery должно быть получено количество позиций, заказанных одним покупателем



Ст. преподаватель

Н.С. Могильникова

Экзаменационный Билет № 2

1. Создать представление для БД Логистическая компания, которое будет высчитывать количество отправленных товаров с определенным названием и определенным отправителем:

	senderID_d	name_sender	itemID_d	name_it	sum(quantity_d)
▶	1	КотМатрос Корм	8	Лекарство от паразитов	3

2. Создать триггер на событие DELETE, который будет удалять строку из таблицы customer и добавлять эти данные в дополнительную таблицу archive

	id_c	fio_c	passport_info	city_c	address_c
	1	Вежелинков Кирилл Кириллович	5653 382743	2	ул. Мира 15, кв. 287
	2	Антонов Антон Алексеевич	7464 838473	6	ул. Ленина 68б, 23
	3	Миронова Анастасия Игоревна	7347 384738	3	ул. Солдатова 28, кв. 47
	4	Лихачева Алена Андреевна	5452 563728	4	ул. Самолетная 3, кв 93
	5	Горин Геннадий Алексеевич	8273 547837	7	ул. Набережная 15, кв. 294
	6	Симонов Сергей Владимирович	7374 982838	6	ул. Техническая 3б, 37
	7	Петрова Екатерина Ивановна	6233 734636	6	ул. Гагарина 27, кв. 193
▶	20	Ангелина Кукурузова	8888 888888	1	ул. Кукурузная 15, кв. 115
▲	NULL	NULL	NULL	NULL	NULL

	id_c	fio_c	passport_info	city_c	address_c
	20	Ангелина Кукурузова	8888 888888	1	ул. Кукурузная 15, кв. 115
▶*	NULL	NULL	NULL	NULL	NULL

archive 1 x

3. Создать процедуру с использованием цикла, которая позволит получить товарные чеки для каждого покупателя по их идентификаторам id_c, например до 9 покупателя:

	id_d	date_d	id_c	fio_c	id_item	name_it	quantity_d	price	sum_price
▶	11	2023-12-01	6	Симонов Сергей Владимирович	3	Шампунь Бравл Старс	3	250	750
	12	2023-12-01	6	Симонов Сергей Владимирович	5	Корм для стерилизованных кошек	2	1200	2400
	13	2023-12-01	6	Симонов Сергей Владимирович	6	Корм для котят	1	550	550

Result 1 Result 2 Result 3 Result 4 Result 5 Result 6 Result 7 x Result 8

Экзаменационный Билет № 3

1. Создать представление (views) для БД Логистическая компания. Данное представление отображает заказ одного покупателя по его уникальному идентификатору (id_c) и в отдельный вычисляемый столбец (sum_price) выводит стоимость каждого товара, умноженного на количество:

	id_d	date_d	id_c	fio_c	name_it	quantity_d	price	sum_price
▶	6	2023-12-13	3	Миронова Анастасия Игоревна	Крем для лица	2	199	398
	7	2023-12-13	3	Миронова Анастасия Игоревна	Крем для рук	1	150	150

2. Создать триггер на событие UPDATE, который при изменении данных о получателе в таблице customer будет вносить старые данные в таблицу archive:

	id_c	fio_c	passport_info	city_c	address_c
	5	Горин Геннадий Алексеевич	8273 547837	7	ул. Набережная 15, кв. 294
	6	Симонов Сергей Владимирович	7374 982838	6	ул. Техническая 36, 37
	7	Петрова Екатерина Ивановна	6233 734636	6	ул. Гагарина 27, кв. 193
▶	23	Михайлов Максим Максимович	7333 747574	4	ул. Михаила 6, кв. 66
*	NULL	NULL	NULL	NULL	NULL

Изменение данных о получателе с id_c = 23

	id_c	fio_c	passport_info	city_c	address_c
▶	20	Ангелина Кукурузова	8888 888888	1	ул. Кукурузная 15, кв. 115
	21	sjdjdnfj	0000 000888	1	jsdnjewncfj
	23	Михайлов Максим Максимович	7333 666666	4	ул. Михаила 6, кв. 66
*	NULL	NULL	NULL	NULL	NULL

3. Создать процедуру с использованием цикла, благодаря которой можно получить товарные чеки для каждого покупателя по их идентификаторам id_c до покупателя с определенным идентификатором, например до 8 покупателя:

	id_d	date_d	id_c	fio_c	id_item	name_it	quantity_d	price	sum_price
▶	11	2023-12-01	6	Симонов Сергей Владимирович	3	Шампунь Бравл Старс	3	250	750
	12	2023-12-01	6	Симонов Сергей Владимирович	5	Корм для стерилизованных кошек	2	1200	2400
	13	2023-12-01	6	Симонов Сергей Владимирович	6	Корм для котят	1	550	550

Result 1
Result 2
Result 3
Result 4
Result 5
Result 6
Result 7 ×
 Result 8

Экзаменационный Билет № 4

1. Создать представление «Книги» для БД Библиотека института следующего вида:

#	ID	Title	FIO	Name	Year	Count
1	1	Книга1	Иванов Иван Иванович	Издатель1	2000	100
2	2	Книга2	Петров Петр Петрович	Издатель2	2001	200
3	3	Книга3	Сидоров Сидор Сидорович	Издатель3	2002	300
4	4	Книга4	Кузнецов Кузьма Кузьмич	Издатель4	2003	400
5	5	Книга5	Максимов Максим Максимович	Издатель5	2004	500
6	6	Книга6	Соколов Сокол Соколович	Издатель6	2005	600
7	7	Книга7	Тарасов Тарас Тарасович	Издатель7	2006	700
8	8	Книга8	Захаров Захар Захарович	Издатель8	2007	800
9	9	Книга9	Михайлов Михаил Михайлович	Издатель9	2008	900
10	10	Книга10	Алексеев Алексей Алексеевич	Издатель10	2009	1000

2. Создать триггер на событие удаления данных из таблицы, который будет удалять записи в таблице «Читательский билет» после удаления студентов:

#	ID	StudentID	EditionID	Date	ReturnDate	ReturnPeriod
1	1	1	1	2022-01-01	2022-01-15	15
2	2	2	2	2022-02-02	2022-02-16	15
3	3	3	3	2022-03-03	2022-03-17	15
4	4	4	4	2022-04-04	NULL	15
5	5	5	5	2022-05-05	NULL	15
6	6	6	6	2022-06-06	NULL	15
7	7	7	7	2022-07-07	2022-07-21	15
8	9	9	9	2022-09-09	2022-09-23	15
9	10	10	10	2022-10-10	2022-10-24	15
10	11	1	1	2022-01-01	NULL	15
*	NULL	NULL	NULL	NULL	NULL	NULL

3. Создать процедуру с использованием цикла, которая покажет всех студентов, у кого есть несданные книги.

Экзаменационный Билет № 5

1. Создать представление «Читательский билет» для БД Библиотека института следующего вида:

#	ID	FullName	StudentCard	BirthDate	Course	LibraryCardID	Date	ReturnDate	ReturnPeriod	BookEditionID	Title	FIO	Name	Year	Count
1	1	Студент1	1111	1990-01-01	1	1	2022-01-01	2022-01-15	15	1	Книга1	Иванов Иван Иванович	Издатель1	2000	100
2	2	Студент2	2222	1991-02-02	2	2	2022-02-02	2022-02-16	15	2	Книга2	Петров Петр Петрович	Издатель2	2001	200
3	3	Студент3	3333	1992-03-03	3	3	2022-03-03	2022-03-17	15	3	Книга3	Сидоров Сидор Сидорович	Издатель3	2002	300
4	4	Студент4	4444	1993-04-04	4	4	2022-04-04	2022-04-18	15	4	Книга4	Кузнецов Кузьма Кузьмич	Издатель4	2003	400
5	5	Студент5	5555	1994-05-05	3	5	2022-05-05	2022-05-20	15	5	Книга5	Максимов Максим Максимович	Издатель5	2004	500
6	6	Студент6	6666	1995-06-06	2	6	2022-06-06	2022-06-21	15	6	Книга6	Соколов Сокол Соколович	Издатель6	2005	600
7	7	Студент7	7777	1996-07-07	1	7	2022-07-07	2022-07-21	15	7	Книга7	Тарасов Тарас Тарасович	Издатель7	2006	700
8	8	Студент8	7777	1996-07-07	1	8	2022-08-08	2022-08-22	15	8	Книга8	Захаров Захар Захарович	Издатель8	2007	800
9	9	Студент9	7777	1996-07-07	1	9	2022-09-09	2022-09-23	15	9	Книга9	Михайлов Михаил Михайлович	Издатель9	2008	900
10	10	Студент10	7777	1996-07-07	1	10	2022-10-10	2022-10-24	15	10	Книга10	Алексеев Алексей Алексеевич	Издатель10	2009	1000

2. Создать триггер на событие обновления одной из таблиц, который будет увеличивать количество книг в библиотеке, если студент вернул книгу:

#	ID	BookID	PublisherID	Year	Count
1	1	1	1	2000	100
2	2	2	2	2001	200
3	3	3	3	2002	300
4	4	4	4	2003	400
5	5	5	5	2004	500
6	6	6	6	2005	600
7	7	7	7	2006	700
8	8	8	8	2007	800
9	9	9	9	2008	900
10	10	10	10	2009	1000

3. Создадим процедуру с использованием входного и выходного параметра, которая должна показать количество книг, выпущенных в 2022 году:

#	@count
1	10

Экзаменационный Билет № 6

1. Создать представление «Книги» для БД Библиотека института следующего вида:

#	ID	Title	FIO	Name	Year	Count
1	1	Книга1	Иванов Иван Иванович	Издатель1	2000	100
2	2	Книга2	Петров Петр Петрович	Издатель2	2001	200
3	3	Книга3	Сидоров Сидор Сидорович	Издатель3	2002	300
4	4	Книга4	Кузнецов Кузьма Кузьмич	Издатель4	2003	400
5	5	Книга5	Максимов Максим Максимович	Издатель5	2004	500
6	6	Книга6	Соколов Сокол Соколович	Издатель6	2005	600
7	7	Книга7	Тарасов Тарас Тарасович	Издатель7	2006	700
8	8	Книга8	Захаров Захар Захарович	Издатель8	2007	800
9	9	Книга9	Михайлов Михаил Михайлович	Издатель9	2008	900
10	10	Книга10	Алексеев Алексей Алексеевич	Издатель10	2009	1000

Рисунок 10 - Представление «Книги»

2. Создать триггер, который увеличивает или уменьшает количество книг в таблице «Издания» при создании новой записи в таблице «Читательский билет».

Если книгу вернули – то увеличивает, если книгу взяли – то уменьшает.

3. Создадим процедуру с использованием цикла, которая покажет студента, который дольше всех не возвращает книгу

:

#	maxDelayStudentID
1	1

Экзаменационный Билет № 7

1. Создать представление для БД Аэропорт, которое покажет номер пассажира и все билеты когда-либо им купленные.
2. Создать триггер на событие обновление данных из таблицы рейс, если в таблице рейс происходит обновление, то в статусе должно появиться слово – рейс задержан:

123 Id	ВремяПриб	ВремяОтпр	Опоздание	123 IdСамолёт	abc МестоОтправления	abc МестоПрибытия	123 Терминал	abc Статус
3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск1	Уфа1	2	Отменён

123 Id	ВремяПриб	ВремяОтпр	Опоздание	123 IdСамолёт	abc МестоОтправления	abc МестоПрибытия	123 Терминал	abc Статус
3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	Задержан
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск1	Уфа1	2	Отменён

3. Создать процедуру с использованием цикла, которая ищет номера пассажиров и номера их билетов за сегодняшнюю дату.

Экзаменационный Билет № 8

1. Создать представление «Билеты с датой покупки» для БД Аэропорт:

id	Пассажир	ДатаПокупки	Рейс	ПрошедшиеДни
2	2	2023-12-05 05:11:36	3	14
3	3	2023-12-01 00:00:00	4	18
4	1	2023-12-19 12:40:14	4	0
5	3	2023-12-19 19:32:17	3	0

2. Создать триггер на событие удаление данных одной из таблиц, а именно, при удалении самолета у рейса, в котором он задействован, должен появиться статус – рейс отменен:

3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	13	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	13	Новосибирск1	Уфа1	2	Отменён

id	ВремяПриб	ВремяОтпр	Опоздание	idСамолёт	МестоОтправления	МестоПрибытия	Терминал	Статус
3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск1	Уфа1	2	Отменён

3. Создать процедуру с использованием входного и выходного параметра, которая ищет id сотрудника по его имени:

123 НомерСотрудника

1

Экзаменационный Билет № 9

1. Создать представление «Общее время в пути» для БД Аэропорт следующего вида:

id	ВремяОтпр	ВремяПриб	Опоздание	ВремяВПути	ВремяВПутиСОпозданием
3	2023-12-19 10:16:00	2023-12-23 07:17:00	NULL	93:01:00	NULL
4	2023-12-01 04:13:00	2023-12-08 17:35:49	00:11:00	181:22:00	181:33:00
5	2023-12-20 17:36:00	2023-12-21 11:00:00	NULL	17:24:00	NULL
6	2023-12-20 17:36:00	2023-12-21 11:00:00	NULL	17:24:00	NULL

2. Создать триггер, который при добавлении нового билета изменит его дату на текущую:

123 id	123 Пассажир	🕒 ДатаПокупки	123 Рейс
2	2	2023-12-05 05:11:36	3
3	3	2023-12-01 00:00:00	4
4	1	2023-12-19 12:40:14	4
5	3	2023-12-19 19:32:17	3
[NULL]	2	[NULL]	3

123 id	123 Пассажир	🕒 ДатаПокупки	123 Рейс
2	2	2023-12-05 05:11:36	3
3	3	2023-12-01 00:00:00	4
4	1	2023-12-19 12:40:14	4
5	3	2023-12-19 19:32:17	3
6	2	2023-12-19 23:50:44	3

3. Создать процедуру с использованием цикла которая ищет самолёты определенной авиакомпании:

123 БортовойНомер	123 НормерСтойнки
2	2

Экзаменационный Билет № 10

1. Создать представление «Результаты экзаменов» для БД Музыкальная школа следующего вида:

TitleSpeciality	FIOStudent	TitleSubject	Mark
Фортепиано	Иванов Иван	Сольфеджио	2
Гитара	Васильев Василий	Сольфеджио	5
Скрипка	Михаил Круг	Сольфеджио	4
Баян	Саша Белый	Сольфеджио	4
Флейта	Сергей Зверев	Сольфеджио	3
Фортепиано	Иванов Иван	Игра на инструменте	5
Гитара	Васильев Василий	Игра на инструменте	3
Скрипка	Михаил Круг	Игра на инструменте	3
Баян	Саша Белый	Игра на инструменте	4
Флейта	Сергей Зверев	Игра на инструменте	2
Фортепиано	Иванов Иван	Муз.литература	2
Гитара	Васильев Василий	Муз.литература	4
Скрипка	Михаил Круг	Муз.литература	3
Баян	Саша Белый	Муз.литература	3
Флейта	Сергей Зверев	Муз.литература	3

2. Создать триггер, который удаляет записи в таблице «Результаты», соответствующие одиннадцатому коду ученика. То есть все записи, где указан номер студента 11, должны удалиться при обновлении данной таблицы.

3. Создать процедуру, которая выведет названия и оценки за экзамены по каждому из учеников школы (через цикл).

Экзаменационный Билет № 11

1. Создать представление «Преподаватели» для БД Музыкальная школа следующего вида:

idTeacher	FIOTeacher	TitleSpeciality	TitleSubject
1	Борис Моисеев	Флейта	Сольфеджио
2	Александр Вертинский	Скрипка	Хор
3	Василий Иванович	Баян	Игра на инструменте
4	Эрнесто Назарет	Фортепиано	ОКФ
5	Антон Чигур	Гитара	Муз.литература

2. Создать триггер, который при удалении записи из таблицы exams удаляет все строки в таблице results, где имелись оценки по данному экзамену. То есть, к примеру, был удален экзамен под номером 1 из таблицы exams и из таблицы results должны удалиться все строки, где есть оценки за 1 экзамен.

3. Создать процедуру (с использованием цикла), которая выведет по каждому студенту его средний балл за все экзамены.

Экзаменационный Билет № 12

1. Создать представление для БД Музыкальная школа, которое выводит оценки учеников по определенной специальности, например, Игра на инструменте, следующего вида:

TitleSpeciality	FIOStudent	TitleSubject	Mark
Фортепиано	Иванов Иван	Игра на инструменте	5
Гитара	Васильев Василий	Игра на инструменте	3
Скрипка	Михаил Круг	Игра на инструменте	3
Баян	Саша Белый	Игра на инструменте	4
Флейта	Сергей Зверев	Игра на инструменте	2

2. Создать триггер, который при удалении студента из таблицы Students удалит все записи об этом студенте также и из таблицы Results.

3. Написать процедуру (с использованием цикла), которая выведет оценки по каждой специальности в отдельной вкладке.

Экзаменационный Билет № 13

1. Создать представление (views) для базы данных Логистическая компания. Первое представление отображает заказ одного покупателя по его уникальному идентификатору (id_c) и в отдельный вычисляемый столбец (sum_price) выводит стоимость каждого товара, умноженного на количество:

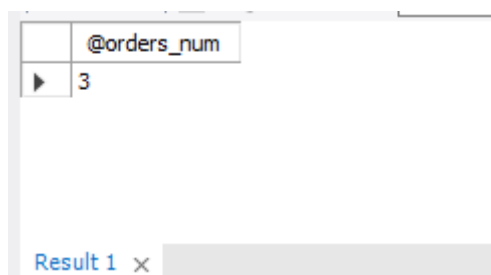
	id_d	date_d	id_c	fio_c	name_it	quantity_d	price	sum_price
▶	6	2023-12-13	3	Миронова Анастасия Игоревна	Крем для лица	2	199	398
	7	2023-12-13	3	Миронова Анастасия Игоревна	Крем для рук	1	150	150

2. Написать триггер для операции AFTER INSERT, который при попытке ввода даты из диапазона от 2023-12-31 до 2024-01-07 выдает сигнал об ошибке с сообщением «Такая дата доставки товара некорректна» (см. рис. 19-20):

14	2023-11-11	10	4	1	8	7
NULL	2024-01-02	2	1	1	1	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 19 – Попытка ввести некорректную дату в таблицу

3. Написать процедуру с входным и выходным параметрами, в результате которой по уникальному идентификатору покупателя customerID_d из таблицы delivery должно быть получено количество позиций, заказанных одним покупателем



Ст. преподаватель

Н.С. Могильникова

Экзаменационный Билет № 14

1. Создать представление для БД Логистическая компания, которое будет высчитывать количество отправленных товаров с определенным названием и определенным отправителем:

	senderID_d	name_sender	itemID_d	name_it	sum(quantity_d)
►	1	КотМатрос Корм	8	Лекарство от паразитов	3

2. Создать триггер на событие DELETE, который будет удалять строку из таблицы customer и добавлять эти данные в дополнительную таблицу archive

	id_c	fio_c	passport_info	city_c	address_c
	1	Вежелинков Кирилл Кириллович	5653 382743	2	ул. Мира 15, кв. 287
	2	Антонов Антон Алексеевич	7464 838473	6	ул. Ленина 68б, 23
	3	Миронова Анастасия Игоревна	7347 384738	3	ул. Солдатова 28, кв. 47
	4	Лихачева Алена Андреевна	5452 563728	4	ул. Самолетная 3, кв 93
	5	Горин Геннадий Алексеевич	8273 547837	7	ул. Набережная 15, кв. 294
	6	Симонов Сергей Владимирович	7374 982838	6	ул. Техническая 3б, 37
	7	Петрова Екатерина Ивановна	6233 734636	6	ул. Гагарина 27, кв. 193
►	20	Ангелина Кукурузова	8888 888888	1	ул. Кукурузная 15, кв. 115
▲	NULL	NULL	NULL	NULL	NULL

	id_c	fio_c	passport_info	city_c	address_c
►*	20	Ангелина Кукурузова	8888 888888	1	ул. Кукурузная 15, кв. 115
	NULL	NULL	NULL	NULL	NULL

archive 1 x

3. Создать процедуру с использованием цикла, которая позволит получить товарные чеки для каждого покупателя по их идентификаторам id_c, например до 9 покупателя:

	id_d	date_d	id_c	fio_c	id_item	name_it	quantity_d	price	sum_price
►	11	2023-12-01	6	Симонов Сергей Владимирович	3	Шампунь Бравл Старс	3	250	750
	12	2023-12-01	6	Симонов Сергей Владимирович	5	Корм для стерилизованных кошек	2	1200	2400
	13	2023-12-01	6	Симонов Сергей Владимирович	6	Корм для котят	1	550	550

Result 1 Result 2 Result 3 Result 4 Result 5 Result 6 Result 7 x Result 8

Экзаменационный Билет № 15

1. Создать представление (views) для БД Логистическая компания. Данное представление отображает заказ одного покупателя по его уникальному идентификатору (id_c) и в отдельный вычисляемый столбец (sum_price) выводит стоимость каждого товара, умноженного на количество:

	id_d	date_d	id_c	fio_c	name_it	quantity_d	price	sum_price
▶	6	2023-12-13	3	Миронова Анастасия Игоревна	Крем для лица	2	199	398
	7	2023-12-13	3	Миронова Анастасия Игоревна	Крем для рук	1	150	150

2. Создать триггер на событие UPDATE, который при изменении данных о получателе в таблице customer будет вносить старые данные в таблицу archive:

	id_c	fio_c	passport_info	city_c	address_c
	5	Горин Геннадий Алексеевич	8273 547837	7	ул. Набережная 15, кв. 294
	6	Симонов Сергей Владимирович	7374 982838	6	ул. Техническая 36, 37
	7	Петрова Екатерина Ивановна	6233 734636	6	ул. Гагарина 27, кв. 193
▶	23	Михайлов Максим Максимович	7333 747574	4	ул. Михаила 6, кв. 66
*	NULL	NULL	NULL	NULL	NULL

Изменение данных о получателе с id_c = 23

	id_c	fio_c	passport_info	city_c	address_c
▶	20	Ангелина Кукурузова	8888 888888	1	ул. Кукурузная 15, кв. 115
	21	sjdjdnfj	0000 000888	1	jsdnjewncfj
	23	Михайлов Максим Максимович	7333 666666	4	ул. Михаила 6, кв. 66
*	NULL	NULL	NULL	NULL	NULL

3. Создать процедуру с использованием цикла, благодаря которой можно получить товарные чеки для каждого покупателя по их идентификаторам id_c до покупателя с определенным идентификатором, например до 8 покупателя:

	id_d	date_d	id_c	fio_c	id_item	name_it	quantity_d	price	sum_price
▶	11	2023-12-01	6	Симонов Сергей Владимирович	3	Шампунь Бравл Старс	3	250	750
	12	2023-12-01	6	Симонов Сергей Владимирович	5	Корм для стерилизованных кошек	2	1200	2400
	13	2023-12-01	6	Симонов Сергей Владимирович	6	Корм для котят	1	550	550

Result 1
Result 2
Result 3
Result 4
Result 5
Result 6
Result 7
Result 8

Экзаменационный Билет № 16

1. Создать представление «Книги» для БД Библиотека института следующего вида:

#	ID	Title	FIO	Name	Year	Count
1	1	Книга1	Иванов Иван Иванович	Издатель1	2000	100
2	2	Книга2	Петров Петр Петрович	Издатель2	2001	200
3	3	Книга3	Сидоров Сидор Сидорович	Издатель3	2002	300
4	4	Книга4	Кузнецов Кузьма Кузьмич	Издатель4	2003	400
5	5	Книга5	Максимов Максим Максимович	Издатель5	2004	500
6	6	Книга6	Соколов Сокол Соколович	Издатель6	2005	600
7	7	Книга7	Тарасов Тарас Тарасович	Издатель7	2006	700
8	8	Книга8	Захаров Захар Захарович	Издатель8	2007	800
9	9	Книга9	Михайлов Михаил Михайлович	Издатель9	2008	900
10	10	Книга10	Алексеев Алексей Алексеевич	Издатель10	2009	1000

2. Создать триггер на событие удаления данных из таблицы, который будет удалять записи в таблице «Читательский билет» после удаления студентов:

#	ID	StudentID	EditionID	Date	ReturnDate	ReturnPeriod
1	1	1	1	2022-01-01	2022-01-15	15
2	2	2	2	2022-02-02	2022-02-16	15
3	3	3	3	2022-03-03	2022-03-17	15
4	4	4	4	2022-04-04	NULL	15
5	5	5	5	2022-05-05	NULL	15
6	6	6	6	2022-06-06	NULL	15
7	7	7	7	2022-07-07	2022-07-21	15
8	9	9	9	2022-09-09	2022-09-23	15
9	10	10	10	2022-10-10	2022-10-24	15
10	11	1	1	2022-01-01	NULL	15
*	NULL	NULL	NULL	NULL	NULL	NULL

3. Создать процедуру с использованием цикла, которая покажет всех студентов, у кого есть несданные книги.

Экзаменационный Билет № 17

1. Создать представление «Читательский билет» для БД Библиотека института следующего вида:

#	ID	FullName	StudentCard	BirthDate	Course	LibraryCardID	Date	ReturnDate	ReturnPeriod	BookEditionID	Title	FIO	Name	Year	Count
1	1	Студент1	1111	1990-01-01	1	1	2022-01-01	2022-01-15	15	1	Книга1	Иванов Иван Иванович	Издатель1	2000	100
2	2	Студент2	2222	1991-02-02	2	2	2022-02-02	2022-02-16	15	2	Книга2	Петров Петр Петрович	Издатель2	2001	200
3	3	Студент3	3333	1992-03-03	3	3	2022-03-03	2022-03-17	15	3	Книга3	Сидоров Сидор Сидорович	Издатель3	2002	300
4	4	Студент4	4444	1993-04-04	4	4	2022-04-04	2022-04-18	15	4	Книга4	Кузнецов Кузьма Кузьмич	Издатель4	2003	400
5	5	Студент5	5555	1994-05-05	3	5	2022-05-05	2022-05-20	15	5	Книга5	Максимов Максим Максимович	Издатель5	2004	500
6	6	Студент6	6666	1995-06-06	2	6	2022-06-06	2022-06-21	15	6	Книга6	Соколов Сокол Соколович	Издатель6	2005	600
7	7	Студент7	7777	1996-07-07	1	7	2022-07-07	2022-07-21	15	7	Книга7	Тарасов Тарас Тарасович	Издатель7	2006	700
8	8	Студент8	7777	1996-07-07	1	8	2022-08-08	2022-08-22	15	8	Книга8	Захаров Захар Захарович	Издатель8	2007	800
9	9	Студент9	7777	1996-07-07	1	9	2022-09-09	2022-09-23	15	9	Книга9	Михайлов Михаил Михайлович	Издатель9	2008	900
10	10	Студент10	7777	1996-07-07	1	10	2022-10-10	2022-10-24	15	10	Книга10	Алексеев Алексей Алексеевич	Издатель10	2009	1000

2. Создать триггер на событие обновления одной из таблиц, который будет увеличивать количество книг в библиотеке, если студент вернул книгу:

#	ID	BookID	PublisherID	Year	Count
1	1	1	1	2000	100
2	2	2	2	2001	200
3	3	3	3	2002	300
4	4	4	4	2003	400
5	5	5	5	2004	500
6	6	6	6	2005	600
7	7	7	7	2006	700
8	8	8	8	2007	800
9	9	9	9	2008	900
10	10	10	10	2009	1000

3. Создадим процедуру с использованием входного и выходного параметра, которая должна показать количество книг, выпущенных в 2022 году:

#	@count
1	10

Экзаменационный Билет № 18

1. Создать представление «Книги» для БД Библиотека института следующего вида:

#	ID	Title	FIO	Name	Year	Count
1	1	Книга1	Иванов Иван Иванович	Издатель1	2000	100
2	2	Книга2	Петров Петр Петрович	Издатель2	2001	200
3	3	Книга3	Сидоров Сидор Сидорович	Издатель3	2002	300
4	4	Книга4	Кузнецов Кузьма Кузьмич	Издатель4	2003	400
5	5	Книга5	Максимов Максим Максимович	Издатель5	2004	500
6	6	Книга6	Соколов Сокол Соколович	Издатель6	2005	600
7	7	Книга7	Тарасов Тарас Тарасович	Издатель7	2006	700
8	8	Книга8	Захаров Захар Захарович	Издатель8	2007	800
9	9	Книга9	Михайлов Михаил Михайлович	Издатель9	2008	900
10	10	Книга10	Алексеев Алексей Алексеевич	Издатель10	2009	1000

Рисунок 10 - Представление «Книги»

2. Создать триггер, который увеличивает или уменьшает количество книг в таблице «Издания» при создании новой записи в таблице «Читательский билет».

Если книгу вернули – то увеличивает, если книгу взяли – то уменьшает.

3. Создадим процедуру с использованием цикла, которая покажет студента, который дольше всех не возвращает книгу

:

#	maxDelayStudentID
1	1

Экзаменационный Билет № 19

1. Создать представление для БД Аэропорт, которое покажет номер пассажира и все билеты когда-либо им купленные.
2. Создать триггер на событие обновление данных из таблицы рейс, если в таблице рейс происходит обновление, то в статусе должно появиться слово – рейс задержан:

148 Id	ВремяПриб	ВремяОтпр	Опоздание	123 IdСамолёт	abc МестоОтправления	abc МестоПрибытия	123 Терминал	abc Статус
3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск1	Уфа1	2	Отменён

148 Id	ВремяПриб	ВремяОтпр	Опоздание	123 IdСамолёт	abc МестоОтправления	abc МестоПрибытия	123 Терминал	abc Статус
3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	Задержан
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск1	Уфа1	2	Отменён

3. Создать процедуру с использованием цикла, которая ищет номера пассажиров и номера их билетов за сегодняшнюю дату.

Экзаменационный Билет № 20

1. Создать представление «Билеты с датой покупки» для БД Аэропорт:

id	Пассажир	ДатаПокупки	Рейс	ПрошедшиеДни
2	2	2023-12-05 05:11:36	3	14
3	3	2023-12-01 00:00:00	4	18
4	1	2023-12-19 12:40:14	4	0
5	3	2023-12-19 19:32:17	3	0

2. Создать триггер на событие удаление данных одной из таблиц, а именно, при удалении самолета у рейса, в котором он задействован, должен появиться статус – рейс отменен:

3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	13	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	13	Новосибирск1	Уфа1	2	Отменён

id	ВремяПриб	ВремяОтпр	Опоздание	idСамолёт	МестоОтправления	МестоПрибытия	Терминал	Статус
3	2023-12-23 07:17:00	2023-12-19 10:16:00	[NULL]	2	Пермь	Москва	1	
4	2023-12-08 17:35:49	2023-12-01 04:13:00	00:11:00	2	Санкт-Петербург	Якутск	3	
5	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск	Уфа	2	Отменён
6	2023-12-21 11:00:00	2023-12-20 17:36:00	[NULL]	[NULL]	Новосибирск1	Уфа1	2	Отменён

3. Создать процедуру с использованием входного и выходного параметра, которая ищет id сотрудника по его имени:

123 НомерСотрудника

1

Экзаменационный Билет № 21

1. Создать представление «Общее время в пути» для БД Аэропорт следующего вида:

id	ВремяОтпр	ВремяПриб	Опоздание	ВремяВПути	ВремяВПутиСОпозданием
3	2023-12-19 10:16:00	2023-12-23 07:17:00	NULL	93:01:00	NULL
4	2023-12-01 04:13:00	2023-12-08 17:35:49	00:11:00	181:22:00	181:33:00
5	2023-12-20 17:36:00	2023-12-21 11:00:00	NULL	17:24:00	NULL
6	2023-12-20 17:36:00	2023-12-21 11:00:00	NULL	17:24:00	NULL

2. Создать триггер, который при добавлении нового билета изменит его дату на текущую:

123 id	123 Пассажир	🕒 ДатаПокупки	123 Рейс
2	2	2023-12-05 05:11:36	3
3	3	2023-12-01 00:00:00	4
4	1	2023-12-19 12:40:14	4
5	3	2023-12-19 19:32:17	3
[NULL]	2	[NULL]	3

123 id	123 Пассажир	🕒 ДатаПокупки	123 Рейс
2	2	2023-12-05 05:11:36	3
3	3	2023-12-01 00:00:00	4
4	1	2023-12-19 12:40:14	4
5	3	2023-12-19 19:32:17	3
6	2	2023-12-19 23:50:44	3

3. Создать процедуру с использованием цикла, которая ищет самолёты определенной авиакомпании:

123 БортовойНомер	123 НормерСтойнки
2	2

Экзаменационный Билет № 22

1. Создать представление «Результаты экзаменов» для БД Музыкальная школа следующего вида:

TitleSpeciality	FIOStudent	TitleSubject	Mark
Фортепиано	Иванов Иван	Сольфеджио	2
Гитара	Васильев Василий	Сольфеджио	5
Скрипка	Михаил Круг	Сольфеджио	4
Баян	Саша Белый	Сольфеджио	4
Флейта	Сергей Зверев	Сольфеджио	3
Фортепиано	Иванов Иван	Игра на инструменте	5
Гитара	Васильев Василий	Игра на инструменте	3
Скрипка	Михаил Круг	Игра на инструменте	3
Баян	Саша Белый	Игра на инструменте	4
Флейта	Сергей Зверев	Игра на инструменте	2
Фортепиано	Иванов Иван	Муз.литература	2
Гитара	Васильев Василий	Муз.литература	4
Скрипка	Михаил Круг	Муз.литература	3
Баян	Саша Белый	Муз.литература	3
Флейта	Сергей Зверев	Муз.литература	3

2. Создать триггер, который удаляет записи в таблице «Результаты», соответствующие одиннадцатому коду ученика. То есть все записи, где указан номер студента 11, должны удалиться при обновлении данной таблицы.

3. Создать процедуру, которая выведет названия и оценки за экзамены по каждому из учеников школы (через цикл).

Экзаменационный Билет № 23

1. Создать представление «Преподаватели» для БД Музыкальная школа следующего вида:

idTeacher	FIOTeacher	TitleSpeciality	TitleSubject
1	Борис Моисеев	Флейта	Сольфеджио
2	Александр Вертинский	Скрипка	Хор
3	Василий Иванович	Баян	Игра на инструменте
4	Эрнесто Назарет	Фортепиано	ОКФ
5	Антон Чигур	Гитара	Муз.литература

2. Создать триггер, который при удалении записи из таблицы exams удаляет все строки в таблице results, где имелись оценки по данному экзамену. То есть, к примеру, был удален экзамен под номером 1 из таблицы exams и из таблицы results должны удалиться все строки, где есть оценки за 1 экзамен.

3. Создать процедуру (с использованием цикла), которая выведет по каждому студенту его средний балл за все экзамены.

Экзаменационный Билет № 24

1. Создать представление для БД Музыкальная школа, которое выводит оценки учеников по определенной специальности, например, Игра на инструменте, следующего вида:

TitleSpeciality	FIOStudent	TitleSubject	Mark
Фортепиано	Иванов Иван	Игра на инструменте	5
Гитара	Васильев Василий	Игра на инструменте	3
Скрипка	Михаил Круг	Игра на инструменте	3
Баян	Саша Белый	Игра на инструменте	4
Флейта	Сергей Зверев	Игра на инструменте	2

2. Создать триггер, который при удалении студента из таблицы Students удалит все записи об этом студенте также и из таблицы Results.

3. Написать процедуру (с использованием цикла), которая выведет оценки по каждой специальности в отдельной вкладке.

Критерии оценки уровня сформированности компетенций по результатам проведения экзамена:

- оценка «отлично»: обучающийся выполнил три практических задания из билета в полной мере (все описанные функции, процедуры, триггеры работают корректно и запускаются без ошибок); дан развернутый ответ на поставленные вопросы по коду представлений, процедур и триггеров; найденные в ходе проверки недочеты и ошибки, были самостоятельно исправлены студентом; компетенции сформированы в полном объеме;

- оценка «хорошо»: обучающийся выполнил два задания из билета полностью, третье задание – частично (под частично подразумевается, что часть функций работает корректно, но, например, выдает не все необходимые столбцы); дан полный ответ на вопросы по коду одного из заданий; компетенции сформированы в достаточном объеме;

- оценка «удовлетворительно»: обучающийся выполнил только два задания из билета; частично смог ответить на вопросы по написанному коду; компетенции сформированы частично;

- оценки «неудовлетворительно»: обучающийся не выполнил ни одного задания из билета/либо созданная процедура/триггер/представление запускается с ошибками, не смог самостоятельно найти ошибки в написанном коде; компетенции не сформированы;