Чемпионат Свердловской области «Абилимпикс»

СОГЛАСОВАНО

Свердловское региональное отделение Общероссийской общественной организации инвалидов «Всероссийское общество глухих»

Председатель _____ Черемера Л.А.

«6» февраля 2025 г.

Свердловская областная Организация Общероссийской общественной организации инвалидов «Всероссийское ордена Трудового Красного Знамени общество слепых»

Председатель _____ Юдина М.А.

«6» февраля 2025 г.

УТВЕРЖДАЮ

Региональный центр развития движения «Абилимпикс»

Руководитель _____ Чешко С.Л.

«6» февраля 2025 г.

Работодатель

_____ / Васева И.В.

(подпись) *(ФИО)*

«___» _____ 2025 г.

Конкурсное задание по компетенции «Администрирование баз данных»

Екатеринбург, 2025 г.

РЕГИОНАЛЬНЫЙ ЧЕМПИОНАТ «АБИЛИМПИКС» 2025

Утверждено советом
по компетенции
Администрирование баз данных
Протокол № АдмБД-2412/1 от
20.12.2024
Председатель совета:
О.А. Москвин

КОНКУРСНОЕ ЗАДАНИЕ
по компетенции
Администрирование баз данных



Администрирование баз данных

Москва 2025

Оглавление

1. Описание компетенции 3
1.1. Актуальность компетенции 3
1.2 Профессии, по которым участники смогут трудоустроиться после освоения данной компетенции 4
1.3. Ссылка на образовательный и/или профессиональный стандарт (конкретные стандарты) 4
1.4. Требования к квалификации 5
2. Конкурсное задание 8
2.1. Краткое описание задания 8
2.2. Структура и подробное описание конкурсного задания 8
2.3. Последовательность выполнения задания 11
Школьники 11
Студенты 15
Специалисты 18
2.4. 30% изменение конкурсного задания 22
2.5. Особые указания 22
2.6 Критерии оценки выполнения задания 23
3.Перечень используемого оборудования, инструментов и расходных материалов 32
3.1 Перечень оборудования на 1-го участника (школьники, студенты, специалисты) 32
3.2. На 1-го эксперта (конкурсная площадка) 33
3.3. Общая инфраструктура конкурсной площадки 34
3.4. Комната участников 34
3.5. Дополнительные требования/комментарии 35
4. Минимальные требования к оснащению рабочих мест с учетом всех основных нозологий 36
5. Требования охраны труда и техники безопасности 40
5.1 Общие положения 40
5.2 Требования перед началом работы 40
5.3 Требования во время работы 41
5.4 Требования по окончании работы 41
5.5 Дополнительные рекомендации 41

1. Описание компетенции

В цифровую эпоху данные – это не просто информация, это стратегический ресурс, определяющий конкурентоспособность и успех в любой сфере. Компетенция «Администрирование баз данных» готовит специалистов, способных не только хранить и обрабатывать данные, но и извлекать из них ценную информацию, обеспечивая принятие обоснованных решений.

Участники осваивают полный цикл работы с базами данных, используя передовые образовательные практики:

- **Проектирование баз данных:** создание эффективных и масштабируемых схем, учитывающих требования производительности и современные подходы к организации данных (нормализация, оптимизация).
- **Администрирование СУБД:** настройка, обеспечение безопасности и высокой доступности данных, используя актуальные инструменты и технологии.
- **Интеграция данных:** автоматизация обмена данными с внешними системами через API – ключевой навык в условиях цифровой трансформации.
- **SQL-анализ:** написание SQL-запросов для анализа данных, подготовки отчетности и поддержки бизнес-логики.
- **Визуализация данных:** создание интерактивных дашбордов с помощью BI-инструментов для наглядного представления результатов и принятия обоснованных решений.

1.1. Актуальность компетенции

Сегодня данные – это не просто «новая нефть», это кровь цифровой экономики. От их качества, доступности и безопасности напрямую зависит функционирование бизнеса, государственных структур и общества в целом. Специалисты по базам данных играют ключевую роль в обеспечении бесперебойной работы информационных систем и извлечении ценной информации из данных.

Актуальность компетенции обусловлена:

- **Взрывным ростом объемов данных (Big Data).**
  Многим организациям требуется эффективное хранение и обработка огромных массивов информации. Недостаток квалифицированных кадров приводит к потерям данных и упущенным возможностям.
- **Необходимостью принимать решения на основе данных (Data-Driven Decision Making).**
  Бизнес полагается на анализ данных для оптимизации и развития. Отсутствие навыков анализа данных ограничивает карьерные перспективы выпускников.
- **Критической важностью безопасности данных.**
  Защита информации – приоритет в условиях киберугроз. Недостаточная подготовка в области безопасности данных создает риски для организаций и общества.
- **Цифровой трансформацией всех отраслей.**
  Современные организации используют множество разрозненных информационных систем. Интеграция данных между этими системами становится необходимостью для создания единого информационного пространства и автоматизации бизнес-процессов.

Навыки администрирования баз данных востребованы везде: от электронной коммерции и логистики до финансов, здравоохранения и государственного управления.

Компетенция «Администрирование баз данных» полностью отвечает современным требованиям рынка труда, предоставляя необходимые навыки для эффективного управления данными в условиях постоянно растущих объемов информации, необходимости принятия решений на основе данных и критической важности информационной безопасности. Освоение данной компетенции позволяет участникам стать востребованными специалистами, готовыми к решению сложных задач в различных отраслях экономики.

1.2 Профессии, по которым участники смогут трудоустроиться после освоения данной компетенции

Компетенция "Администрирование баз данных" обеспечивает подготовку специалистов для следующих профессий:

- **Школьники:**
  - Помощник администратора баз данных
  - Оператор баз данных.
- **Студенты:**
  - Администратор баз данных (DBA).
  - Инженер технической поддержки баз данных.
  - Специалист технической поддержки.
  - Разработчик баз данных / SQL-разработчик.
- **Специалисты:**
  - Администратор баз данных (DBA).
  - Разработчик баз данных / SQL-разработчик.
  - Младший архитектор баз данных.
  - Разработчик интеграционных решений.

1.3. Ссылка на образовательный и/или профессиональный стандарт (конкретные стандарты)

Школьники	**Студенты**	**Специалисты**
ФГОС среднего (полного) общего образования	ФГОС СПО по специальностям: 09.02.05 Прикладная информатика по отраслям, 09.02.07 Информационные системы и программирование	ФГОС СПО по специальности: 09.02.07 Информационные системы и программирование
ФГОС СПО по специальности: 09.02.07 Информационные системы и программирование	ФГОС ВО по направлениям подготовки: 09.03.01 Информатика и вычислительная техника (бакалавр), 09.03.03 Прикладная информатика (бакалавр), 38.03.05 Бизнес-информатика (бакалавриат).	ФГОС ВО по направлениям подготовки: 09.03.01 Информатика и вычислительная техника (бакалавр), 09.03.03 Прикладная информатика (бакалавр), 38.03.05 Бизнес-информатика (бакалавриат)

	Проф стандарты: 06.011 Администратор баз данных, 06.015 Специалист по информационным системам, 06.039 Специалист по информационно-коммуникационным технологиям.	Проф стандарты: 06.011 Администратор баз данных, 06.015 Специалист по ин формационным системам, 06.039 Специалист по информационно-коммуникационным технологиям.

1.4. Требования к квалификации

Школьники	**Студенты**	**Специалисты**
Знать: ● базовые понятия теории баз данных; ● что такое реляционная база данных и ее основные преимущества; ● типы данных в базах данных (например, INT, VARCHAR, DATE) и их ключевые свойства; ● основы проектирования структуры и оптимизации базы данных (таблицы, связи, ключи); ● основы безопасности баз данных (защита личных данных, принцип наименьших привилегий) ● назначение и применение основных встроенных функций SQL; ● базовые принципы интеграции и работы с инструментами визуализации данных. **Уметь:** ● создавать схемы баз данных и проектировать таблицы в MySQL	**Знать:** ● основы теории баз данных, включая концепции реляционных баз данных; ● типы данных (например, INT, VARCHAR, DATE) и их ключевые свойства; ● принципы проектирования структуры баз данных (таблицы, связи, индексы, первичные и внешние ключи); ● основы нормализации базы данных; ● методы импорта данных из файлов (CSV, Excel) и API; ● основы безопасности баз данных (защита личных данных, принцип наименьших привилегий); ● основы использования SQL для анализа данных (агрегация, фильтрация, группировка); ● принципы интеграции данных через API; ● Методы визуализации данных с помощью BI-инструментов.	**Знать:** ● принципы проектирования сложных реляционных баз данных, включая создание внешних ключей, индексов и связей между таблицами; ● концепции нормализации базы данных и оптимизации запросов; ● методы работы с внешними источниками данных, включая интеграцию через API; ● принципы безопасности баз данных, включая разграничение прав доступа и защиту личных данных; ● основы написания триггеров и автоматизации процессов в базе данных; ● подходы к работе с большими наборами данных (300+ строк) и их анализу с использованием SQL; ● методы построения визуализаций для анализа данных и

Workbench; ● организовывать связи между таблицами, устанавливать первичные и внешние ключи; ● импортировать данные в базы данных из файлов (например, CSV); ● настраивать права доступа для пользователей базы данных; ● писать простые SQL-запросы для анализа данных, включая сортировку, фильтрацию и группировку; ● создавать интеграции, конструировать отчетные диаграммы и визуализации в системах BI-аналитики на основе подключенных данных.	**Уметь:** ● разрабатывать структуры баз данных с использованием MySQL Workbench; ● создавать схемы баз данных и проектировать таблицы с внешними ключами и индексами; ● импортировать данные из внешних источников (CSV, Excel) в таблицы базы данных; ● использовать API для получения данных и интеграции их в базу данных; ● настраивать права доступа для пользователей базы данных в соответствии с принципами безопасности; ● разрабатывать SQL-запросы для анализа данных (сложные условия, сортировка, группировка, агрегация); ● создавать и использовать дашборды для анализа данных с применением BI-инструментов.	бизнес-отчетов; ● возможности и принципы использования BI-инструментов для создания интерактивных дашбордов. **Уметь:** ● проектировать сложные структуры баз данных с использованием MySQL Workbench, включая настройку индексов и внешних ключей; ● реализовывать триггеры для автоматизации рутинных процессов; ● автоматизировать процесс получения данных через API, используя Python; ● настраивать права доступа для пользователей с различными уровнями доступа, в соответствии с принципом минимально необходимых привилегий; ● писать сложные SQL-запросы с использованием вложенных подзапросов, оконных функций и агрегатов для анализа данных; ● выявлять закономерности в данных (например, влияние погодных условий на логистику) и представлять их в виде аналитических отчетов; ● разрабатывать интерактивные дашборды в BI-инструментах, включая фильтры, карты и таблицы; ● Обеспечивать обновляемость дашборда при

		изменении данных в базе.

2. Конкурсное задание

2.1. Краткое описание задания

Дед Мороз готовится к Новому году и получает тысячи писем от детей со всей страны. Ему необходимо не только ответить на все письма, но и доставить подарки вовремя. Для этого участники помогают настроить систему управления письмами, подарками и доставкой, с учетом погодных условий для оптимизации маршрутов.

Задача участников — спроектировать базу данных, заполнить ее информацией из различных источников, интегрировать данные через API, написать SQL-запросы для анализа и визуализировать результаты с помощью дашборда.

Задание разделено на 4 модуля:

1. Проектирование базы данных и заполнение базы данных данными.
2. Настройка прав доступа для пользователей.
3. Написание SQL-запросов для анализа.
4. Визуализация данных в BI.

Каждая категория участников (**школьники**, **студенты**, **специалисты**) выполняет задание соответствующей сложности.

Программное обеспечение для реализации задания:

- MySQL (Workbench, Server, Community, Connector/J) — для проектирования базы данных, написания запросов, настройки прав доступа и интеграции с другими системами.
- Python (Anaconda и Jupyter Notebook) — для автоматизации интеграции данных через API (для студентов и специалистов).
- BI-инструмент — для построения дашборда и визуализации результатов.
- Редактор для работы с файлами TXT.
- Assistapp — ПО инклюзивной доступности.

2.2. Структура и подробное описание конкурсного задания

Конкурсное задание для школьников

Наименование модуля	Время проведения модуля	Полученный результат
Модуль А: Создание структуры базы данных и заполнение данными	50 минут	Создана схема DedMoroz_[номер участника]. Разработана структура базы данных с учетом принципов нормализации данных и созданы таблицы. Создана ER-диаграмма. Таблицы заполнены данными.

Модуль Б: Настройка прав доступа для пользователей	40 минут	Созданы пользователи с разными уровнями доступа (DedMoroz, Snegurochka, Elf_logist). Права проверены через SHOW GRANTS.
Модуль В: Написание SQL-запросов для анализа	40 минут	Созданы запросы для анализа данных: популярность городов по количеству писем, подарки с нулевым запасом, отмененные доставки, самый популярный подарок.
Модуль Г: Построение дашборда с визуализацией данных	50 минут	Данные интегрированы в BI-систему. Построен дашборд: индикаторы остатков по складам, популярность подарков (столбчатая диаграмма), содержит названия и пояснения к визуализациям. Дашборд опубликован.
Общее время выполнения конкурсного задания: 180 минут (3 часа)		

Конкурсное задание для студентов

Наименование модуля	**Время проведения модуля**	**Полученный результат**
Модуль А: Создание структуры базы данных и заполнение данными	60 минут	Создана схема DedMoroz_[номер участника]. Разработана структура базы данных с учетом принципов нормализации данных и созданы таблицы. Создана ER-диаграмма. Таблицы заполнены данными. Интегрированы исторические данные о погоде через Open-Meteo API. Предоставлены SQL-запросы и Python-код для воспроизведения результата.
Модуль Б: Настройка прав доступа для пользователей	40 минут	Созданы пользователи с разными уровнями доступа (DedMoroz, Snegurochka, Elf_logist). Права проверены через SHOW GRANTS. Предоставлены SQL-запросы для воспроизведения результата.
Модуль В: Написание SQL-запросов для анализа	60 минут	Созданы запросы для анализа данных: города с тяжелыми погодными условиями, текущие маршруты доставки, проверка складских остатков, популярность подарков по возрастам. Запросы поддерживают сортировку и агрегированные результаты. Предоставлены SQL-запросы для воспроизведения результата.

Модуль Г4: Построение дашборда с визуализацией данных	60 минут	Данные интегрированы в BI-систему. Разработан дашборд с визуализациями: активность городов по письмам, складские остатки, статус доставок. Дашборд опубликован, предоставлены ссылки.
Общее время выполнения конкурсного задания: 220 минут (3 часа 40 минут)		

Конкурсное задание для специалистов

Наименование модуля	**Время проведения модуля**	**Полученный результат**
Модуль А: Создание структуры базы данных и заполнение данными	50 минут	Создана схема DedMoroz_[номер участника]. Разработана структура базы данных с учетом принципов нормализации данных и созданы таблицы. Создана ER-диаграмма. Таблицы заполнены данными. Интегрированы исторические данные о погоде через Open-Meteo API. Предоставлены SQL-запросы и Python-код для воспроизведения результата.
Модуль Б: Настройка прав доступа для пользователей	45 минут	Созданы пользователи с разными уровнями доступа (DedMoroz, Snegurochka, Elf_logist). Права проверены через SHOW GRANTS. Создан триггер на таблицу “Письма”. Участниками предоставлены сохраненные SQL-запросы, позволяющие достигнуть описанный результат.
Модуль В: Написание SQL-запросов для анализа	50 минут	Созданы запросы для анализа: количества дней с неблагоприятными погодными условиями, текущих активных доставок, определения точек срыва доставок,определения достаточности запасов подарков. Запросы поддерживают сортировку и агрегированные результаты. Предоставлены SQL-запросы для воспроизведения результата.
Модуль Г: Построение дашборда с визуализацией данных	75 минут	Данные интегрированы в BI-систему. Разработан дашборд с визуализациями: активность писем, критические запасы подарков, проблемные маршруты доставок, влияние температуры на доставку, региональная нагрузка. Дашборд опубликован, предоставлены ссылки.
Общее время выполнения конкурсного задания: 220 минут (3 часа 40 минут)		

Участник может самостоятельно распределить время на выполнение каждого модуля.

2.3. Последовательность выполнения задания

Постановка задачи

Цель задания — разработка базы данных для управления данными, связанными с подготовкой Деда Мороза к Новому году. База данных должна отражать всю необходимую информацию о письмах, подарках, маршрутах доставки и погодных условиях, чтобы оптимизировать процесс логистики.

Участникам предстоит:

- Создать структуру базы данных.
- Заполнить таблицы данными из предоставленных файлов и внешних источников (через API).
- Настроить права доступа для пользователей, соответствующих легенде.
- Разработать SQL-запросы для анализа данных.
- Визуализировать ключевые данные в BI-инструменте.

При создании базы участники чемпионата должны продемонстрировать свои знания и навыки обработки информационных массивов и первичного анализа данных.

Школьники

Новый год – время чудес и подарков! Но за всеми праздничными хлопотами стоит огромная работа. Представьте себя в роли участника IT-команды Деда Мороза! Ежегодно он получает тысячи писем с пожеланиями от детей со всей страны.

Дед Мороз готовится получает тысячи писем от детей со всей страны. В письме дети пишут Деду Морозу о том, какой подарок они хотели бы от него получить на Новый год. Каждый ребенок может попросить у Деда Мороза только один подарок. Дед Мороз читает каждое письмо и записывает имя и возраст автора письма, город и дату отправления письма, категорию и название подарка. Подарки хранятся на складах. У Деда Мороза существует несколько складов в разных городах, но в одном городе может быть только один склад. О каждом подарке известно к какой категории он относится, на каком складе хранится и в каком количестве. Деду Морозу необходимо доставить подарки вовремя и в соответствии с полученными письмами. Команда Деда Мороза серьезно относится к доставке подарков и старается сделать это максимально точно и безупречно. Но иногда могут возникать непредвиденные обстоятельства. Чтобы держать ситуацию под контролем у каждого подарка есть статус, например, "сборка", "в пути", "доставлен", “отменен”, а для каждой доставки записывается дата отправки и дата доставки, если подарок уже доставлен.

Чтобы ни один ребенок не остался без подарка, необходимо четко организовать учет писем, складских запасов и доставки. В этом году Дед Мороз поручил вам создать современную систему управления данными, которая поможет ему справиться с этой важной задачей.

Цель задания:

Разработать полноценную систему управления данными для Деда Мороза, включающую проектирование базы данных, настройку прав доступа, анализ данных с помощью SQL и визуализацию результатов в виде интерактивного дашборда.

Для выполнения и сохранения результатов заданий необходимо определить свой [номер участника] в соответствии с номером по жеребьевке в папке Абилимпикс_школьники, которая находится на Рабочем столе участника. Все создаваемые файлы сохранять в папке Конкурсное_задание[номер участника] на рабочем столе.

Модуль А: Создание структуры базы данных и заполнение данными

Основа любого чуда – это четкая организация. Прежде чем подарки отправятся в путь, необходимо создать надежное хранилище для всей информации. Представьте, что каждое письмо, каждый подарок и каждая доставка – это важные элементы большой головоломки. Ваша задача – создать структуру, которая позволит собрать все элементы вместе.)

Но как организовать хранение данных о письмах, подарках, складах и доставках, чтобы избежать путаницы и обеспечить быстрый доступ к информации?

Задачи:

1. **Открыть MySQL Workbench.**
   Запустите программу MySQL Workbench – инструмент, с помощью которого вы будете создавать и управлять базой данных.
2. **Создать схему базы данных.**
   Создайте новую схему (это как отдельная папка для вашей базы данных) с уникальным названием, DedMoroz_[номер участника]. Установите кодировку utf8mb4 для корректной работы с русским текстом.
3. **Спроектировать схему базы данных.**
   - **Проанализировать предоставленные данные и спроектировать схему базы данных.**
     Организаторы предоставляют файлы с данными, которые нужно будет загрузить в базу. Внимательно изучите структуру этих файлов (какие данные в них содержатся, какие поля есть в каждой таблице). Это поможет понять, какая информация должна храниться в базе данных и как ее лучше организовать.

     На основе этого анализа, используя принципы нормализации (избегайте повторения данных!) и ER-моделирование (схема, показывающая связи между разными типами информации), спроектируйте структуру базы данных.

     Продумайте, какие таблицы понадобятся (например, таблица для писем, таблица для подарков, таблица для складов) и какие поля будут в каждой таблице (например, в таблице "Письма" могут быть поля "Имя ребенка", "Город", "Желаемый подарок").

     Учитывайте, что каждая таблица должна хранить информацию только об одном типе сущности (например, только о письмах, только о подарках). Каждая таблица должна иметь первичный ключ (уникальный идентификатор каждой записи), а для связи между таблицами используйте внешние ключи. Выбирайте подходящие типы данных для каждого поля (например, INT для чисел, VARCHAR для текста, DATE для дат).

- **Создать таблицы в MySQL.**
  Создайте таблицы в MySQL в соответствии со спроектированной схемой. Убедитесь, что для всех таблиц корректно определены PRIMARY KEY и, где необходимо, AUTO_INCREMENT, внешние ключи и типы данных.
  Для проверки решения поочередно запустите код DESCRIBE [Наименование_таблицы] для каждой созданной таблицы.

4. **Создать ER-диаграмму.**
   На основе созданной структуры базы данных постройте ER-диаграмму в MySQL Workbench. Эта диаграмма наглядно покажет, как связаны между собой различные таблицы и данные в базе. Укажите на диаграмме типы существующих в схеме связей (один-к-одному, один-ко-многим).
5. **Импортировать данные из предоставленных файлов (CSV).**
   Импортируйте данные из файлов, предоставленных организаторами, в созданные таблицы. После импорта проверьте, все ли данные загрузились корректно и нет ли ошибок.

Модуль Б: Настройка прав доступа для пользователей

В команде Деда Мороза у каждого своя роль и свои обязанности. Важно обеспечить, чтобы каждый специалист имел/а доступ только к необходимой информации.
Представьте, что у каждого члена команды есть свой ключ от определенной части хранилища данных. Снегурочка отвечает за письма и подарки, Эльф Логист – за доставку, а Дед Мороз – за все.

Как разграничить доступ к данным, чтобы предотвратить случайные ошибки и обеспечить безопасность информации?

Задачи:

1. **Создать пользователей MySQL.**
   Создайте три учетные записи пользователей в базе данных: DedMoroz (полный доступ), Snegurochka (доступ к таблицам, содержащим информацию о письмах и подарках), Elf_logist (доступ к таблицам, содержащим информацию о доставках). Придумайте надежные пароли для каждого пользователя.
2. **Настроить права доступа для каждого пользователя.**
   Настройте права доступа так, чтобы каждый пользователь мог выполнять только те действия, которые ему необходимы для работы. Например, Снегурочка должна иметь возможность добавлять, изменять и просматривать информацию о письмах и подарках, но не должна иметь возможности изменять информацию о доставке. Эльфу нужен доступ к таблицам, содержащим информацию о доставках, но он не должен изменять таблицы, связанные с подарками.
3. **Проверить корректность настроенных прав.**
   Убедитесь, что права доступа настроены правильно, используя специальные команды SQL (SHOW GRANTS). Сохраните результаты проверки.

Модуль В: Написание SQL-запросов для анализа

Чтобы принимать верные решения, Деду Морозу нужна точная информация. С помощью SQL вы сможете извлечь из базы данных ценные знания.

Представьте, что SQL – это волшебный язык, который позволяет задавать вопросы базе данных и получать на них точные ответы.

Как получить ответы на важные вопросы о письмах, подарках и доставке, используя SQL?

Задачи:

1. **Написать SQL-запрос, который покажет количество писем по городам.**
   Дед Мороз хочет знать, из каких городов приходит больше всего писем. Напишите запрос, который посчитает количество писем из каждого города и отсортирует результаты по убыванию количества.
2. **Написать SQL-запрос, который отобразит подарки, количество которых на складе равно нулю.**
   Снегурочке нужно знать, какие подарки закончились на складах. Напишите запрос, который выведет список таких подарков.
3. **Написать SQL-запрос для поиска детей, чьи доставки отменены.**
   Дед Мороз хочет знать, кому из детей не достанется подарок из-за отмены доставки. Напишите запрос, который выведет имена этих детей.
4. **Написать SQL-запрос, чтобы определить самый популярный подарок.**
   Дед Мороз хочет знать, какой подарок просят чаще всего. Напишите запрос, который найдет самый популярный подарок среди всех писем.

Модуль Г: Построение дашборда с визуализацией данных

Визуализация данных помогает быстро увидеть картину целиком и принять взвешенные решения. Создайте дашборд, который станет главным инструментом для Деда Мороза. Представьте, что дашборд – это панель управления, на которой отображаются все самые важные показатели работы команды Деда Мороза.

Как представить данные в наглядном и удобном для анализа виде?

Задачи:

1. **Подключить базу данных к BI-инструменту (или использовать CSV-файлы).**
   Войдите в личный кабинет BI-инструмента (данные для входа предоставляются организаторами) и подключите его к созданной базе данных MySQL. Если подключение невозможно, загрузите данные из CSV-файлов, полученных в первом модуле.
2. **Создать интерактивные визуализации (чарты).**
   - **Наличие подарков на складах:** создайте *индикаторы* для каждого из трех складов (A, B и C), показывающие *суммарное количество подарков* на каждом складе.
   - **О каких подарках мечтают дети:** создайте *столбчатую диаграмму*, где по горизонтальной оси будут *названия подарков*, а высота столбцов будет соответствовать *количеству упоминаний каждого подарка в письмах*.
3. **Скомпоновать визуализации на дашборде и опубликовать.**
   - Создайте дашборд “Дашборд Деда Мороза”.
   - Разместите созданные визуализации на дашборде в логичном порядке, добавьте заголовки и подписи, чтобы дашборд был понятным и удобным в использовании.
   - Опубликуйте дашборд, включив публичный доступ.

Студенты

Новый год – время чудес и подарков! Дед Мороз готовится к Новому году и получает тысячи писем от детей со всей страны. В письме дети пишут Деду Морозу о том, какой подарок они хотели бы от него получить на Новый год.

Каждый ребенок может попросить у Деда Мороза только один подарок. Дед Мороз читает каждое письмо и записывает имя и возраст автора письма, город и дату отправления письма, категорию и название подарка. Подарки хранятся на складах. У Деда Мороза существует несколько складов в разных городах, но в одном городе может быть только один склад.

О каждом подарке известно к какой категории он относится, на каком складе хранится и в каком количестве. Деду Морозу необходимо доставить подарки вовремя и в соответствии с полученными письмами. Команда Деда Мороза серьезно относится к доставке подарков и старается сделать это максимально точно и безупречно. Но иногда могут возникать непредвиденные обстоятельства, такие как плохая погода и низкие температуры. Чтобы держать ситуацию под контролем у каждого подарка есть статус, например, "сборка", "в пути", "доставлен", “отменен”; для каждой доставки записывается дата отправки и дата доставки, если подарок уже доставлен; а для каждого города доставки фиксируется минимальная температура.

За всеми праздничными хлопотами стоит огромная работа. Чтобы ни один ребенок не остался без подарка, необходимо четко организовать учет писем, складских запасов и доставки, а в этом году еще и учитывать погодные условия (минимальные температуры для каждого города доставки).

Представьте себя в роли IT-команды Деда Мороза! В этом году Дед Мороз поручил вам создать современную систему управления данными, которая поможет ему справиться с этой важной задачей. Необходимо настроить систему управления письмами, подарками и доставкой, с учетом погодных условий для оптимизации маршрутов.

Цель задания:

Разработать базу данных для Деда Мороза, чтобы эффективно управлять письмами, подарками, доставкой и учитывать погодные условия. Участники продемонстрируют навыки проектирования, оптимизации, работы с API, анализа данных и визуализации.

Для выполнения и сохранения результатов заданий необходимо определить свой [номер участника] в соответствии с номером по жеребьевке в папке Абилимпикс_студенты, которая находится на Рабочем столе участника. Все создаваемые файлы сохранять в папке Конкурсное_задание[номер участника] на рабочем столе.

Модуль А: Создание структуры базы данных и заполнение данными

Дед Мороз получает письма от детей и распределяет подарки по складам, чтобы организовать их доставку. Но в этом году задач стало больше: нужно учитывать не только письма, но и погодные условия, чтобы избежать сбоев в маршрутах. Помогите создать эффективную базу данных, чтобы все успеть к Новому году!

Как создать эффективную и оптимизированную базу данных, учитывающую письма, подарки, доставку и погодные условия, чтобы избежать проблем с доставкой?

Задачи:

1. **Открыть MySQL Workbench.**
   Запустите программу MySQL Workbench – инструмент, с помощью которого вы будете создавать и управлять базой данных.
2. **Создать схему базы данных.**
   Создайте новую схему с уникальным названием, например, DedMoroz_[номер участника]. Установите кодировку utf8mb4 для корректной работы с русским текстом.
3. **Спроектировать схему базы данных.**
   - **Проанализировать предоставленные данные и спроектировать схему базы данных.**
     Организаторы предоставляют файлы с данными. Изучите их структуру. Спроектируйте нормализованную структуру базы данных.

     Продумайте, какие таблицы понадобятся для хранения информации о городах, подарках, письмах, доставках и погоде, и какие поля будут в каждой таблице. Учитывайте, что каждая таблица должна хранить информацию только об одном типе сущности. Каждая таблица должна иметь первичный ключ, а для связи между таблицами используйте внешние ключи. Выбирайте подходящие типы данных для каждого поля. Оптимизируйте размеры полей.

   - **Создать таблицы в MySQL.**
     Создайте таблицы в MySQL в соответствии со спроектированной схемой, используя SQL-запросы CREATE TABLE. Убедитесь, что для всех таблиц корректно определены PRIMARY KEY и, где необходимо, AUTO_INCREMENT, внешние ключи и типы данных. Для проверки решения поочередно запустите код DESCRIBE [Наименование_таблицы] для каждой созданной таблицы.
   - **Создать индексы.**
     Создайте индексы для оптимизации запросов: по идентификаторам городов в таблице писем, идентификаторам складов в таблице складских запасов, идентификаторам писем в таблице доставок, а также комбинированный индекс для таблицы погоды по идентификатору города и дате (или смысловые аналоги, в соответствие с выбранной структурой данных).

     Сохраните SQL-запросы в файле module1_create_tables.sql.
4. **Создать ER-диаграмму.**
   На основе созданной структуры базы данных постройте ER-диаграмму в MySQL Workbench. Эта диаграмма наглядно покажет, как связаны между собой различные таблицы и данные в базе. Укажите на диаграмме типы связей. Сохраните диаграмму как файл .mwb и экспортируйте в PNG.
5. **Импортировать данные из предоставленных файлов (CSV).**
   Импортируйте данные из файлов, предоставленных организаторами, в созданные таблицы. После импорта проверьте, все ли данные загрузились корректно и нет ли ошибок.
6. **Интегрировать данные о погоде через API.**
   Создайте Python-скрипт в Jupyter Notebook. Используйте Open-Meteo API для получения данных о погоде (минимальная температура) за декабрь 2024 года для каждого города в системе. Сохраните код в файл module1_weather_api.py.

Модуль Б: Настройка прав доступа для пользователей

В команде Деда Мороза у каждого своя роль и свои обязанности. Важно обеспечить, чтобы каждый специалист имел/а доступ только к необходимой информации. Представьте, что у каждого члена команды есть свой ключ от определенной части хранилища данных. Снегурочка отвечает за письма и подарки, Эльф Логист – за доставку, а Дед Мороз – за все.

Как разграничить доступ к данным, чтобы предотвратить случайные ошибки и обеспечить безопасность информации?

Задачи:

1. **Создать пользователей MySQL.**
   Создайте три учетные записи пользователей в базе данных: DedMoroz (полный доступ), Snegurochka (доступ к таблицам, содержащим информацию о письмах и подарках), Elf_logist (доступ к таблицам, содержащим информацию о доставках). Придумайте надежные пароли для каждого пользователя.
2. **Настроить права доступа для каждого пользователя.**
   Настройте права доступа так, чтобы каждый пользователь мог выполнять только те действия, которые ему необходимы для работы. Например, Снегурочка должна иметь возможность добавлять, изменять и просматривать информацию о письмах и подарках, но не должна иметь возможности изменять информацию о доставке. Эльфу нужен доступ к таблицам, содержащим информацию о доставках, но он не должен изменять таблицы, связанные с подарками.
3. **Проверить корректность настроенных прав.**
   Убедитесь, что права доступа настроены правильно, используя специальные команды SQL (SHOW GRANTS). Сохраните результаты проверки.

Сохраните SQL-код в файл module2_user_access.sql.

Модуль B: Написание SQL-запросов для анализа

С приближением Нового года Дед Мороз и его команда сталкиваются с управленческими и логистическими проблемами.

Как с помощью SQL решить практические задачи и оптимизировать доставку?

Задачи:

1. **Города с самыми тяжелыми погодными условиями.**
   Напишите запрос, который покажет список городов и количество дней, когда температура опускалась ниже значения - 10. Сортируйте по количеству таких дней.
2. **Проблемы с доставкой.**
   Напишите запрос, который покажет список городов, куда сейчас направляются подарки, и количество таких подарков (только для доставок в пути). Сортируйте по количеству.
3. **Хватит ли подарков на складе для всех доставок?**
   Напишите запрос, который покажет разницу между доступным количеством подарков и требуемым для доставок в статусе "сборка". Сортируйте от меньшего к большему.
4. **Самый активный возраст детей и их предпочтения.**
   Напишите запрос, который покажет возраст детей, общее количество писем от этого возраста, самую популярную категорию подарков для этого возраста и самый популярный подарок в этой категории. Сортируйте по количеству писем.

Сохраните SQL-запросы в файл module3_analysis_queries.sql.

Модуль Г: Построение дашборда с визуализацией данных

Дед Мороз использует дашборды для принятия управленческих решений.

Как создать интуитивно понятный дашборд, отвечающий на ключевые вопросы и помогающий оперативно находить проблемные зоны?

Задачи:

1. **Подключить базу данных к BI-инструменту (или использовать CSV-файлы).**
   Войдите в личный кабинет BI-инструмента (данные для входа предоставляются организаторами) и подключите его к созданной базе данных MySQL. Если подключение невозможно, загрузите данные из CSV-файлов, полученных в первом модуле.
2. **Создать интерактивные визуализации (чарты).**
   - **Анализ писем (Активность городов).**
     Создайте визуализацию, показывающую, из каких городов отправляют больше всего писем.
   - **Остатки подарков.**
     Создайте визуализацию, показывающую, какие подарки заканчиваются на складах (менее 20 в наличии).
   - **Статус доставки.**
     Создайте визуализацию, показывающую распределение статусов доставок по городам.
3. **Скомпоновать визуализации на дашборде и опубликовать.**
   - Создайте дашборд “Дашборд Деда Мороза”.
   - Разместите созданные визуализации на дашборде в логичном порядке, добавьте заголовки и подписи, чтобы дашборд был понятным и удобным в использовании.
   - Опубликуйте дашборд, включив публичный доступ.

Специалисты

Новый год – время чудес и подарков! Дед Мороз готовится к Новому году и получает тысячи писем от детей со всей страны. В письме дети пишут Деду Морозу о том, какой подарок они хотели бы от него получить на Новый год.

Каждый ребенок может попросить у Деда Мороза только один подарок. Дед Мороз читает каждое письмо и записывает имя и возраст автора письма, город и дату отправления письма, категорию и название подарка. Подарки хранятся на складах. У Деда Мороза существует несколько складов в разных городах, но в одном городе может быть только один склад.

О каждом подарке известно к какой категории он относится, на каком складе хранится и в каком количестве. Деду Морозу необходимо доставить подарки вовремя и в соответствии с полученными письмами. Команда Деда Мороза серьезно относится к доставке подарков и старается сделать это максимально точно и безупречно. Но иногда могут возникать непредвиденные обстоятельства, такие как плохая погода и низкие температуры. Чтобы держать ситуацию под контролем у каждого подарка есть статус, например, "сборка", "в пути", "доставлен", “отменен”; для каждой доставки записывается дата отправки и дата доставки, если подарок уже доставлен; а для каждого города доставки фиксируется минимальная температура.

За всеми праздничными хлопотами стоит огромная работа. Чтобы ни один ребенок не остался без подарка, необходимо четко организовать учет писем, складских запасов и доставки, а в этом году еще и учитывать погодные условия (минимальные температуры для каждого города доставки).

В этом году Дед Мороз поручил вам создать современную систему управления данными, которая поможет ему справиться с этой важной задачей. Необходимо настроить систему управления письмами, подарками и доставкой, с учетом погодных условий для оптимизации маршрутов.

Цель задания:

Разработать базу данных для Деда Мороза, чтобы эффективно управлять письмами, подарками, доставкой и учитывать погодные условия. Участники продемонстрируют навыки проектирования, оптимизации, работы с API, анализа данных и визуализации.

Для выполнения и сохранения результатов заданий необходимо определить свой [номер участника] в соответствии с номером по жеребьевке в папке Абилимпикс_специалисты, которая находится на Рабочем столе участника. Все создаваемые файлы сохранять в папке Конкурсное_задание[номер участника] на рабочем столе.

Модуль А: Создание структуры базы данных и заполнение данными

Дед Мороз получает письма от детей и распределяет подарки по складам, чтобы организовать их доставку. Но в этом году задач стало больше: нужно учитывать не только письма, но и погодные условия, чтобы избежать сбоев в маршрутах. Помогите создать эффективную базу данных, чтобы все успеть к Новому году!

Как создать эффективную и оптимизированную базу данных, учитывающую письма, подарки, доставку и погодные условия, чтобы избежать проблем с доставкой?

Задачи:

1. **Открыть MySQL Workbench.**
   Запустите программу MySQL Workbench – инструмент, с помощью которого вы будете создавать и управлять базой данных.
2. **Создать схему базы данных.**
   Создайте новую схему с уникальным названием, DedMoroz_[номер участника]. Установите кодировку utf8mb4 для корректной работы с русским текстом.
3. **Спроектировать схему базы данных.**
   - **Проанализировать предоставленные данные и спроектировать схему базы данных.**
     Организаторы предоставляют файлы с данными. Изучите их структуру. Спроектируйте нормализованную структуру базы данных.

     Продумайте, какие таблицы понадобятся для хранения информации о городах, подарках, письмах, доставках и погоде, и какие поля будут в каждой таблице. Учитывайте, что каждая таблица должна хранить информацию только об одном типе сущности. Каждая таблица должна иметь первичный ключ, а для связи между таблицами используйте внешние ключи. Выбирайте подходящие типы данных для каждого поля. Оптимизируйте размеры полей.

- **Создать таблицы в MySQL.**
  Создайте таблицы в MySQL в соответствии со спроектированной схемой, используя SQL-запросы CREATE TABLE. Убедитесь, что для всех таблиц корректно определены PRIMARY KEY и, где необходимо, AUTO_INCREMENT, внешние ключи и типы данных. Для проверки решения поочередно запустите код DESCRIBE [Наименование_таблицы] для каждой созданной таблицы.
- **Создать индексы.**
  Создайте индексы для оптимизации запросов: по идентификаторам городов в таблице писем, идентификаторам складов в таблице складских запасов, идентификаторам писем в таблице доставок, а также, комбинированный индекс для таблицы погоды по идентификатору города и дате (или смысловые аналоги, в соответствие с выбранной структурой данных).

  Сохраните SQL-запросы в файле module1_create_tables.sql.

4. **Создать ER-диаграмму.**
   На основе созданной структуры базы данных постройте ER-диаграмму в MySQL Workbench. Эта диаграмма наглядно покажет, как связаны между собой различные таблицы и данные в базе. Укажите на диаграмме типы связей. Сохраните диаграмму как файл .mwb и экспортируйте в PNG.
5. **Импортировать данные из предоставленных файлов (CSV).**
   Импортируйте данные из файлов, предоставленных организаторами, в созданные таблицы. После импорта проверьте, все ли данные загрузились корректно и нет ли ошибок.
6. **Интегрировать данные о погоде через API.**
   Создайте Python-скрипт в Jupyter Notebook. Используйте Open-Meteo API для получения данных о погоде (минимальная температура) за декабрь 2024 года для каждого города в системе. Сохраните код в файл module1_weather_api.py.

Модуль Б: Настройка прав доступа для пользователей

В команде Деда Мороза у каждого своя роль и свои обязанности. Важно обеспечить, чтобы каждый специалист имел/а доступ только к необходимой информации.

Представьте, что у каждого члена команды есть свой ключ от определенной части хранилища данных. Снегурочка отвечает за письма и подарки, Эльф Логист – за доставку, а Дед Мороз – за все.

Как разграничить доступ к данным, чтобы предотвратить случайные ошибки и обеспечить безопасность информации?

Задачи:

1. **Создать пользователей MySQL.**
   Создайте три учетные записи пользователей в базе данных: DedMoroz (полный доступ), Snegurochka (доступ к таблицам, содержащим информацию о письмах и подарках), Elf_logist (доступ к таблицам, содержащим информацию о доставках). Придумайте надежные пароли для каждого пользователя.
2. **Настроить права доступа для каждого пользователя.**
   Настройте права доступа так, чтобы каждый пользователь мог выполнять только те действия, которые ему необходимы для работы. Например, Снегурочка должна иметь возможность добавлять, изменять и просматривать информацию о письмах и подарках, но не должна иметь

возможности изменять информацию о доставке. Эльфу нужен доступ к таблицам, содержащим информацию о доставках, но он не должен изменять таблицы, связанные с подарками.

3. **Проверить корректность настроенных прав.**
   Убедитесь, что права доступа настроены правильно, используя специальные команды SQL (SHOW GRANTS). Сохраните результаты проверки.
4. **Создать триггер.**
   Создайте триггер, который будет автоматически контролировать возраст ребенка при добавлении нового письма. Триггер должен срабатывать перед добавлением новой записи и генерировать ошибку, если возраст ребенка больше 16 лет включительно.

Сохраните SQL-код в файл module2_user_access.sql.

Модуль В: Написание SQL-запросов для анализа

Дед Мороз анализирует данные, чтобы понять, какие города отправляют больше всего писем, хватает ли подарков на складах и как погодные условия и время в пути влияют на доставку.

Как с помощью SQL решить аналитические задачи и оптимизировать доставку?

Задачи:

1. **Города с самыми тяжелыми погодными условиями.**
   Напишите запрос, который покажет список городов и количество дней, когда минимальная температура опускалась ниже значения -10. Сортируйте по количеству таких дней.
2. **Города с высокой загруженностью доставок.**
   Напишите запрос, который покажет города с наибольшей логистической нагрузкой, включая текущие маршруты и успешно завершенные доставки.
3. **Определение точек срыва доставки.**
   Напишите запрос, который покажет информацию о доставках, длительность которых превышала 3 дня с момента отправки до момента доставки, и свяжет эту информацию с погодными условиями. Сортируйте по длительности маршрута.
4. **Хватит ли подарков на складе для всех доставок?**
   Напишите запрос, который покажет разницу между доступным количеством подарков и требуемым для доставок в статусе "сборка". Сортируйте от меньшего к большему.

Сохраните SQL-запросы в файл module3_analysis_queries.sql.

Модуль Г: Построение дашборда с визуализацией данных

Дед Мороз использует дашборды для принятия управленческих решений.

Как создать интуитивно понятный дашборд, отвечающий на ключевые вопросы и помогающий оперативно находить проблемные зоны?

Задачи:

1. **Подключить базу данных к BI-инструменту (или использовать CSV-файлы):** Подключите BI-инструмент к базе данных или загрузите данные из CSV-файлов и настройте связи между таблицами.
2. **Создать интерактивные визуализации (чарты):**

- **Динамика активности писем.**
  Создайте визуализацию, показывающую, как меняется количество писем на протяжении времени.
- **Запасы критических подарков.**
  Создайте визуализацию, показывающую подарки, запасы которых находятся на критическом уровне (менее 20 штук), и их возможную нехватку.
- **Проблемные маршруты доставки.**
  Создайте визуализацию, показывающую маршруты с задержками, длительность которых с момента отправки до момента доставки превысила 3 дня.
- **Влияние температуры на доставку.**
  Создайте визуализацию, показывающую влияние низких температур на доставку.
- **Региональная нагрузка.**
  Создайте визуализацию, показывающую общую логистическую нагрузку по регионам.

3. **Скомпоновать визуализации на дашборде и опубликовать.**
   - Создайте дашборд “Дашборд Деда Мороза”.
   - Разместите созданные визуализации на дашборде в логичном порядке, добавьте заголовки и подписи, чтобы дашборд был понятным и удобным в использовании.
   - Опубликуйте дашборд, включив публичный доступ.

2.4. 30% изменение конкурсного задания

С целью проверки адаптивности участников к изменяющимся условиям, конкурсное задание может быть изменено не более чем на 30%.

Изменения могут включать уточнение формулировок и условий отдельных задач. При этом, изменения не повлияют на общую сложность и суть конкурсного задания, а также на основные навыки, необходимые для его выполнения (проектирование, администрирование, SQL-анализ, визуализация).

2.5. Особые указания

1. Для визуализации данных будет использоваться BI-сервис, название которого будет сообщено дополнительно и опубликовано перед проведением конкурсного задания. Параметры подключения к сервису (логин, пароль и другие необходимые параметры) также будут предоставлены перед проведением конкурсного задания.
2. Во время выполнения задания участникам разрешено использовать только те интернет-ресурсы, которые непосредственно необходимы для выполнения конкурсных задач. К разрешенным ресурсам относятся ресурсы, посещение которых явно указано в конкурсном задании, например, официальная документация API, доступ к BI-сервису, онлайн-справочники по синтаксису. Использование поисковых систем для поиска готовых решений, общение в мессенджерах, посещение социальных сетей и других ресурсов, не связанных с выполнением задания, запрещено.
   Любые попытки нечестного поведения, использование посторонней помощи, доступ к запрещенным ресурсам или любое другое нарушение правил, могут повлечь за собой санкции (например, уменьшение времени, отведенного на выполнение задания или снятие баллов).
3. Для обеспечения честности и объективности оценки, будет использоваться протокол безопасности, который может включать использование брандмауэра для ограничения доступа в интернет, запись экрана участников, наблюдение за рабочими местами или другие меры. Детали протокола безопасности будут сообщены перед проведением конкурсного задания.

4. Организаторы оставляют за собой право разработать и предоставить участникам дополнительные инструкции по выполнению задания в единой для всех форме.
5. Все указанные детали, включая название BI-сервиса, параметры подключения, детали протокола безопасности и любые другие важные сведения, будут опубликованы или направлены всем участникам одновременно.

2.6 Критерии оценки выполнения задания

Критерии оценки выполнения задания для школьников

Наименование модуля	**Задание**	**Максимальный балл**
Модуль А: Создание структуры базы данных и заполнение данными	1.Запуск программы MySQL Workbench. 2. Создание схемы базы данных. 2. Разработка структуры базы данных с учетом принципов нормализации данных. Создание таблиц с первичными и внешними ключами. Проверка корректности выполнения через команду DESCRIBE. 3. Создание ER-диаграммы отображающей корректность созданной структуры базы данных. 4. Заполнение таблиц данными из предоставленных CSV файлов.	30
Модуль Б: Настройка прав доступа для пользователей	1. Создание трех пользователей с различными уровнями доступа: DedMoroz, Snegurochka и Elf_logist. 2. Настройка прав доступа для каждого пользователя. 3. Проверка прав доступа через SHOW GRANTS	20
Модуль В: Написание SQL-запросов для анализа	1. Создание запроса для определения количества писем, отправленных из каждого города. 2. Создание запроса для нахождения подарков с нулевым запасом на складе. 3. Создание запроса для поиска детей, чьи подарки были отменены. 4. Создание запроса для нахождения самого популярного подарка среди детей.	20
Модуль Г: Построение дашборда с визуализацией данных	1. Подключение данных к BI-системе. 2. Создание визуализаций (индикаторов и графика) для отображения остатков подарков по складам и популярности подарков. 3. Создание дашборда с размещением всех созданных визуализаций, названий и пояснений. Публикация дашборда в публичный доступ.	30
	ИТОГО:	**100**

Подробные критерии выполнения задания для школьников

Задание	Наименование критерия	Максимальные баллы
Модуль 1. Создание структуры базы данных и заполнение данными		
1.1	Запущена программа MySQL Workbench.	2
1.2	Создана схема базы данных с именем DedMoroz_[номер участника].	4
1.3	Созданы таблицы с корректными полями и типами данных. Таблицы логически соответствуют файлам, предоставляющим данные.	5
1.3	Корректность выполнения созданных таблиц проверена через команду DESCRIBE.	4
1.3	База данных спроектирована корректно, нормализована (как минимум до 3НФ), корректно определены первичные и внешние ключи, обеспечивающие ссылочную целостность и отсутствие избыточности данных. Выбраны оптимальные типы данных для атрибутов.	5
1.4	Создана ER-диаграмма, которая полностью соответствует нормализованной структуре базы данных. Отображены все таблицы и все связи между ними с правильной кардинальностью. Диаграмма экспортирована в формат .png.	5
1.5	Данные из всех предоставленных CSV-файлов корректно импортированы во все соответствующие таблицы. Проверено количеством записей и корректностью данных в таблицах.	5
	Итого:	**30**
Модуль Б. Настройка прав доступа для пользователей		
2.1	Созданы три пользователя: DedMoroz, Snegurochka, Elf_logist	6
2.2	Настроен полный доступ для DedMoroz ко всем таблицам	3
2.2	Настроен полный доступ для Snegurochka к таблицам, содержащим информацию о подарках и письмах и запрет на изменение таблицы доставки.	4
2.2	Настроен полный доступ для Elf_logist к таблицам, содержащим информацию о доставках и и запрет на изменение таблицы писем.	5
2.3	Права доступа проверены через SHOW GRANTS	2
	Итого:	**20**
Модуль В. Написание SQL-запросов для анализа		
3.1	Представлен корректный ответ на задачу 3.1	5

3.2	Представлен корректный ответ на задачу 3.2	5
3.3	Представлен корректный ответ на задачу 3.3	5
3.4	Представлен корректный ответ на задачу 3.4	5
	Итого:	**20**
Модуль Г. Построение дашборда с визуализацией данных		
4.1	Данные подключены к BI (подключена база данных, либо данные загружены из файлов)	5
4.2	Создан индикатор для количества подарков на Складе А	3
4.1	Создан индикатор для количества подарков на Складе В	3
4.1	Создан индикатор для количества подарков на Складе С	3
4.1	Создана столбчатая диаграмма, отображающая количество упоминаний каждого подарка в письмах детей	4
4.2	Создан дашборд с заголовком “Дашборд Деда Мороза”, 3 индикатора и столбчатая диаграмма размещены на дашборде. Созданы логичные заголовки для группы индикаторов и диаграммы.	5
4.3	Публичный доступ к дашборду настроен и предоставлена ссылка	2
4.3	Участник соблюдал правила поведения и не превысил общее время выполнения конкурсного задания	5
	Итого:	**30**

Критерии оценки выполнения задания для студентов

Наименование модуля	**Задание**	**Максимальный балл**
Модуль 1: Создание структуры базы данных и заполнение данными	1. Запуск программы MySQL Workbench. 2. Создание схемы базы данных. 3. Разработка структуры базы данных с учетом принципов нормализации данных. Создание таблиц с первичными и внешними ключами. Создание индексов для оптимизации запросов. Проверка корректности выполнения через команду DESCRIBE. 3. Создание ER-диаграммы отображающей корректность созданной структуры базы данных. 4. Заполнение таблиц данными из предоставленных CSV файлов. 5. Создание Python-скрипта, реализующего интеграцию	30

	базы данных с Open-Meteo API.	
Модуль 2: Настройка прав доступа для пользователей	1. Создание трех пользователей с различными уровнями доступа: DedMoroz, Snegurochka и Elf_logist. 2. Настройка прав доступа для каждого пользователя. 3. Проверка прав доступа через SHOW GRANTS	20
Модуль 3: Написание SQL-запросов для анализа	1. Создание SQL-запроса для определения количества дней с неблагоприятными погодными условиями. 2. Создание SQL-запроса для анализа текущих активных доставок. 3. Создание SQL-запроса для определения достаточности запасов подарков на складе. 4. Создание SQL-запроса для определения активного возраста детей и их предпочтений.	20
Модуль 4: Построение дашборда с визуализацией данных	1. Подключение базы данных к BI. 2. Создание визуализаций (графиков) для отображения активности городов на основе количества писем, остатков подарков на складах с учетом критических запасов, распределения статуса доставок. 3. Создание дашборда включающего визуализации, заголовки и пояснения. Публикация дашборда в публичный доступ.	30
	ИТОГО:	**100**

Подробные критерии выполнения задания для студентов

Задание	**Наименование критерия**	**Максимальные баллы**
Модуль А. Создание структуры базы данных и заполнение данными		
1.1	Запущена программа MySQL Workbench.	2
1.2	Создана схема базы данных с именем DedMoroz_[номер участника].	1
1.3	Созданы таблицы с корректными полями и типами данных. Таблицы логически соответствуют файлам, предоставляющим данные.	4
1.3	Корректность выполнения созданных таблиц проверена через команду DESCRIBE.	2
1.3	База данных спроектирована корректно, нормализована (как минимум до 3НФ), корректно определены первичные и внешние ключи, обеспечивающие ссылочную целостность и отсутствие избыточности данных. Выбраны оптимальные типы данных для атрибутов.	5
1.3	Созданы простые и комбинированный индексы для оптимизации	5

	запросов.	
1.4	Создана ER-диаграмма, которая полностью соответствует нормализованной структуре базы данных. Отображены все таблицы и все связи между ними с правильной кардинальностью. Диаграмма экспортирована в формат .png.	2
1.5	Предоставлен SQL-код для импорта данных из предоставленных CSV-файлов в релевантные таблицы.	4
1.6	Предоставлен Python-скрипт, реализующий интеграцию с API и корректно загружающий данные о погоде в таблицу "Погода".	5
	Итого:	**30**
Модуль Б. Настройка прав доступа для пользователей		
2.1	Предоставлен SQL-код для создания трех пользователей: DedMoroz, Snegurochka, Elf_logist.	5
2.2	Предоставлен SQL-код для настройки полного доступа пользователя DedMoroz ко всем таблицам базы данных.	4
2.2	Настроен полный доступ для Snegurochka к таблицам, содержащим информацию о подарках и письмах и запрет на изменение таблицы доставки.	4
2.2	Настроен полный доступ для Elf_logist к таблицам, содержащим информацию о доставках и и запрет на изменение таблицы писем.	4
2.3	Предоставлен SQL-код для проверки корректности прав через SHOW GRANTS для каждого из трех пользователей.	3
	Итого:	**20**
Модуль В. Написание SQL-запросов для анализа		
3.1	Предоставлен SQL-запрос для определения количества дней с неблагоприятными погодными условиями по городам.	5
3.2	Предоставлен SQL-запрос для анализа текущих активных доставок (статус "доставляется"/”в пути”) и количества подарков по городам.	5
3.3	Предоставлен SQL-запрос для определения достаточности запасов подарков на складе для всех доставок, находящихся на этапе “сборка”.	5
3.4	Предоставлен SQL-запрос для определения активного возраста детей, отправляющих письма Деду Морозу и их предпочтений.	5
	Итого:	**20**
Модуль Г. Построение дашборда с визуализацией данных		
4.1	Данные подключены к BI (подключена база данных, либо данные загружены из файлов)	5

4.2	Визуализация отвечает на вопросы задачи Анализ писем.	5
4.2	Визуализация отвечает на вопросы задачи Остатки подарков.	5
4.2	Визуализация отвечает на вопросы задачи Статус доставки.	5
4.3	Создан дашборд с заголовком “Дашборд Деда Мороза”, на дашборде расположены созданные визуализации с заголовками и пояснениями.	3
4.4	Публичный доступ к дашборду настроен и предоставлена ссылка.	2
4.3	Участник соблюдал правила поведения и не превысил общее время выполнения конкурсного задания	5
	Итого:	**30**

Критерии оценки выполнения задания для специалистов

Наименование модуля	**Задание**	**Максимальный балл**
Модуль А: Создание структуры базы данных и заполнение данными	1. Запуск программы MySQL Workbench. 2. Создание схемы базы данных. 3. Разработка структуры базы данных с учетом принципов нормализации данных. Создание таблиц с первичными и внешними ключами. Создание индексов для оптимизации запросов. Проверка корректности выполнения через команду DESCRIBE. 3. Создание ER-диаграммы отображающей корректность созданной структуры базы данных. 4. Заполнение таблиц данными из предоставленных CSV файлов. 5. Создание Python-скрипта, реализующего интеграцию базы данных с Open-Meteo API.	30
Модуль Б: Настройка прав доступа для пользователей	1. Создание трех пользователей с различными уровнями доступа: DedMoroz, Snegurochka и Elf_logist. 2. Настройка прав доступа для каждого пользователя. 3. Проверка прав доступа через SHOW GRANTS 4. Создание триггера на таблицу “Письма”.	20
Модуль В: Написание SQL-запросов для анализа	1. Создание SQL-запроса для определения количества дней с неблагоприятными погодными условиями. 2. Создание SQL-запроса для анализа текущих активных доставок. 3. Создание SQL-запроса для определения точек срыва	20

	доставки. 4. Создание SQL-запроса для определения достаточности запасов.	
Модуль Г: Построение дашборда с визуализацией данных	1. Подключение базы данных к BI. 2. Создание визуализаций (графиков) для отображения активности писем, критических запасов подарков, проблемных маршрутов доставок, влияния температуры на доставку, региональной нагрузки. 3. Создание дашборда включающего визуализации, заголовки и пояснения. 4. Публикация дашборда в публичный доступ.	30
	ИТОГО:	**100**

Подробные критерии выполнения задания для студентов

Задание	Наименование критерия	Максимальные баллы
Модуль А. Создание структуры базы данных и заполнение данными		
1.1	Запущена программа MySQL Workbench.	2
1.2	Создана схема базы данных с именем DedMoroz_[номер участника].	1
1.3	Созданы таблицы с корректными полями и типами данных. Таблицы логически соответствуют файлам, предоставляющим данные.	4
1.3	Корректность выполнения созданных таблиц проверена через команду DESCRIBE.	2
1.3	База данных спроектирована корректно, нормализована (как минимум до 3НФ), корректно определены первичные и внешние ключи, обеспечивающие ссылочную целостность и отсутствие избыточности данных. Выбраны оптимальные типы данных для атрибутов.	5
1.3	Созданы простые и комбинированный индексы для оптимизации запросов.	5
1.4	Создана ER-диаграмма, которая полностью соответствует нормализованной структуре базы данных. Отображены все таблицы и все связи между ними с правильной кардинальностью. Диаграмма экспортирована в формат .png.	2
1.5	Предоставлен SQL-код для импорта данных из предоставленных CSV-файлов в релевантные таблицы.	4
1.6	Предоставлен Python-скрипт, реализующий интеграцию с API и корректно загружающий данные о погоде в таблицу "Погода".	5
	Итого:	**30**

Модуль Б. Настройка прав доступа для пользователей		
2.1	Предоставлен SQL-код для создания трех пользователей: DedMoroz, Snegurochka, Elf_logist.	3
2.2	Предоставлен SQL-код для настройки полного доступа пользователя DedMoroz ко всем таблицам базы данных.	3
2.2	Настроен полный доступ для Snegurochka к таблицам, содержащим информацию о подарках и письмах и запрет на изменение таблицы доставки.	4
2.2	Настроен полный доступ для Elf_logist к таблицам, содержащим информацию о доставках и и запрет на изменение таблицы писем.	4
2.3	Предоставлен SQL-код для проверки корректности прав через SHOW GRANTS для каждого из трех пользователей.	2
2.4	Предоставлен SQL-код для создания триггера, который автоматически контролирует возраст ребенка при создании нового письма Деду Морозу в таблице Письма	4
	Итого:	**20**
Модуль В. Написание SQL-запросов для анализа		
3.1	Предоставлен SQL-запрос для определения количества дней с неблагоприятными погодными условиями по городам.	5
3.2	Предоставлен SQL-запрос для анализа текущих активных доставок (статус "доставляется"/”в пути”) и количества подарков по городам.	5
3.3	Предоставлен SQL-запрос для определения точек срывов доставки.	5
3.4	Предоставлен SQL-запрос для определения достаточности запасов подарков на складе для всех доставок, находящихся на этапе “сборка”.	5
	Итого:	**20**
Модуль Г. Построение дашборда с визуализацией данных		
4.1	Данные подключены к BI (подключена база данных, либо данные загружены из файлов)	5
4.2	Визуализация отображает динамику активности писем по дням.	3
4.2	Визуализация отображает критически низкие запасы подарков.	3
4.2	Визуализация отображает проблемные маршруты доставки.	3
4.2	Визуализация отображает влияние погодных условий.	3
4.2	Визуализация отображает региональную нагрузку.	3
4.3	Создан дашборд с заголовком “Дашборд Деда Мороза”, на дашборде	3

	расположены созданные визуализации с заголовками и пояснениями.	
4.3	Публичный доступ к дашборду настроен и предоставлена ссылка.	2
4.3	Участник соблюдал правила поведения и не превысил общее время выполнения конкурсного задания	5
	Итого:	**30**

3. Перечень используемого оборудования, инструментов и расходных материалов (указывается для всех категорий участников отдельно или единый для всех категорий участников).

*Возможно использование аналогов (с аналогичными характеристиками) указанного оборудования, инструментов и расходных материалов.

3.1 Перечень оборудования на 1-го участника (школьники, студенты, специалисты)

ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ НА 1-ГО УЧАСТНИКА (конкурсная площадка)

№	Наименование	Ссылка на сайт с тех. характеристиками либо тех. характеристики оборудования, инструментов	Ед. измерения	Кол-во
1	Комплект ЭВМ: моноблок или персональный компьютер с клавиатурой, мышью, комплектом необходимых кабелей	Диагональ монитора 22" или более, ОЗУ 8 ГБ или более, CPU с поддержкой запуска СУБД	шт	1
2	ПО Операционная система	Windows 10 и выше или аналог	шт	1
3	ПО для работы с базами данных	MySQL Server, MySQL Connector/J, Community и Workbench, последняя версия	шт	1
4	ПО инклюзивной доступности	Assistapp	шт	1
5	ПО Python и Anaconda для подключения к Jupyter Notebook	Последняя версия Python с библиотеками requests и mysql.connector, Anaconda и Jupyter Notebook	шт	1
6	ПО для BI и визуализации	ПО для BI и визуализации	шт	1
7	ПО для создания отчетов	Текстовый редактор TXT	шт	1
8	Доступ в Интернет и браузер	Высокоскоростное подключение (интернет-канал с пропускной способностью не менее 10 Мбит/с (как на загрузку, так и на отдачу), задержкой не более 150 мс и потерями пакетов не более 1%)	точка	1
9	Стол	Офисный, 80х60 или более	шт	1
10	Стул	Офисный	шт	1

ПЕРЕЧЕНЬ РАСХОДНЫХ МАТЕРИАЛОВ НА 1 УЧАСТНИКА

№	Наименование	Ссылка на сайт с тех. характеристиками либо тех. характеристики оборудования, инструментов	Ед. измерения	Кол-во
1	Ручка	На усмотрение организатора	шт	1
2	Листы бумаги	Формат А4	шт	10

РАСХОДНЫЕ МАТЕРИАЛЫ, ОБОРУДОВАНИЕ И ИНСТРУМЕНТЫ, КОТОРЫЕ УЧАСТНИКИ ДОЛЖНЫ ИМЕТЬ ПРИ СЕБЕ (Toolbox)

Не требуется.

РАСХОДНЫЕ МАТЕРИАЛЫ И ОБОРУДОВАНИЕ, ЗАПРЕЩЕННЫЕ НА ПЛОЩАДКЕ

Те, что не указаны в перечне.

ДОПОЛНИТЕЛЬНОЕ ОБОРУДОВАНИЕ, ИНСТРУМЕНТЫ, КОТОРОЕ МОЖЕТ ПРИВЕСТИ С СОБОЙ УЧАСТНИК

Не требуется.

3.2. На 1-го эксперта (конкурсная площадка)

ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И МЕБЕЛИ

№	Наименование	Ссылка на сайт с тех. характеристиками либо тех. характеристики оборудования, инструментов	Ед. измерения	Кол-во
1	Комплект ЭВМ: моноблок или персональный компьютер с клавиатурой, мышью, комплектом необходимых кабелей	Диагональ монитора 22" или более, ОЗУ 8 ГБ или более, CPU с поддержкой запуска СУБД	шт	1
2	ПО Операционная система	Windows 10 и выше или аналог	шт	1
3	ПО для работы с базами данных	MySQL Server, MySQL Connector/J, Community и Workbench, последняя версия	шт	1
4	ПО инклюзивной доступности	Assistapp	шт	1
5	ПО Python и Anaconda для подключения к Jupyter Notebook	Последняя версия Python с библиотеками requests и mysql.connector, Anaconda и Jupyter Notebook	шт	1

6	ПО для BI и визуализации	ПО для BI и визуализации	шт	1
7	ПО для создания отчетов	Текстовый редактор TXT	шт	1
8	Доступ в Интернет и браузер	Высокоскоростное подключение (интернет-канал с пропускной способностью не менее 10 Мбит/с (как на загрузку, так и на отдачу), задержкой не более 150 мс и потерями пакетов не более 1%)	точка	1
9	Стол	Офисный, 80х60 или более	шт	1
10	Стул	Офисный	шт	1

3.3. Общая инфраструктура конкурсной площадки

ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ, ИНСТРУМЕНТОВ, СРЕДСТВ ИНДИВИДУАЛЬНОЙ ЗАЩИТЫ И Т.П.

№	Наименование	Ссылка на сайт с тех. характеристиками либо тех. характеристики оборудования	Ед. измерения	Кол-во
1	Аптечка	На усмотрение организатора	шт	1
2	Огнетушитель	Класс В - 55 В, Класс А - 2 А или на усмотрение организатора	шт	1
3	Кулер	На усмотрение организатора	шт	1
4	Одноразовые стаканчики	На усмотрение организатора	шт	100

3.4. Комната участников

ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ, МЕБЕЛИ, КАНЦЕЛЯРИИ И Т.П.

№	Наименование	Ссылка на сайт с тех. характеристиками либо тех. характеристики оборудования	Ед. измерения	Кол-во
1	Вешалка	На усмотрение организатора	шт	1
2	Стол	На усмотрение организатора	шт	1
3	Стул	На усмотрение организатора	шт	5
4	Шкафчики (ячейки) индивидуальные	На усмотрение организатора	шт	1

5	Мусорная корзина	На усмотрение организатора	шт	1

3.5. Дополнительные требования/комментарии

Количество точек питания и их характеристики

№	Наименование	Тех. характеристики
1	Количество точек электропитания	По согласованию с главным экспертом

4. Минимальные требования к оснащению рабочих мест с учетом всех основных нозологий

Данный раздел включает требования к оснащению рабочих мест для участников с разными нозологиями, разработанные в соответствии с п. 8.2.1 Свода правил СП 59.13330.2016 «Доступность зданий и сооружений для маломобильных групп населения». Все требования учитывают адаптацию рабочего пространства для создания комфортных и безопасных условий труда.

Нозология	**Площадь, м.кв.**	**Ширина прохода между рабочими местами, м.**	**Специализированное оборудование, количество. ***
Рабочее место участника с нарушением слуха	3000x1900	1,5 м	Для участников с нарушением слуха необходимо предусмотреть: а) наличие звукоусиливающей аппаратуры, акустической системы, информационной индукционной системы, наличие индивидуальных наушников; б) наличие на площадке переводчика русского жестового языка (сурдопереводчика); в) оформление конкурсного задания в доступной
Рабочее место участника с нарушением зрения	3000x1900	1,5 м	Для участников с нарушением зрения необходимо: а) текстовое описание конкурсного задания в плоскопечатном виде с крупным размером шрифта, учитывающим состояние зрительного анализатора участника с остаточным зрением (в формате Microsoft Word не менее 16-18 пт), дублированного рельефно точечным шрифтом Брайля (при необходимости); - лупа с подсветкой для слабовидящих; электронная лупа; б) для рабочего места, предполагающего работу на компьютере - оснащение специальным компьютерным оборудованием и оргтехникой: - видеоувеличитель; - программы экранного доступа NVDA и JAWS18 (при необходимости); - брайлевский дисплей (при необходимости); в) для рабочего места участника с нарушением зрения, имеющего собаку-проводника, необходимо предусмотреть место для собаки-проводника; г) оснащение (оборудование) специального рабочего места тифлотехническими

			ориентирами и устройствами, с возможностью использования крупного рельефно-контрастного шрифта и шрифта Брайля, акустическими навигационными средствами, обеспечивающими беспрепятственное нахождение инвалидом по зрению - слепого своего рабочего места и выполнение трудовых функций; д) индивидуальное равномерное освещение не менее 300 люкс.
Рабочее место участника с нарушением ОДА	3000х1900	1,5 м	Оснащение (оборудование) специального рабочего места оборудованием, обеспечивающим реализацию эргономических принципов: а) увеличение размера зоны на одно место с учетом подъезда и разворота кресла-коляски, увеличения ширины прохода между рядами верстаков; б) для участников, передвигающихся в кресле-коляске, необходимо выделить 1 - 2 первых рабочих места в ряду у дверного проема; в) оснащение (оборудование) специального рабочего места специальными механизмами и устройствами, позволяющими изменять высоту и наклон рабочей поверхности, положение сиденья рабочего стула по высоте и наклону, угол наклона спинки рабочего стула, оснащение специальным сиденьем, обеспечивающим компенсацию усилия при вставании.
Рабочее место участника с соматическими заболеваниями	3000х1900	1,5 м	Специальные требования к условиям труда инвалидов вследствие заболеваний сердечно-сосудистой системы, а также инвалидов вследствие других соматических заболеваний, предусматривают отсутствие: а) вредных химических веществ, включая аллергены, канцерогены, оксиды металлов, аэрозоли преимущественно фиброгенного действия; б) тепловых излучений; локальной вибрации, электромагнитных излучений, ультрафиолетовой радиации на площадке; в) превышения уровня шума на рабочих местах; г) нарушений уровня освещенности, соответствующей действующим нормативам. Необходимо обеспечить наличие столов с регулируемыми высотой и углом наклона

			поверхности; стульев (кресел) с регулируемыми высотой сиденья и положением спинки (в соответствии со спецификой заболевания).
Рабочее место участника с ментальными нарушениями	3000x1900	1,5 м	Специальные требования к условиям труда инвалидов, имеющих нервно-психические заболевания: а) создание оптимальных и допустимых санитарно-гигиенических условий производственной среды, в том числе: температура воздуха в холодный период года при легкой работе - 21 - 24 °C; при средней тяжести работ - 17 - 20 °C; влажность воздуха в холодный и теплый периоды года 40 - 60 %; отсутствие вредных веществ: аллергенов, канцерогенов, аэрозолей, металлов, оксидов металлов; б) электромагнитное излучение - не выше ПДУ; шум - не выше ПДУ (до 81 дБА); отсутствие локальной и общей вибрации; отсутствие продуктов и препаратов, содержащих живые клетки и споры микроорганизмов, белковые препараты; в) оборудование (технические устройства) должны быть безопасны и комфортны в использовании (устойчивые конструкции, прочная установка и фиксация, простой способ пользования без сложных систем включения и выключения, с автоматическим выключением при неполадках; расстановка и расположение, не создающие помех для подхода, пользования и передвижения; расширенные расстояния между столами, мебелью; не должна затрудняющая доступность устройств;

Примечания:

1. Все рабочие места должны быть оборудованы индивидуальным освещением и иметь доступ к электропитанию.
2. Участники с ментальными нарушениями и нарушениями зрения должны иметь возможность адаптировать интерфейсы программного обеспечения (контраст, размер шрифта, озвучивание) при помощи ПО инклюзивной доступности.
3. Для участников с нарушениями ОДА рекомендуется располагать рабочие места ближе к входу для упрощения доступа.
4. В случае необходимости на площадке должны быть доступны квалифицированные ассистенты для помощи участникам с ограничениями.

Источники:

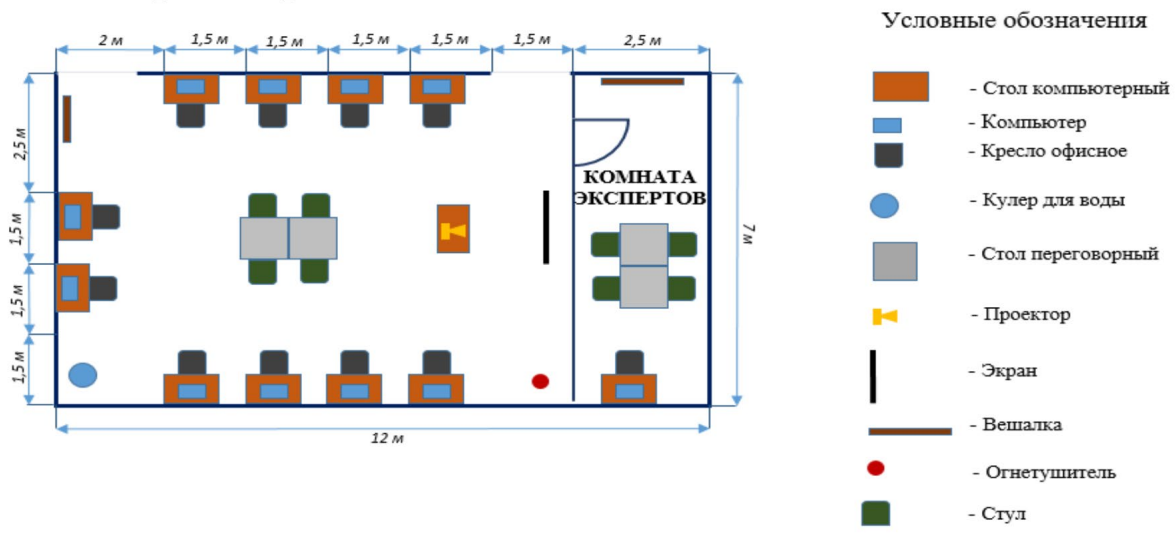
- СП 59.13330.2016 «Доступность зданий и сооружений для маломобильных групп населения».
- Методические рекомендации по адаптации рабочих мест для маломобильных групп населения.

5. Требования охраны труда и техники безопасности

5.1 Общие положения

Настоящий раздел определяет требования охраны труда и техники безопасности для участников и экспертов (далее — участники) соревнований по компетенции "Администрирование баз данных" в рамках чемпионата "Абилимпикс".

1. К выполнению конкурсных заданий допускаются лица, прошедшие инструктаж по технике безопасности, ознакомившиеся с правилами эксплуатации оборудования и не имеющие медицинских противопоказаний.
2. Участники обязаны соблюдать указания экспертов, требования охраны труда и правила внутреннего распорядка на площадке.
3. Освещение рабочих мест должно соответствовать нормам СанПиН 2.2.2/2.4.1340-03.
4. Каждый участник должен знать местоположение:
    - средств пожаротушения;
    - аптечки первой помощи;
    - путей эвакуации и плана здания.
5. В случае возникновения любого происшествия, несчастного случая или неисправности оборудования, необходимо немедленно сообщить эксперту.
6. Схема застройки соревновательной площадки:



5.2 Требования перед началом работы

Перед началом работы участники должны:

- Подготовить рабочую зону, убрать лишние предметы.
- Проверить состояние стола, стула и других элементов рабочего места.
- Убедиться в правильной настройке оборудования:
    - Монитор на расстоянии 50-70 см от глаз;
    - Угол наклона экрана и высота стула отрегулированы для удобства;
    - Кабели питания не препятствуют работе и не создают опасности.

- Проверить отсутствие бликов и засветок на экране монитора.
- Убедиться в наличии всех необходимых материалов и инструментов.

Подключение оборудования и первичная настройка должны производиться в последовательности, установленной инструкцией производителя.

Участник обязан сообщить эксперту о выявленных неисправностях или недостатках оборудования до начала выполнения задания.

5.3 Требования во время работы

1. Во время выполнения конкурсных заданий участники обязаны:
    - Содержать рабочее место в порядке и чистоте.
    - Не размещать жидкости и сыпучие материалы рядом с оборудованием.
    - Выполнять задания в соответствии с инструкцией и указаниями экспертов.
    - Делать регулярные перерывы для снятия зрительного и физического напряжения, согласно установленному расписанию.
2. Запрещается:
    - Разбирать, ремонтировать или перемещать оборудование без разрешения эксперта.
    - Закрывать вентиляционные отверстия оборудования.
    - Прикасаться мокрыми руками к электроприборам.
    - Использовать оборудование не по назначению.
    - Нарушать порядок работы или отвлекать других участников.
3. В случае неисправностей, перегрева или появления запаха гари немедленно отключить оборудование и сообщить об этом эксперту.
4. При травме, внезапном ухудшении самочувствия или другой чрезвычайной ситуации участник обязан прекратить работу и обратиться за помощью к эксперту.

5.4 Требования по окончании работы

После завершения конкурсного задания участники обязаны:

- Привести в порядок рабочее место, убрать материалы и инструменты.
- Сообщить эксперту о завершении задания и любых возникших во время работы неисправностях.
- Завершить работу на компьютере, корректно завершив все запущенные процессы.
- Отключить оборудование в последовательности, установленной инструкцией.

Все выявленные недостатки в оборудовании или его настройке должны быть зафиксированы и переданы для анализа техническому персоналу.

5.5 Дополнительные рекомендации

1. Участникам настоятельно рекомендуется следить за собственным состоянием и в случае усталости сообщить об этом эксперту для возможной корректировки времени выполнения задания.
2. Эксперты обязаны следить за соблюдением правил охраны труда и обеспечивать оперативную помощь участникам в случае необходимости.

3. Организаторы соревнований должны предусмотреть наличие резерва оборудования и инструментов на случай возникновения неисправностей.