**ЗАДАНИЕ**

**ДЕМОНСТРАЦИОННОГО ЭКЗАМЕНА**

(2024 год)

|  |  |
| --- | --- |
| **Код и наименование профессии (специальности) среднего профессионального образования** | 09.02.07 Информационные системы и программирование |
| **Наименование квалификации (направленности)** | Администратор баз данных |
| **Вид аттестации** | Государственная итоговая аттестация |
| **Уровень демонстрационного экзамена** | профильный |
| **Шифр варианта задания** | В3\_\_КОД 09.02.07-1-2024-ПУ |

**Вариант № 3**

|  |
| --- |
| Модуль 1: Разработка, администрирование и защита баз данных |
| Задание модуля 1:  Выберите СУБД и среду для управления инфраструктурой.  Установите ядро выбранной СУБД и среду для управления инфраструктурой SQL (на виртуальную машину или представленный компьютер). При установке задайте имя сервера – «SQLServer\_ номер вашего рабочего места», например SQLServer\_05. У сервера должен быть включен режим смешанной аутентификации.  Включите или создайте пользователя sa, установив пароль «D\_номер вашего рабочего места», например D\_05.  Напишите скрипт, который позволит автоматически:   * создать 9 пользователей d1, d2, d3, …, d9, у которых пароль формируется случайным образом и содержит 8 символов (буквы, цифры); * базы данных Base1, Base 2, Base3, …, Base9; * настроить права доступа пользователей к базам данных. Пользователь d1 имеет доступ только к базе данных Base1, d2 имеет доступ только к базе данных Base2 и т. д. Пользователям необходимо запретить создавать базы данных, отзывать и отклонять разрешения, предоставляющих или запрещающих доступ к базе данных. * создать базу данных BaseAll и таблицу Users, для хранения пользователей и их паролей; * заполнить таблицу Users данными созданных пользователях и паролях.   Хранение паролей в зашифрованном виде очень важно для безопасности доступа к серверу, поэтому создайте скрип, который зашифрует все пароли в таблице Users.  Чтобы предотвратить утрату доступа к аккаунту и потерю данных создайте скрип, который позволит отобразить данные из таблицы Users c расшифрованными паролями.  Напишите скрипт, который позволит провести резервное копирование базы данных BaseAll. Необходимо предоставить скрипт и файл бэкапа.  Напишите скрипт, который позволит провести процедуру восстановления базы данных. |
| Описание предметной области:  Гостиница "Золотая перезагрузка" - это место, где каждый гость может насладиться комфортным проживанием и отдохнуть, находясь в самом центре городской суеты. Гостиница предоставляет широкий выбор номеров, которые отличаются простором, уютом и современным дизайном.  Клиент может забронировать номер через сайт или по телефону. При заселении в гостиницу формируется регистрационная карта гостя. В этой карте указываются данные о клиенте, включая его фамилию, имя, отчество (не обязательное), дату рождения, паспортные данные (серия, номер, дата выдачи, кем выдан), номер телефона, электронную почту, а также данные о предоставленных услугах: даты проживания, номер проживания, вид оплаты, дополнительные услуги, которыми воспользовался клиент за все время проживания, например, бассейн, спортивный зал, прокат велосипедов и т. д.  Гостиница имеет номера оснащены всем необходимым для приятного проживания. Стоимость номера зависит от его класса (эконом-класс, стандарт, полулюкс, люкс, президентский номер).  В зависимости от спроса сотрудники организации могут менять цену на дополнительные услуги.  Для сотрудников организации необходимо хранить глобальный уникальный идентификатор в формате GUID. Всякий раз, когда добавляется новый сотрудник оно должно генерироваться автоматически. Так же необходимо хранить ФИО, дата рождения, телефон, адрес (город, улица, дом, квартира). Отчество является не обязательным для заполнения.  На основе описания предметной области Вам необходимо спроектировать ER-диаграмму для информационной системы. Обязательна 3 нормальная форма с обеспечением ссылочной целостности. При разработке диаграммы обратите внимание на согласованную осмысленную схему именования, создайте необходимые первичные и внешние ключи, определите ограничения внешних ключей, отражающие характер предметной области.  ER - диаграмма должна быть представлена в формате .pdf и содержать таблицы, связи между ними, атрибуты и ключи (типами данных на данном этапе можно пренебречь).  Создайте базу данных, используя предпочтительную платформу, на сервере баз данных, который вам предоставлен. Создайте таблицы основных сущностей, атрибуты, отношения и необходимые ограничения. В любом случае созданные таблицы должны содержать начальные тестовые данные. В каждой таблице должно быть как минимум 2 записи, а в таблице заказы и покупки клиентов 10 записей. Предусмотреть, чтобы на одного клиента было зарегистрировано несколько регистрационных карт, а каждая карта гостя должна содержать несколько дополнительных услуг.  Создайте процедуру, которая проверяет адрес электронной почты, хранящийся в базе данных, на корректность. Корректным является адрес, если он содержит допустимые символы в каждой из частей и части разделены сначала символом @ затем «.» (Формат адреса электронной почты: A-Z0-9 **@** A-Z0-9 **.** A-Z0-9). Дополнительно необходимо проверить, чтобы в адресе отсутствовали символы, из-за которых часто возникают ошибки ( [ " < > ' ]). Процедура выводит все адреса электронных почт с указанием соответствующего признака валидности (1-валидный, 0 – не валидный).  Для хранения истории изменения цен на дополнительные услуги создайте таблицу HistoryCost со следующей структурой: Дата изменения, Услуга, Старое значение цены, Новое значение цены). Создайте триггер, который при изменении цены на дополнительные услуги будет записывать данные в таблицу HistoryCost. Предполагается, что за один раз будет обновление только одной строки |
| Модуль 2: Соадминистрирование баз данных и серверов |
| Задание модуля 2:  Напишите запросы к созданной базе данных и предоставьте их скрипты:   * Выведите список клиентов с указанием дат проживания и суммы к оплате. Сумма оплаты рассчитывается как произведение цены номера за сутки на количество дней проживания + стоимость всех дополнительных услуг, которыми воспользовался клиент. * Удалите из базы данных все услуги, которые ни разу не были заказаны клиентом. * Обновите цену популярного номера, увеличив её на 15 %. |
| Модуль 3: Разработка модулей программного обеспечения для компьютерных систем |
| Задание модуля 3:  Создайте модуль программного обеспечения, который позволит анализировать информацию из созданной базы данных.  Макет окна представлен на рисунке 1.    Рисунок 1 – Макет окна работы с регистрационными картами клиентов  Подключите к приложению созданную базу данных и реализуйте следующий функционал:   * сортировку данных по выбранному полю. Пользователь выбирает поле для сортировки, после включения переключателя варианта сортировки должна происходить сортировка данных в таблице. * фильтрацию записей в таблице по клиенту. Пользователь выбирает из списка клиента и по нажатию кнопки «Фильтровать» должна происходить фильтрация данных в таблице, т. е. отображаться данные о проживании выбранного клиента. По нажатию на кнопку «Показать все» отменяется фильтрация записей. * поиск данных. Пользователь вводит строку поиска и по нажатию кнопки «Найти» должны выделяться цветом ячейки, в которых нашлось совпадение со строкой поиска.   Необходимо так же разработать руководство пользователя для данного модуля. Опишите последовательность действий для выполнения всех функций модуля с использованием скриншотов.  *Требования к разработке*  Название приложения  Используйте соответствующие названия для ваших приложений и файлов. Так, например, наименование настольного приложения должно обязательно включать название компании- заказчика.  Файловая структура  Файловая структура проекта должна отражать логику, заложенную в приложение. Например, все формы содержатся в одной директории, пользовательские визуальные компоненты – в другой, классы сущностей – в третьей.  Структура проекта  Каждая сущность должна быть представлена в программе как минимум одним отдельным классом. Классы должны быть небольшими, понятными и выполнять одну единственную функцию (Single responsibility principle).  Макет и технические характеристики  Все компоненты системы должны иметь единый согласованный внешний вид, а также следующим требованиям:   * разметка и дизайн (предпочтение отдается масштабируемой компоновке); * должно присутствовать ограничение на минимальный размер окна; * должна присутствовать возможность изменения размеров окна, где это необходимо; * увеличение размеров окна должно увеличивать размер контентной части, например, таблицы с данными из БД); * группировка элементов (в логические категории); * использование соответствующих элементов управления (например, выпадающих списков для отображения подстановочных значений из базы данных); * расположение и выравнивание элементов (метки, поля для ввода и т.д.); * последовательный переход фокуса по элементам интерфейса (по нажатию клавиши TAB); * общая компоновка логична, понятна и проста в использовании; * соответствующий заголовок на каждом окне приложения (не должно быть значений по умолчанию типа MainWindow, Form1 и тп).   Обратная связь с пользователем  Уведомляйте пользователя о совершаемых им ошибках или о запрещенных в рамках задания действиях, информируйте об отсутствии результатов поиска и т.п. Окна сообщений соответствующих типов (например, ошибка, предупреждение, информация) должны отображаться с соответствующим заголовком и пиктограммой. Текст сообщения должен быть полезным и информативным, содержать полную информацию о совершенных ошибках пользователя и порядок действий для их исправления. Также можно использовать визуальные подсказки для пользователя при вводе данных.  Обработка ошибок  Не позволяйте пользователю вводить некорректные значения в текстовые поля сущностей. Например, в случае несоответствия типа данных или размера поля введенному значению. Оповестите пользователя о совершенной им ошибке.  При возникновении непредвиденной ошибки приложение не должно аварийно завершать работу.  Оформление кода  Идентификаторы переменных, методов и классов должны отражать суть и/или цель их использования, в том числе и наименования элементов управления (например, не должно быть значений по умолчанию типа Form1, button3).  Идентификаторы должны соответствовать соглашению об именовании (Code Convention) и стилю CamelCase или snake\_case.  Допустимо использование не более одной команды в строке.  Комментарии  Используйте комментарии для пояснения неочевидных фрагментов кода. Запрещено комментирование кода. Хороший код воспринимается как обычный текст. Не используйте комментарии для пояснения очевидных действий. Комментарии должны присутствовать только в местах, которые требуют дополнительного пояснения |