Задание подготовлено в рамках проекта АНО «Лаборатория модернизации образовательных ресурсов» «Кадровый и учебно-методический ресурс формирования общих компетенций обучающихся по программам СПО», который реализуется с использованием гранта Президента Российской Федерации на развитие гражданского общества, предоставленного Фондом президентских грантов.

**Разработчик**

Лысенко Ирина Владимировна, ГАПОУ «Тольяттинский индустриально-педагогический колледж»

Мясин Дмитрий Николаевич, ГАПОУ «Тольяттинский индустриально-педагогический колледж»

**Назначение задания**

Текущий контроль и коррекция деятельности. Уровень II

ПМ.11 Разработка, администрирование и защита баз данных.

МДК.11.01 Технология разработки и защиты баз данных

Тема: Обеспечение качества и надежности баз данных

Ваша компания разрабатывает базы данных (БД). Каждый разработчик участвует в своем этапе разработки объектов БД и передает готовое решение дальше следующему в технологической цепочке. Ваш коллега создает таблицы и схему данных по готовым моделям проектировщиков. Это самая важная стадия физического создания БД. Если разработчик неправильно определит поля таблицы, забудет выделить первичные ключи или выделить их неправильно (не тот тип, не то поле), если забудет выделить внешние ключи для связей и неправильно создаст связи, то потом все объекты БД будут работать неправильно, и вся база не будет функционировать. Вы выполняете работы следующего этапа - создаете запросы, формы и отчеты.

Внимательно изучите схему данных поликлиники, созданную вашим коллегой (источник 1), изучите стандартные требования к таблицам и схеме данных, этапы создания схемы данных (источник 2).

**Составьте план контроля схемы данных перед дальнейшей разработкой БД.**

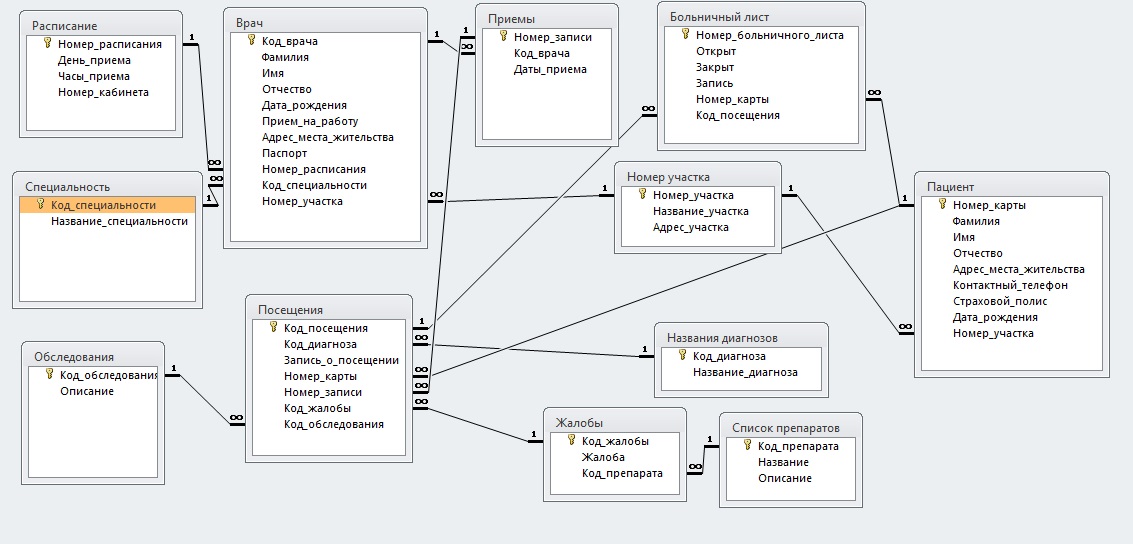
Заполните бланк. Отчеркивайте строку для каждого нового наименования работы.

*Бланк*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Точка контроля | Что проверяем | Как проверяем | Ожидаемый результат |
|  |  |  |  |  |

***Источник 1***

**Схема данных поликлиники**

**

***Источник 2***

**Создание схемы данных**

Создание схемы данных стоит из создания таблиц и ключевых полей, связывания таблиц, оптимизации схемы.

При проектировании таблиц, рекомендуется руководствоваться следующими основными принципами:

1. Информация в таблице не должна дублироваться. Не должно быть дублирования одинаковых данных и между таблицами (это называется избыточность даны). Когда определенная информация хранится только в одной таблице, то и изменять ее придется только в одном месте. Это делает работу более эффективной, а также исключает возможность несовпадения информации в разных таблицах. Например, в одной таблице должны содержаться адреса и телефоны клиентов.

2. Каждая таблица должна содержать информацию только на одну тему. Сведения на каждую тему обрабатываются намного легче, если они содержатся в независимых друг от друга таблицах. Например, адреса и заказы клиентов лучше хранить в разных таблицах, с тем, чтобы при удалении заказа информация о клиенте осталась в базе данных.

3. Каждая таблица должна содержать необходимые поля. Каждое поле в таблице должно содержать отдельные сведения по теме таблицы. Например, в таблице с данными о клиенте могут содержаться поля с названием компании, адресом, городом, страной и номером телефона. При разработке полей для каждой таблицы необходимо помнить, что каждое поле должно быть связано с темой таблицы.

Необходимо четко прописать у всех таблиц имена, имена полей.

Если вы уверены, что все таблицы созданы верно можно переходить к следующему виду работ - создание связей. Для каждой таблицы нужно выделить ключевые поля. База данных должна иметь первичный ключ. Это необходимо для того, чтобы СУБД могла связать данные из разных таблиц, например, данные о клиенте и его заказы. Нужно также выделить внешние ключи (для формирования связей один-ко-многим).

Это самая важная стадия создания БД. Если разработчик неправильно определить поля таблицы, забудет выделить первичные ключи или выделить их неправильно (не тот тип, не то поле), если забудет выделить внешние ключи для связей- то потом, при создании базы данных, эти объекты будут работать неправильно и вся база данных не будет функционировать.

Убедившись, что все связи созданы верно, нужно еще обеспечить оптимальность схемы данных позволяет повысить читабельность схем- простоту и удобство чтения схемы - за счет устранения пересечений линий связи, выравнивание таблиц по сетке и т.д.

Таблица 1.Этапы создания схемы данных

| № | Этап создания схемы данных | Инструмент СУБД и технологические операции |
| --- | --- | --- |
| 1 | Знакомство с инфологической моделью, созданной на стадии проектирования |  |
| 2 | Создание таблиц. Добавление в таблицы первичных и внешних ключей | Создем все таблицы: мастер таблиц\  конструктор таблиц🡪 указывается уникальное имя таблицы🡪 создаются поля таблицы, их типы и свойства.  Помечается ключевое поле \*.Добавляются внешние ключи\*\* |
| 4 | Внесение таблиц в схему данных | Схема данных🡪 добавить таблицы, добавляем все созданные таблицы |
| 5 | Создание связей | Схема данных🡪 перетаскиваем мышью ключ главной таблицы на внешний ключ подчиненной таблицы. Появятся линии связи |
| 6 | Обеспечение целостности данных | Щелкаем по линии связи правой кнопкой мыши🡪 обеспечение целостности данных: каскадное обновление и удаление данных. Появятся значки 1 или ∞ на линии связи. |
| 7 | Оптимизация схемы данных (устранение пересечений линий связи, выравнивание таблиц по сетке и т.д.) | Перетащить мышью и выровнять таблицы по сетке. Мышью перетянуть связи так, чтобы они пересекались минимально |

\*Уникальное поле таблицы, которое однозначно определяет каждую строку таблицы - данные не повторяются, что делает каждую строку уникальной даже при других совпадающих значениях (для двух студентов Иванов Петр будут созданы две строки с разными номерами студентов)

\*\*Внешний ключ - это такое же (по имени и по типу) поле, как ключ в главной таблице. Для реализации связи 1-к-1 связываются 2 первичных ключа. Для связи 1-ко-многим связываются ключ главной таблицы и внешнее добавочное поле с тем же названием и типом.

При разработке большого количества баз данных под заказ в фирмах-разработчиках очень важным показателем качества работы сотрудника, кроме качества исполнения самой базы данных, является время исполнения заданных работ. Оно прямо влияет на производительность работ - а значит прибыль фирмы, и часто должно соответствовать времени, отведенного заказчиком на исполнение заказа. Поэтому при многопоточной разработке контроль времени является одной из важнейших задач и производится на каждой стадии разработки. Каждая точка контроля создается для проверки всего комплекса работ, которые выполняются непрерывно и нет смысла их прерывать для контроля. Обычно это однотипные операции или операции над одинаковыми \ общными объектами базы данных.

*Использованы материалы источников:*

[*https://plati.market/itm/database-clinic/2030084*](https://plati.market/itm/database-clinic/2030084)

[*http://access.avorut.ru/load/bazy\_dannykh\_ms\_access/kontrolnye\_raboty\_po\_bazam\_dannykh\_ms\_access/skachat\_bazu\_dannykh\_access\_biznes\_processy\_polikliniki/3-1-0-275*](http://access.avorut.ru/load/bazy_dannykh_ms_access/kontrolnye_raboty_po_bazam_dannykh_ms_access/skachat_bazu_dannykh_access_biznes_processy_polikliniki/3-1-0-275)

Инструмент проверки

| № | Точка контроля | Что проверяем | Как проверяем | Ожидаемый  результат |
| --- | --- | --- | --- | --- |
| 1 | Контроль создания таблиц | Контроль времени создания таблиц | Отмечаем дату и время начала и окончания работ | 1 час |
| Правильность названия таблиц | Сравнение имен таблиц | Все имена уникальны \не повторяются |
| Правильность создания ключей таблиц | Поиск в каждой таблице первичного ключевого поля таблицы | У каждой таблицы выделен первичный ключ |
| Проверка выполнения требований к ключу - смысловой анализ имени поля ключа | Ключ по смыслу является полем, однозначно определяющим каждую строку \данные в ключевом поле не будут повторяться |
| 2 | Контроль создания связей | Контроль времени создания связей | Отмечаем дату и время начала и окончания работ | 0,5 часа |
| Наличие всех таблиц в схеме данных | Считаем количество таблиц | Количество таблиц соответствует модели |
| Правильность создания связей | Ищем наличие линий связи в соответствии с моделью | Все связи созданы |
| Проверка - в связи участвуют поля с одинаковыми названиями | Связь создана верно |
| Все значки на линии связи 1-∞ | Все связи (типа один-ко-многим) созданы верно |
| В связи участвует первичный ключ первой таблицы и внешний ключ (поле с таким же именем, созданное для связи) второй таблицы | Связи один-ко-многим созданы верно |
| Обеспечение целостности данных | Присутствуют значки на линии связи | Обеспечена целостность данных: каскадное обновление и удаление данных |
| 3 | Оценка оптимальности схемы данных | Контроль времени оценки | Отмечаем дату и время начала и окончания работ | 10 минут |
| Читабельность схемы данных | Проверка оптимальности схемы\пересечений линий связи, выявление возможности устранения пересечений, выдаются рекомендации по дооптимизации | Линии связи пересекаются минимально \линии связи построены верно |

*Подсчет баллов*

|  |  |
| --- | --- |
| За каждый верно указанный этап | 1 балл |
| *Максимально* | *4 баллов* |
| За каждый верно указанный предмет контроля | 1 балл |
| *Максимально* | *6 баллов* |
| За каждый верно указанный способ контроля | 1 балл |
| *Максимально* | *10 баллов* |
| За каждый верно указанный ожидаемый результат | 1 балл |
| *Максимально* | *10 баллов* |
| ***Максимальный балл*** | ***30 баллов*** |