**Кейс №3**

**«База данных: создание таблиц и связей»**

**Выполнение работы:**

Посмотрев на структуру проекта, показанную в предыдущем приложении, вы заметите, что файл с моделями у меня один. К этому я пришел не сразу, а спустя час попыток разделить логику на компоненты. Это решение делало разработку затруднительной, а понимание проекта… нуу… непонятным.

Дело в том, что модели представляют по своей сути объекты, представляющие из себя структуру данных, из которой ORM в будущем создаст таблицы в базе данных.

Однако по скольку база данных используется реляционная, то при попытке установить связи между таблицами приводили к непониманию. Если сейчас все модели находятся в одном файле, то раньше для каждой модели был свой файл. И при попытке выстроить связи между ними возникало много вопросов. Один из них «куда мне положить эту связь». В конечном итоге я соединил все в один файл и поверьте, работать стало в разы проще.

Для начала рассмотрим модель, которую я создал в draw.io, для простоты работы. Она очень часто спасала меня и помогала найти дыры в логике построения информационной системы.

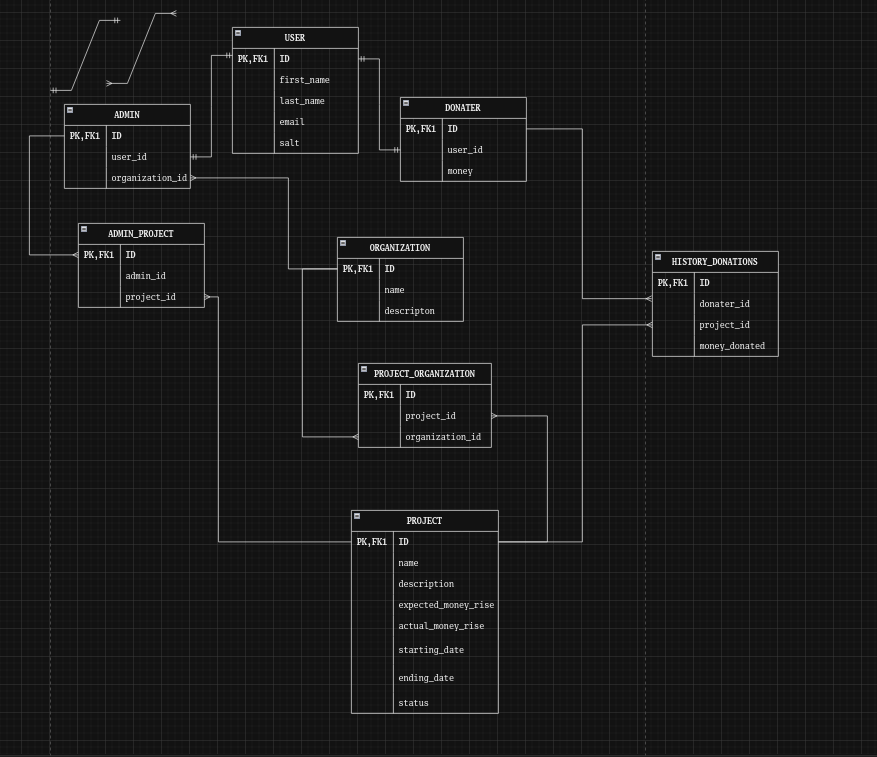


Рисунок 1

Опираясь на созданную модель, я начал создавать модели для базы данных.

Начать решил с пользователя.

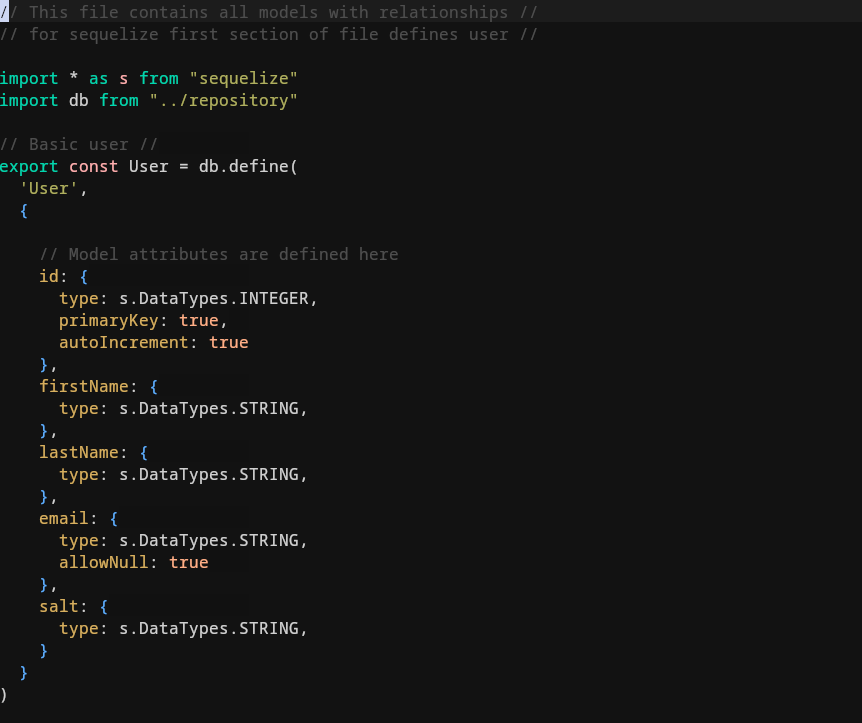


Рисунок 2

Что здесь происходит? По сути это пример представления таблицы базы данных формате модели. Id, firstName, lastName и salt — колонки в базе данных. А type — это тип данных, который эта колонка должна хранить. Далее на основе этих данных ORM создаст таблицу с именем user в СУБД. Это вся суть ORM.

Далее я создал остальные таблицы. По аналогии.

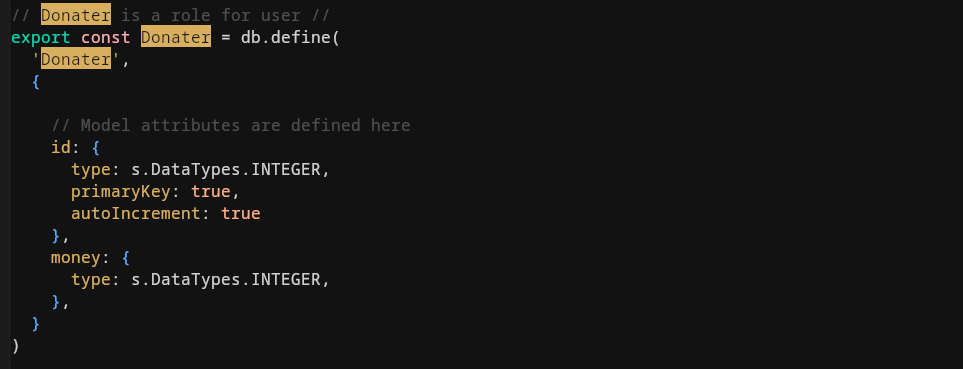


Рисунок 3

Рисунок 4

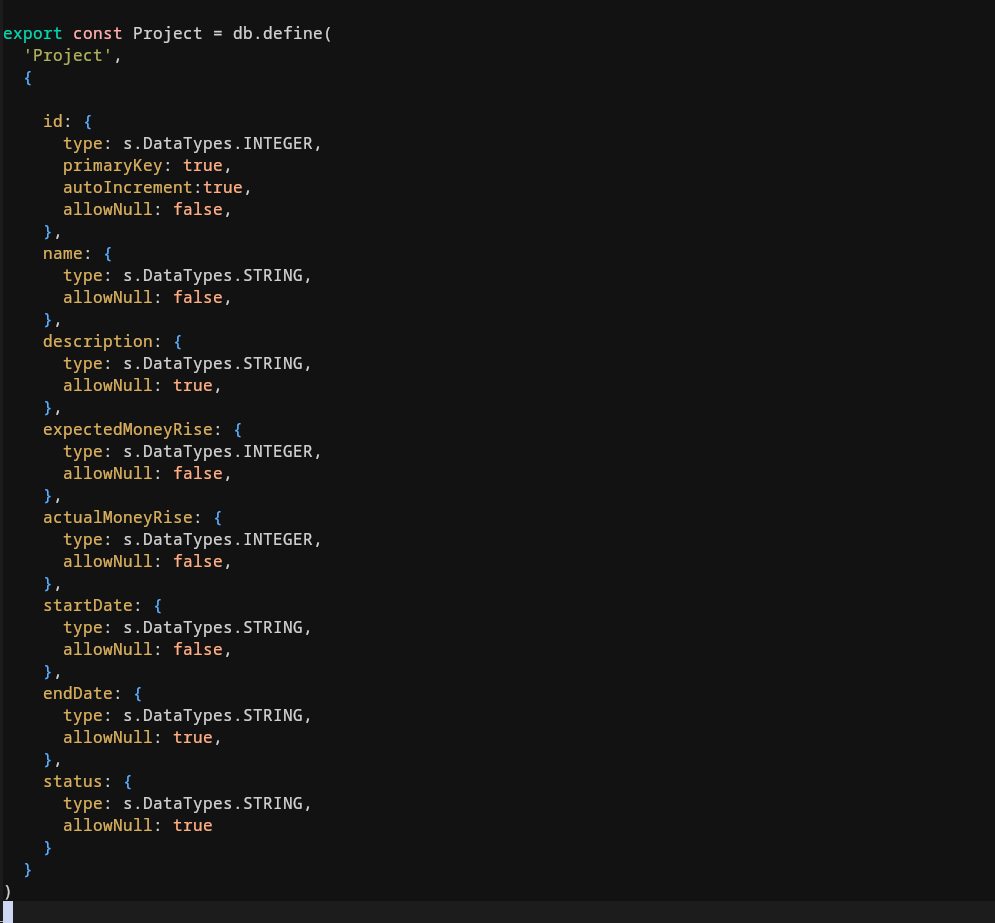
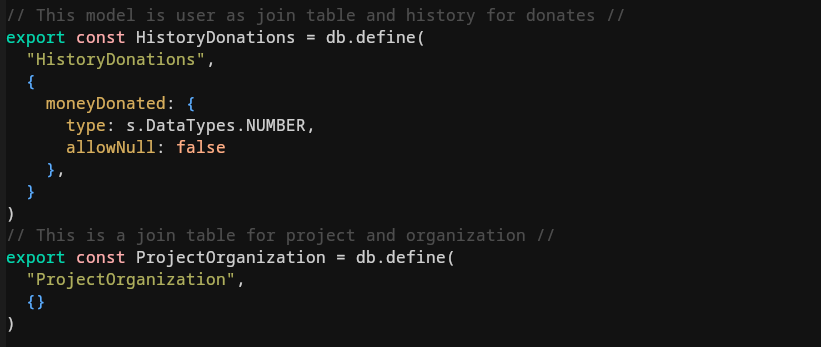
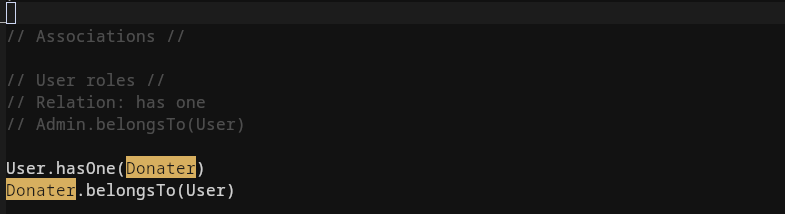


Рисунок 5 - Промежуточный вид формы "RegisterForm"

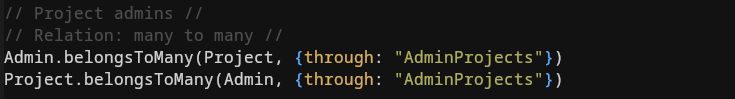
На последнюю таблицу обращу отдельное внимание. Это таблица history\_donations. По своей сути это join таблица между таблицами donater и project. Но она содержит еще одну колонку money. Таким образом она выступает таблицой отчета, кто задонатил, на какой проект и сколько. Причина, по которой в этой таблице нет donaterId и projectId такова, ORM создаст их самостоятельно. По этой причине я не показал создание других join таблиц, потому что там нечего показывать. Они создадутся на этапе установления связей.

Рисунок 6

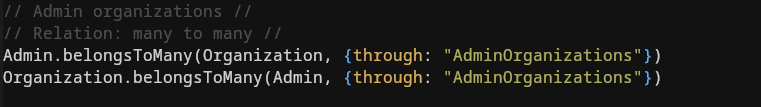
После того, как модели были созданы, я записал связи, опираясь на модель базы данных, показанную в начале этого приложения. Для начала я создал связь между пользователем и донатером. Это связь один к одному.

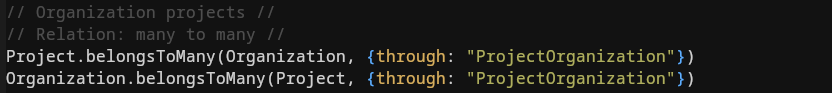
Рисунок 7

Далее создавались связи многие ко многим. Этот тип связи предполагает join таблицу, которую я указал в опциях с помощью поля through. На этом этапе автоматически создадутся join таблица, не объявленные выше.

Рисунок 8

Далее, по аналогии создавались остальные связи.

Рисунок 9

Рисунок 10