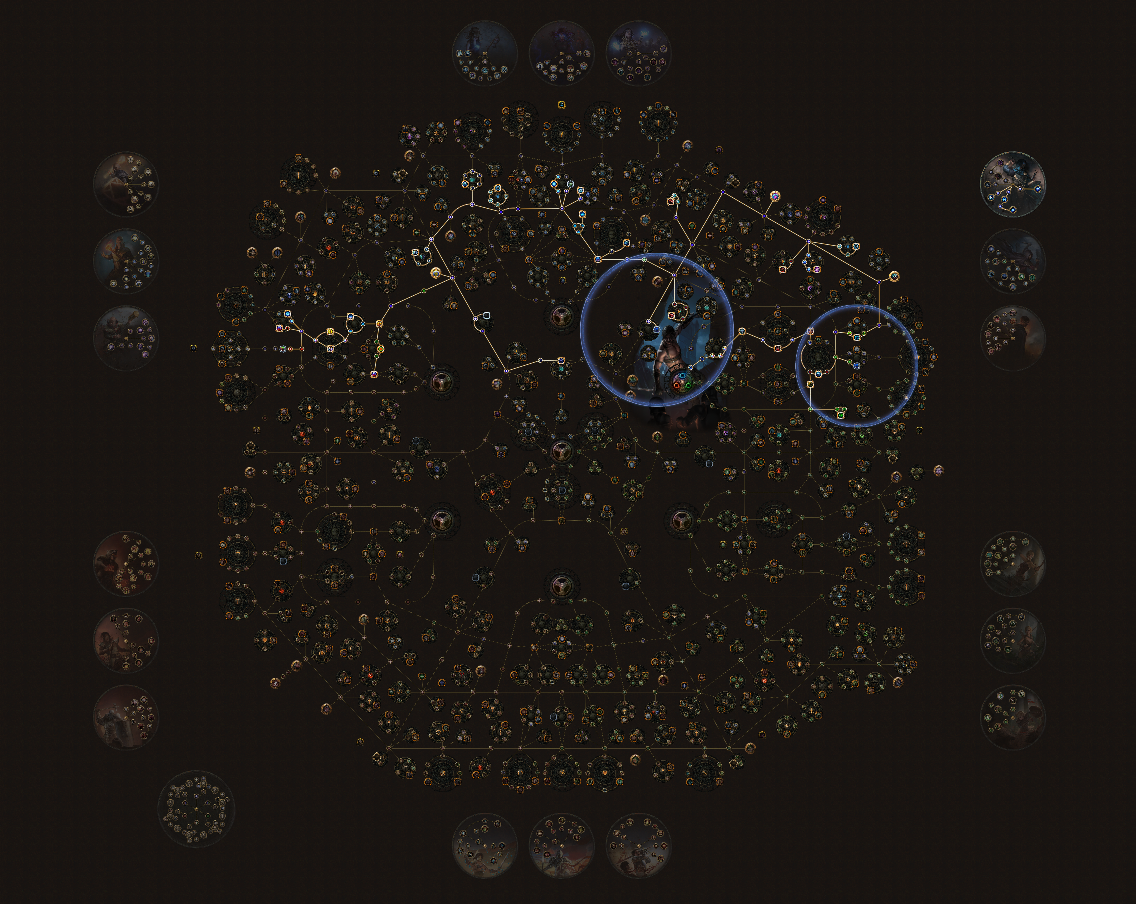
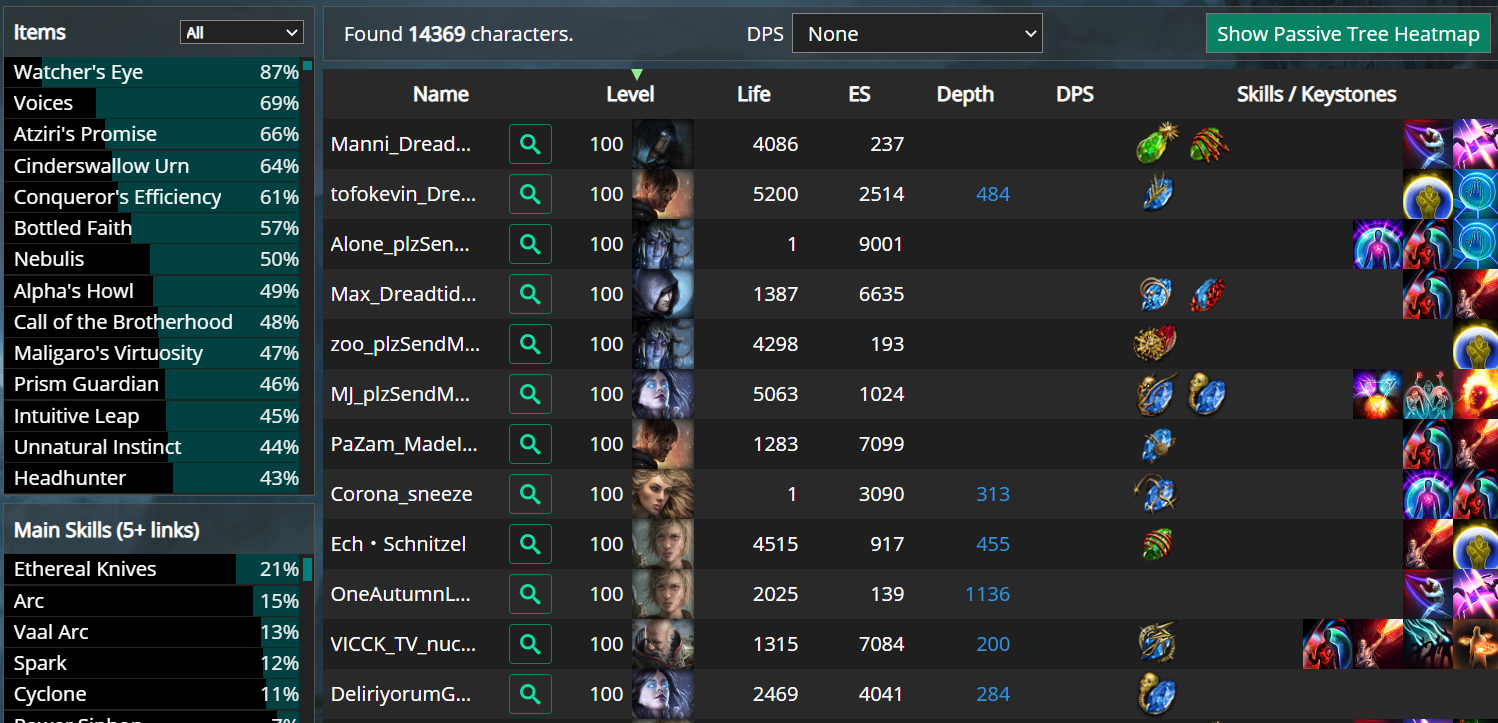
Лабораторная работа №3.

1. Предметная область.

На данный момент Path of exile является самой крупной и активно развивающейся ARPG, о чем можно судить по ее наградам (BAFTA Awards 2020 “Best Evolving Game). Одними из особенностей игры являются довольно большое дерево пассивных умений и комбинация камней умений и камней поддержки, в результате чего мы получаем огромную вариативность билдов и необходимость заранее проектировать своего персонажа (или использовать чужой билд, чем и пользуется большинство игроков). Более того, информация о популярности конкретных классов или билдов особенно полезно в торговле (предметы, необходимые для популярных билдов, как правило, стоят значительно больше). Из-за этого было принято решение разработать базу данных персонажей для Path of Exile.



За основу была взята таблицы с poe.ninja

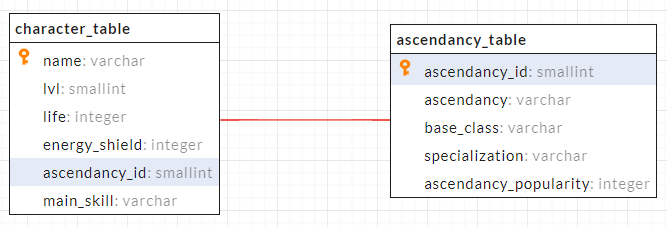
База данных состоит из 2 таблиц.

Ascendancy\_table описывает классы восхождения персонажей и состоит из:

* Ascendancy\_id – id восхождения (ключевое поле)
* Ascendancy – название класса восхождения
* Base\_class – базовый класс персонажа, которому доступно это восхождение (на нем построен индекс)
* Specialization – специализация восхождения
* Ascendancy\_popularity – число персонажей, выбравших это восхождение (изначально 0, изменяется триггером)

Character\_table описывает персонажей:

* Name – имя персонажа (ключевое поле)
* Lvl – (1-100) уровень персонажа
* Life – (>0) hp персонажа
* Energy\_shield – (>=0) энергетический щит персонажа
* Ascendancy\_id – id восхождения, ссылается на ascendancy\_id из ascendancy\_table
* Main\_skill – основной камель умения персонажа

Все поля таблиц не равны null.

1. Нормальная форма

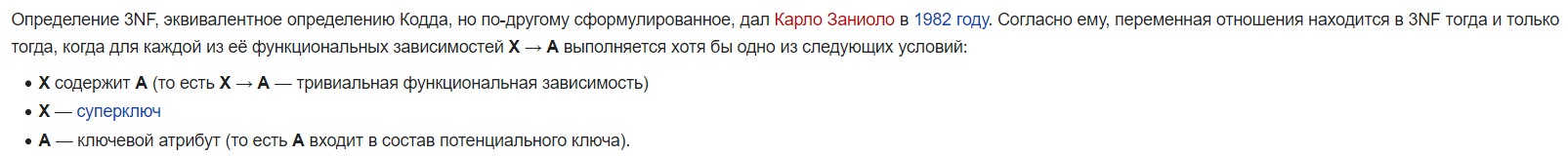
Обе таблицы находятся как минимум в 1 нормальной форме по определению, так как:

* каждый кортеж отношения содержит только одно значение для каждого из атрибутов (значение ни одного столбца нельзя разбить на несколько, сохранив смысл)
* Нет упорядочивания строк сверху вниз (другими словами, порядок строк не несет в себе никакой информации) – смысл обеих таблиц не меняется при перестановке строк.
* Нет упорядочивания столбцов слева направо (другими словами, порядок столбцов не несет в себе никакой информации) – смысл обеих таблиц не меняется при перестановке столбцов.
* Нет повторяющихся строк (в силу уникальности имен персонажей и ascendancy\_id).
* Каждое пересечение строки и столбца содержит ровно одно значение из соответствующего домена (и больше ничего) – в силу реализации вставки и изменения строк.
* Все столбцы являются обычными.

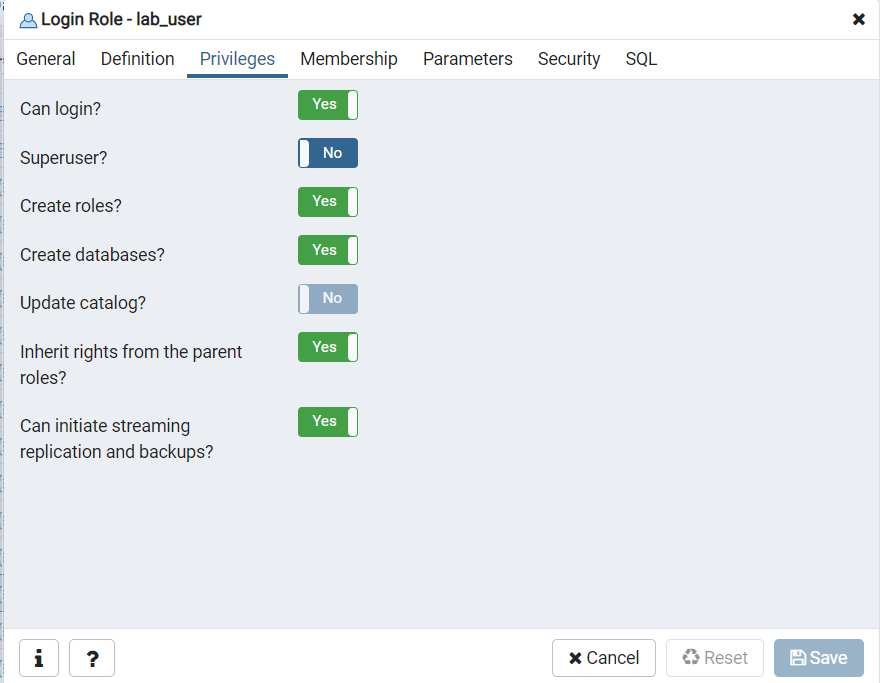
Обе таблицы находятся как минимум в 2 нормальной форме, так как:

* Находятся в 1 нормальной форме.
* Каждый неключевой атрибут неприводимо зависит первичного ключа

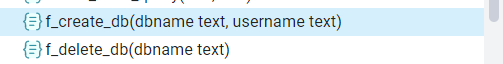
Обе таблицы находятся как минимум в 3 нормальной форме, так как:

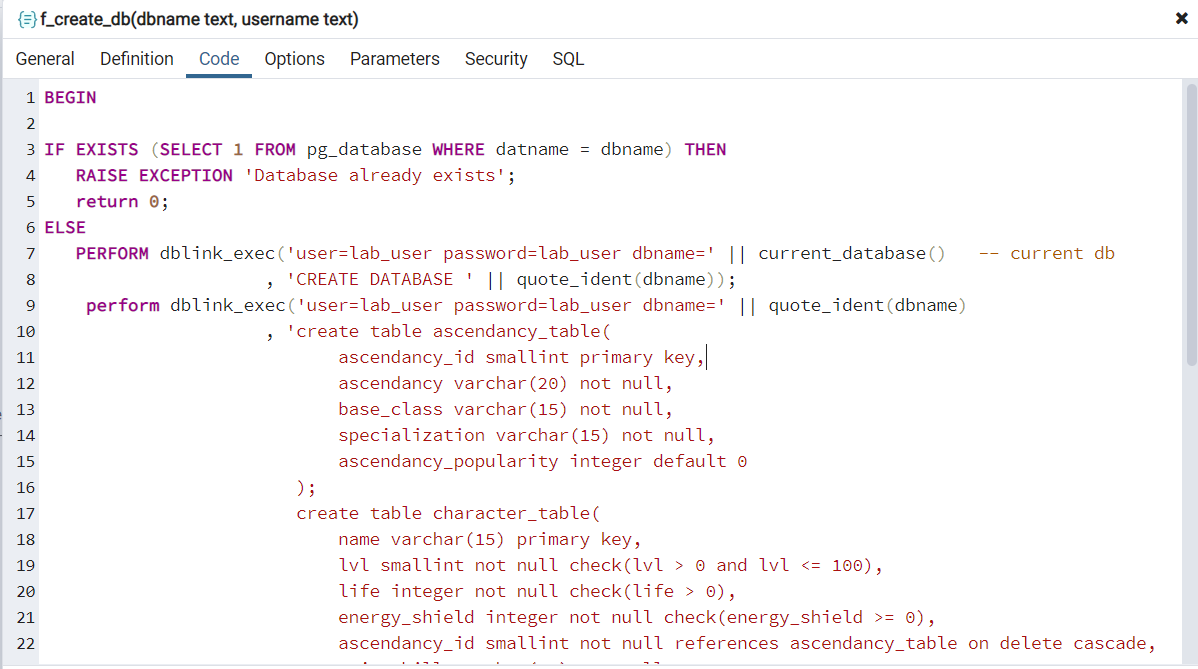
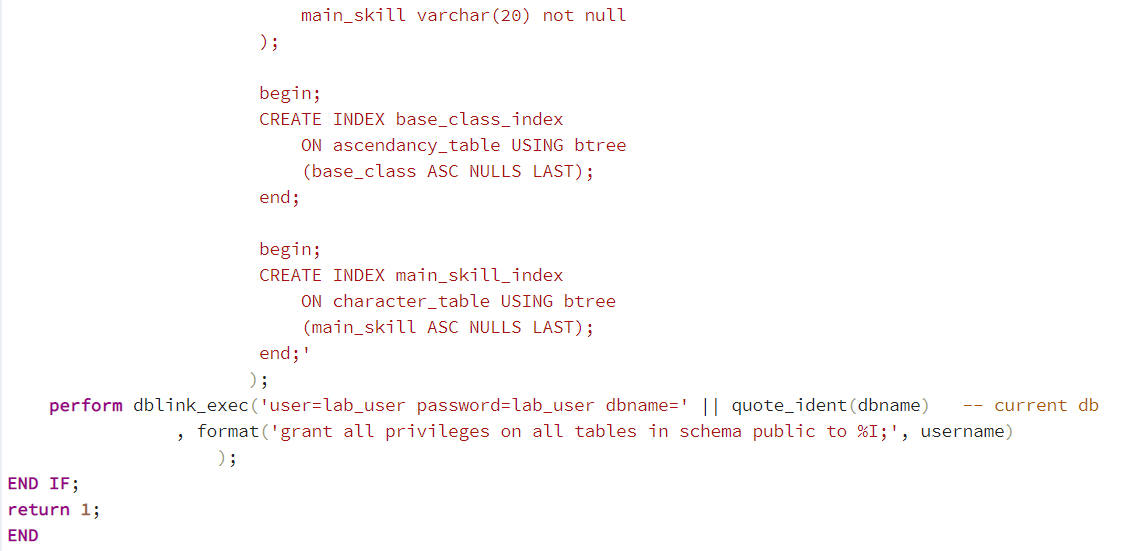
* Находится во 2 нормальной форме
* Выполняется

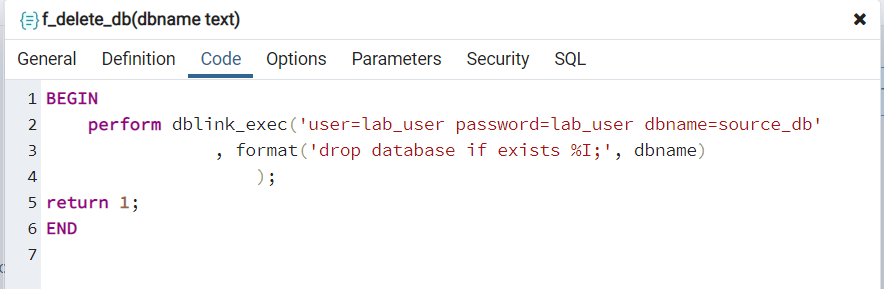
Так как между неключевыми атрибутами нет функциональной зависимости, а для всех функциональных зависимостей справедливо X -> A, где X – первичный ключ, A – неключевой атрибут.

1. Критерии БД
2. Таблицы в 3 нормальной форме
3. БД состоит из 2 таблиц
4. Создание и базы данных и ее удаление выполняются под заранее созданным пользователем, являющимся владельцем базы данных, в которой функции для инициализации новой базы данных (для этого использовался пользователь lab\_user) 

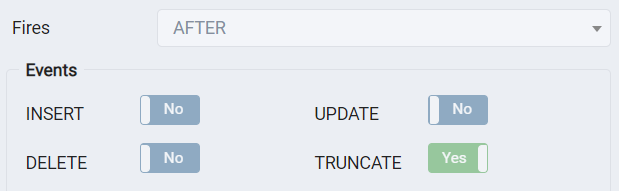
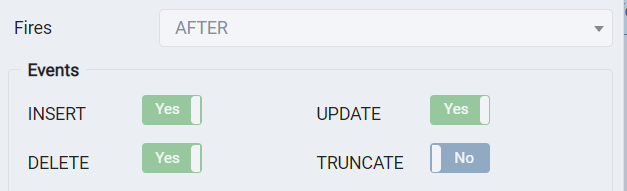
При создании таблицы в гуи требуется ввести логин и пароль пользователя, из-под которого будут выполняться все операции, так как при после инициализации бд этому пользователю выдаются привилегии изменять таблицу, вызывать функции и процедуры и так далее (кроме создания и удаления бд, о чем говорилось выше). Функции и процедуры в бд загружаются скриптом, а таблицы создаются через dblink функции (которые хранятся в заранее созданной дазе данных)



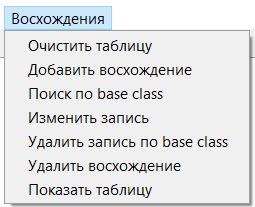
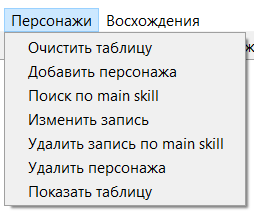
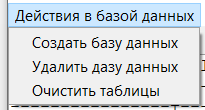
 



1. Индексы созданы на текстовых столбцах base\_class и main\_skill.
2. Поле asncendancy\_popularity изменяется 2 тригерами



1. Реализованные функции.



Ссылка на github: <https://github.com/NoVarlok/poe-database_course>