

1 Инфологические модели и даталогические

Нужно не менее 6-ти сущностей и связь М:М

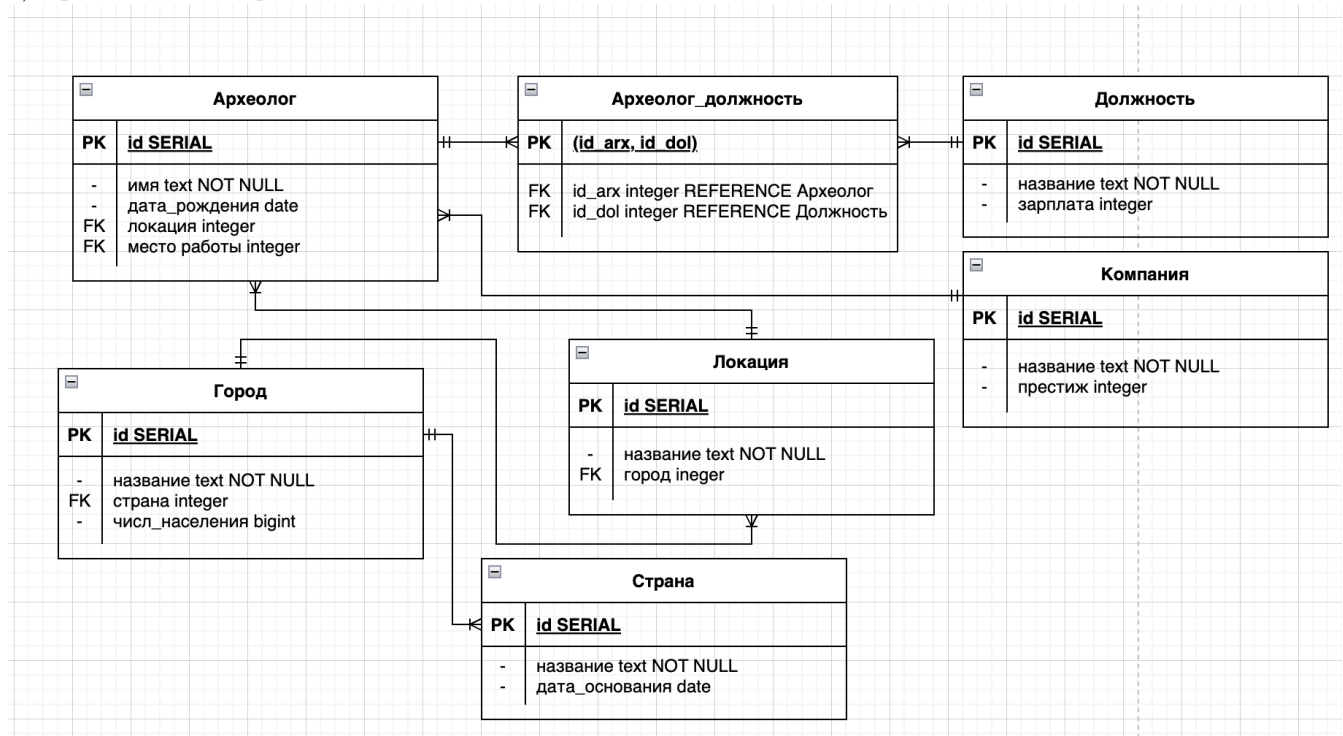
1) Археологические раскопки
археологи(имя, дата рождения, локация, место работы)
должности(название, зарплата)
локации(название, город)
страны(название, дата основания)
города(название, страна, численность населения)
компании(название, престиж)
м:м будет между археологами и должностями

2) Система контроля версий ПО
системы контроля(название, актуальность)
проект(название, дата создания, система контроля)
разработчик(имя, дата рождения, должность)
должности(название, зарплата)
файл(название, расширение, дата создания, местоположение)
ветка разработки(название, проект, создатель)
м:м будет между разработчиками и проектами

3) Системы распределённого доступа
Пользователь(имя, пароль, роль, индикатор)
Роли(название, дата создания, группа доступа)
Ресурсы(название, тип, сервер)
Сервера(название, группа доступа, индикатор)
Индикаторы(название, страна, тип)
Страны(название, население)
м:м будет между Пользователем и Ресурсами

Даталогические модели

1) Археологические раскопки



2 SQL - Запрос

```
SELECT Должность.название  
FROM Археолог  
INNER JOIN Археолог_должность ON Археолог.id = Археолог_должность.id  
INNER JOIN Должность ON Археолог_должность.id = Должность.id  
WHERE (Должность.зарплата > 10000 AND Археолог.имя = 'Иван');
```

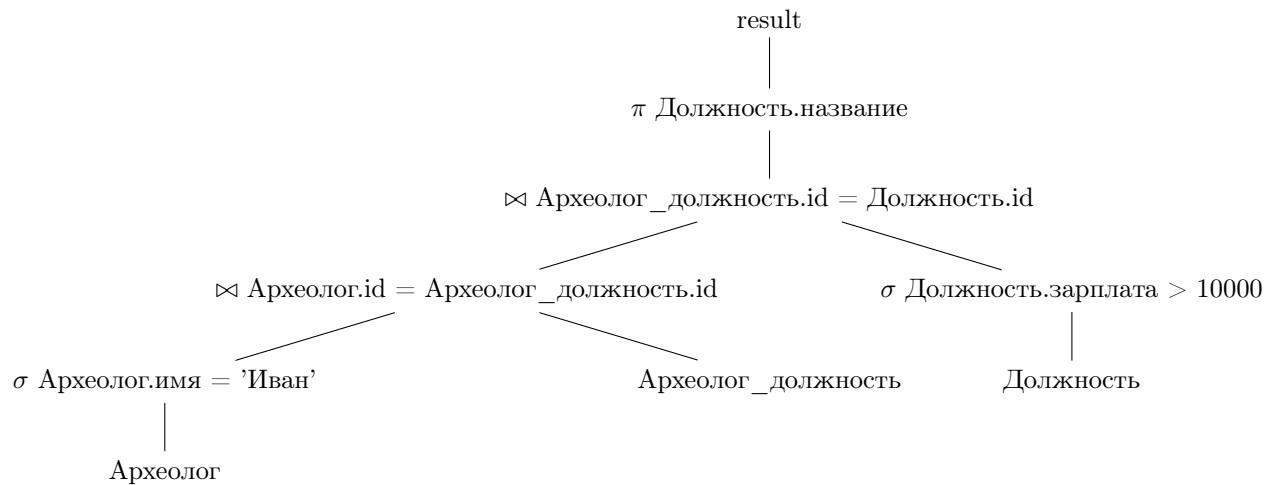
План выполнения

```
CREATE INDEX index_arch ON Археолог (имя) USING HASH;
```

Используем хеш индекс, так как у нас имеется прямое сравнение.

```
CREATE INDEX index_dolj ON Должность USING btree(зарплата);
```

Используем дерево, чтобы облегчить нахождение первого элемента, который будет > 10000, а далее произойдёт выборка бОльших значений.



Этот план выполнения оптимален так как мы достигли цели. На момент соединения строк перебор будет минимальным за счёт отдельных процессов выборки, получилось разделение процессов.