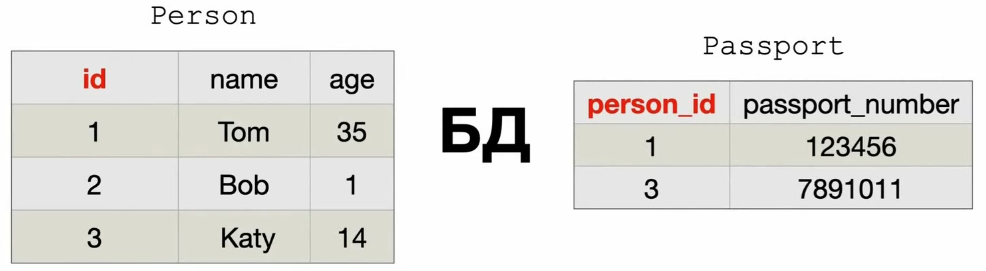
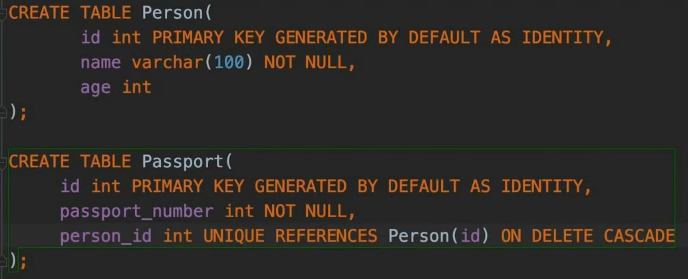
**Отношение One to One**

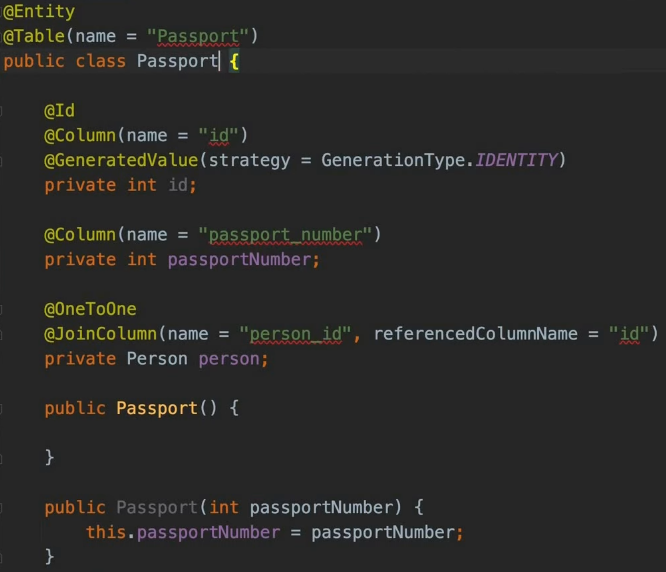
У нас есть таблица людей (**Person**) и таблица паспортов (**Passport**). Между эти двумя таблицами отношение один к одному, тоесть у одного человека может быть только один паспорт и один паспорт может принадлежать только одному человеку.

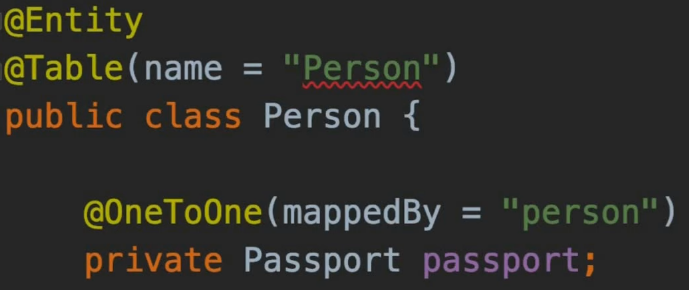
  
**SQL команды для создания таблиц в БД:**



Чтобы выстроить отношение один к одному существуют специальные аннотации:

**@OneToOne** и **@JoinColumn**.

В отличии от отношения Один ко многим здесь нет зеркальной аннотации потому-что обе стороны отношений равноправны, тоесть у нас как паспорт имеет одного человека так и человек имеет один паспорт.

  
  
  
  
  
  
  
Здесь в классе **Passport** использвуем аннотацию **@OneToOne** и используем аннотацию **@JoinColumn**, чтобы обозначить колонку которая является внешним ключом (в нашем случае колонка с внешним ключом одновременно является колонкой для первичного ключа).

На родительской сущности (в классе **Person**) мы используем так же аннотацию **@OneToOne** и используем аргумент **mappedBy**, чтобы указать на название того поля которое в сущности **Passpost** выстраивает связь один к одному.

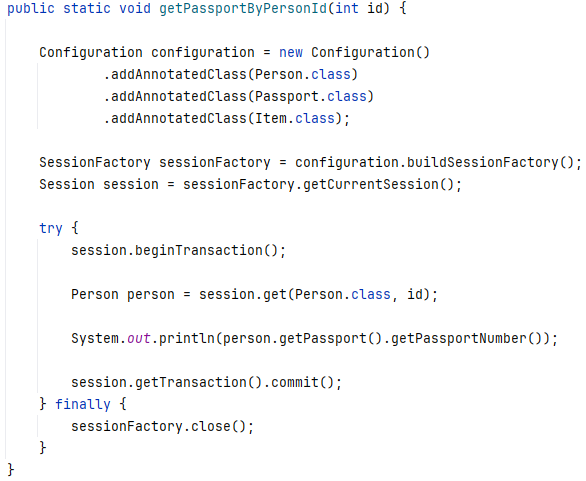
Сущность **Passport** является **owning side** так как у неё есть аннотация **@JoinColumn**.

**owning side –** это та сущность в которой не используется аннотация с аргументом **mappedBy**.

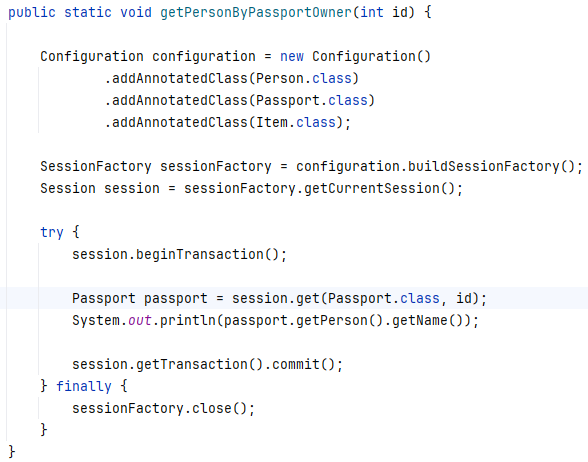
**Примеры:  
1.** Добавление нового человека и нового пасспорта.



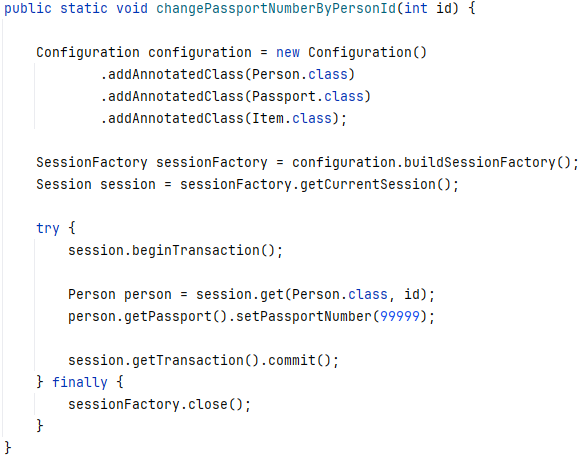
**2.** Получение паспорта по ID человека.



**3.** Получение человека по по владельцу паспорта.



**4.** Изменене номера паспорта у человека.



**5.** Удаление челвоека.

