



УНИВЕРСИТЕТ  
искусственного интеллекта

```
mirror_mod.use_y = True  
mirror_mod.use_z = False  
elif _operation == "MIRROR_Z":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = False  
    mirror_mod.use_z = True
```

```
#selection at the end -add back the deselected mirror modifier  
mirror_ob.select= 1  
modifier_ob.select=1  
bpy.context.scene.objects.active = modifier_ob  
print("Selected" + str(modifier_ob)) # modifier ob is the active ob  
#mirror_ob.select = 0  
#me = bpy.context.selected_objects[0]  
#me.data.objects[me.name].select = 1
```

# СХЕМА ДАННЫХ

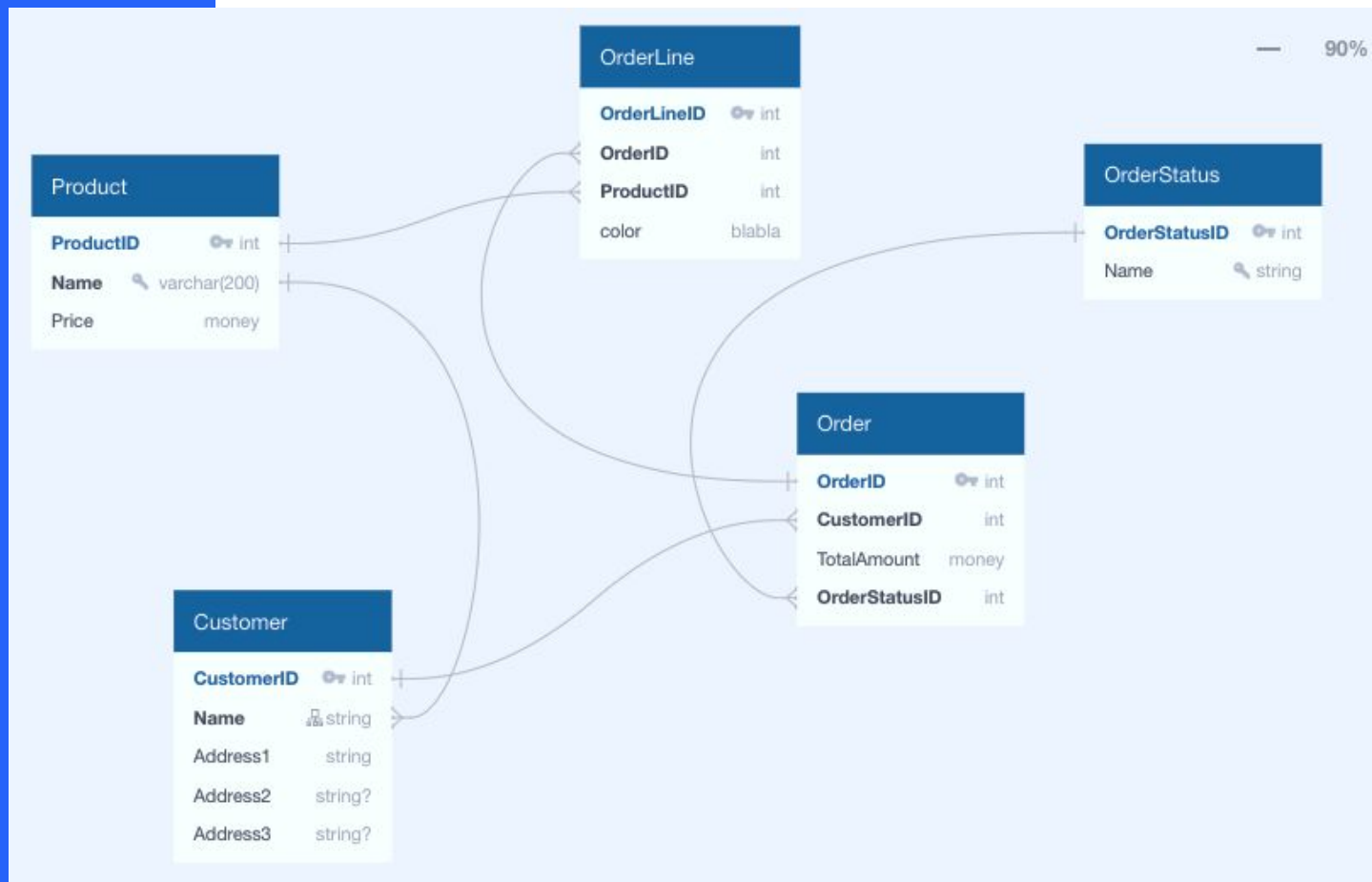
# ПЛАН

- Схема Данных
- Связи: one2many
- Связи: many2many
- Связи: one2one
- Пример проектирования схемы





# Схема Данных



**Схема данных** (схема БД) —  
Описывает структуру базы  
данных или отдельного  
отношения, включает в себя  
описание типов данных  
(атрибутов), ограничений и  
основных ключей.

Схема так же описывает  
взаимодействие отношений  
в БД (таблиц) друг с другом.

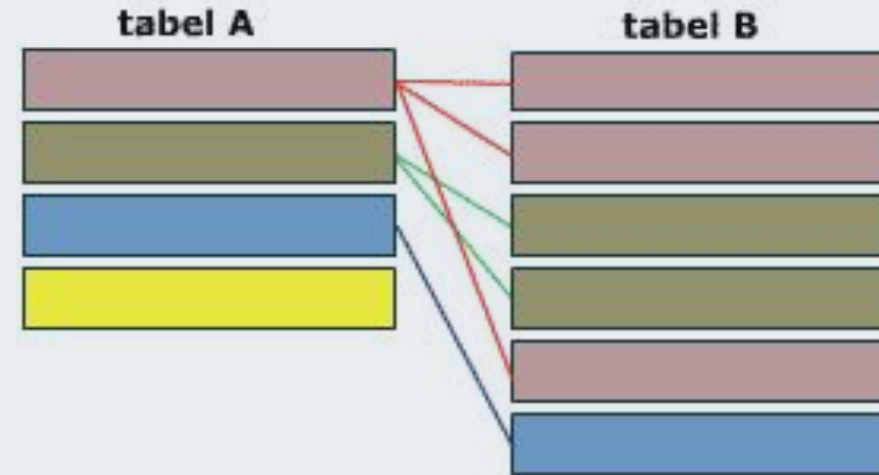
# Связи: one2many

**Одна** запись в таблице **A** может быть связана с **0, 1** или **множеством записей** в таблице B.

**Чтобы опознать ответьте на 2 вопроса**

- 1) Сколько объектов в B могут относиться к объекту A?
- 2) Сколько объектов из A могут относиться к объекту из B?

Один ко Многим (one2many)  
на первый вопрос ответ – **множество**,  
на второй – **один** (или ни одного)

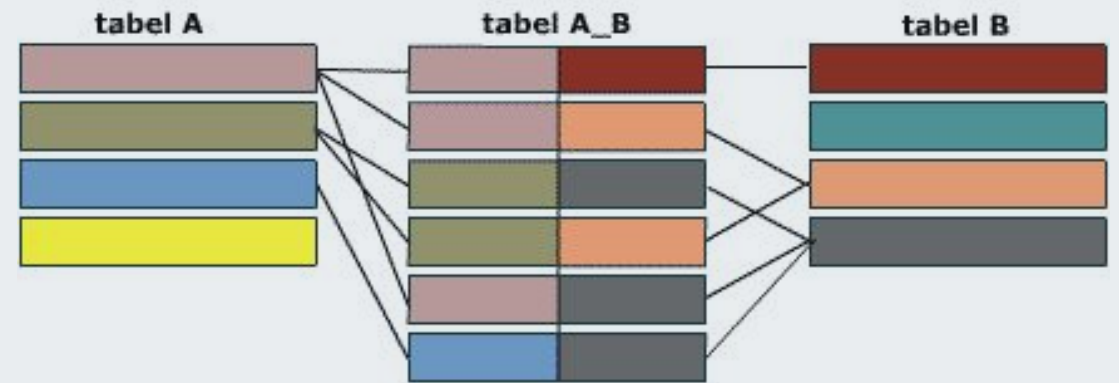


**Примеры:**

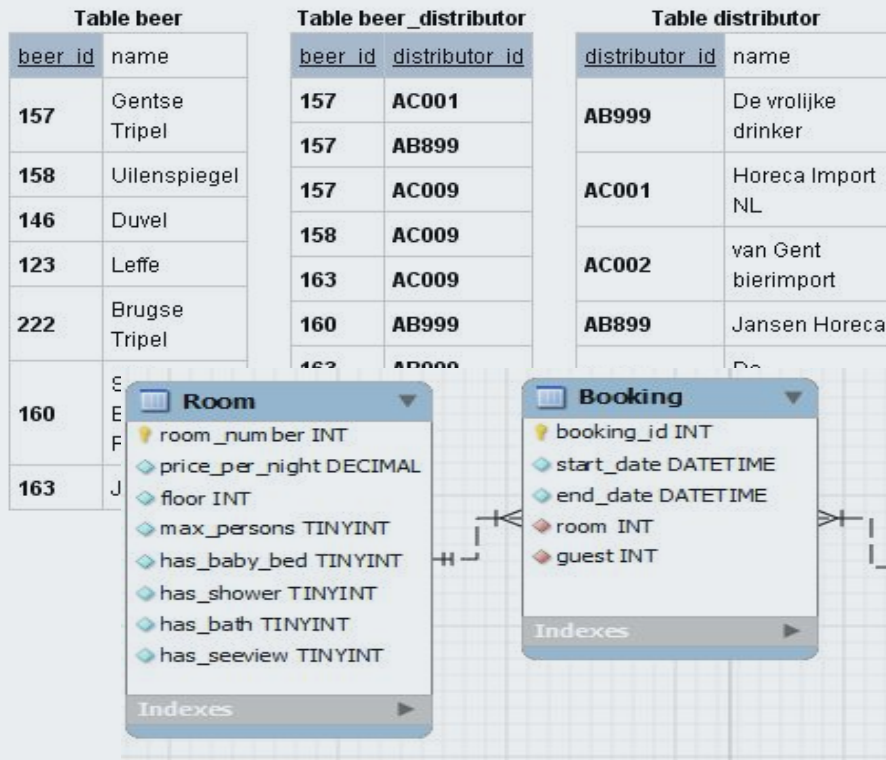
- Машина и ее части. Каждая часть машины одновременно принадлежит только одной машине, но машина может иметь множество частей.
- Дома и улицы. На улице может быть несколько домов, но каждый дом принадлежит только одной улице.

# Связи: many2many

связь, при которой **множественным записям** из одной таблицы (A) могут соответствовать **множественные записи** из другой (B).

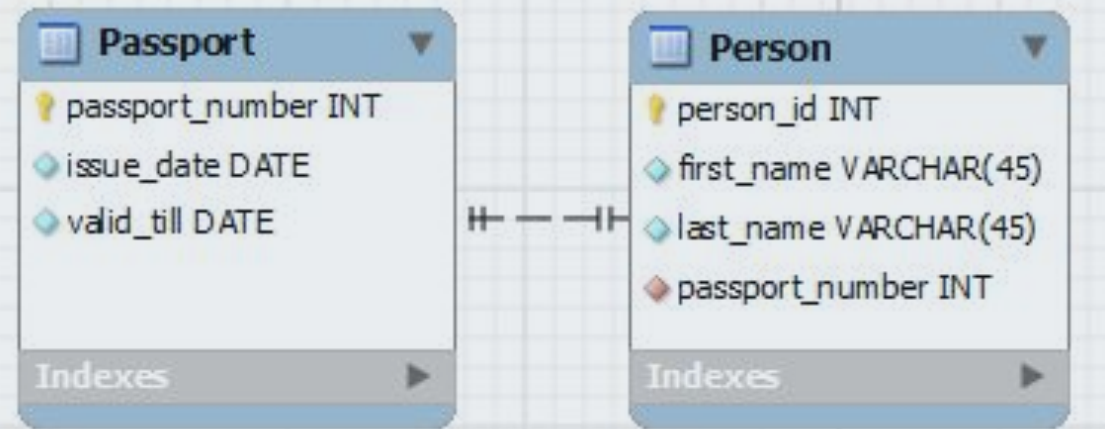
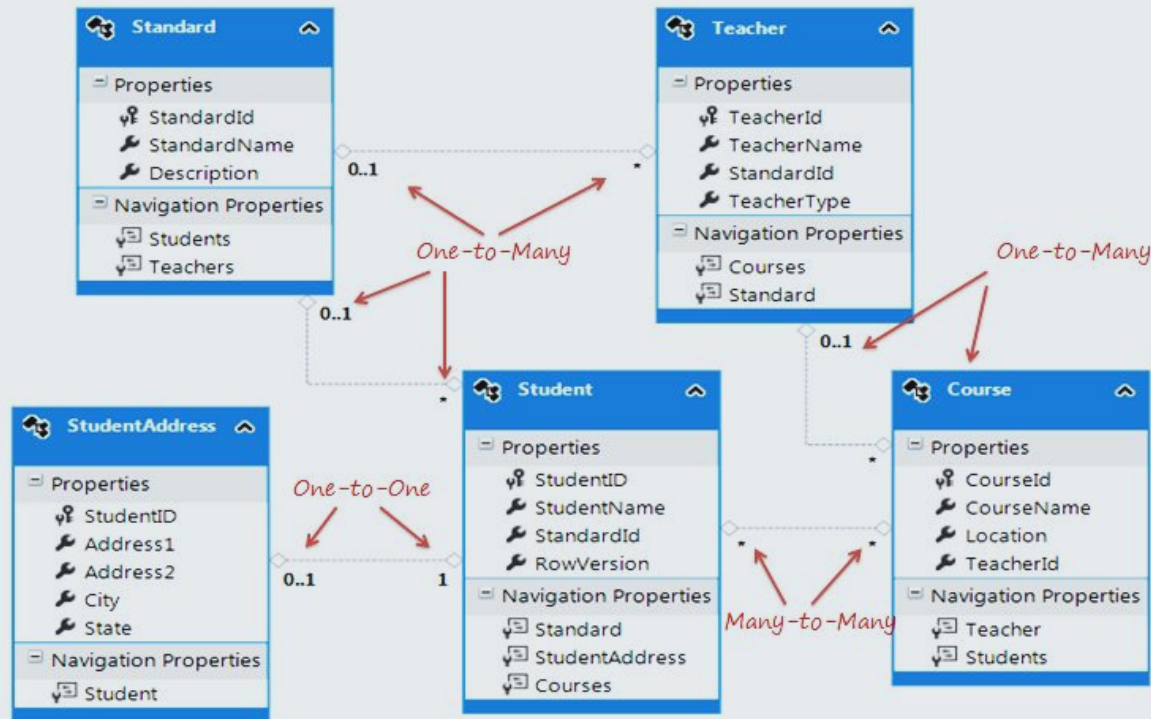


Две таблицы – “источника” и одна соединительная таблица. Первичный ключ соединительной таблицы A\_B – **составной**. Она состоит из двух полей, двух внешних ключей, которые ссылаются на первичные ключи таблиц A и B. Все первичные ключи должны быть уникальными. Это подразумевает и то, что комбинация полей A и B должна быть уникальной в таблице A\_B



# Связи: one2one

связь, при которой **каждый** блок сущности А может быть ассоциирован с **0 или 1** блоком сущности В.



## Пример:

Люди и их паспорта. Каждый человек в стране имеет только один действующий паспорт и каждый паспорт принадлежит только одному человеку.



# ПРИМЕР ПРОЕКТИРОВАНИЯ СХЕМЫ

Проектирование схемы данных универсального магазина

Основные события в интернет магазине:

- Отображение товаров
- Классификация товаров
- Регистрация клиентов
- Добавление товаров в корзину покупок
- Отображение содержимого корзины покупок
- Оформление заказов посетителями

# ПРИМЕР ПРОЕКТИРОВАНИЯ СХЕМЫ 2

Определяем структуру БД (Отношения и типы их связей)

Определяем сущности типы связей и набор атрибутов:

Товары, товары, клиенты и заказы – сущности

заказом и товаром = many2many – Типы связей

клиенты и заказы = one2many – Типы связей

