

## Task 1

Так как каждая строка в таблице обязана быть уникальной, то приходим к тому, что отношение – это набор уникальных в совокупности атрибутов.

Пусть ключа нет, тогда это утверждение равносильно тому, что нет такого набора атрибутов, чтобы они являлись уникальными в совокупности (то есть такими, по которым можно однозначно идентифицировать наблюдение). Приходим к противоречию с первым утверждением о том, что каждое наблюдение обязано быть уникальным.

Значит, каждое отношение (множество атрибутов, которые в совокупности должны быть уникальны) имеет ключ. Ключ также может быть и составной (из нескольких атрибутов).

## Task 2

### Task2.1

**Book**(ISBN:string, name:string, year:int, author:string, number\_of\_pages:int, publisher\_name:string)

*\*ISBN – однозначно определяет книгу*

**Copy**(number:int, shelf\_position:int, book\_isbn:int)

*\*считаем, что номер копии не определяет копию. А если взять вместе с ISBN, то определяется однозначно экземпляр*

**Publisher**(name:string, address:string)

*\*Издательство определяется именем*

**Category**(name:string, super\_category\_name:string)

*\*так как иерархия, то имя категории однозначно определяет родителя*

**Book\_Category\_belongs** (category\_name:string, ISBN:int)

*\*не знаю точно, как лучше назвать, но суть в том, что эта таблица показывает связь между категорией и книгой. Любая пара значений category\_name, ISBN может храниться там (нельзя подчеркнуть один ключ, так как тогда что-то второе должно определяться однозначно, а это не так)*

**Reader**(reader\_id:int, first\_name:string, last\_name:string, address:string, birthday:date)

*\*читатель определяется уникальным id*

**Booking**(reader\_id:int, copy\_num:int, isbn\_of\_copy:int, return\_date:date)

*\*это тоже связь многое ко многим между копией и читателем. Пара читатель – копия определяют однозначно дату. То есть, взятие книги определяется*

## Task 2.2

1)

**Страна**({название:string})

**Город**({название:string, название\_страны:string})

**Улица**({название:string, название\_города:string, название\_страны:string })

**Дом**({номер:int, название\_улицы:string, название\_города:string, название\_страны:string })

**Квартира**({номер:int, номер\_дома:int, название\_улицы:string, название\_города:string, название\_страны:string })

*\*Будем считать, что для двух разных стран города могут повторяться, и для двух разных городов – улицы могут повторяться. Получаем: город определяется названием и страной, улица – городом (который определяется названием и страной) и тд.*

2)

**Команда**({название:string })

**Арбитр**({имя:string})

**Матч**({название\_команды1:string, название\_команды2:string, имя\_арбитра:string})

*\*В этом случае без матча не обойтись, если мы считаем, что каждая команда может сыграть с другой командой несколько раз, и при этом может быть сменен арбитр. Все эти условия определяют, что ключ должен быть 3-х частей (к1, к2, арбитр)*

*Логично, что если команды должны переиграть (то есть не меняются ключи название\_команды1, название\_команды2, то арбитр должен быть уже другой)*

3)

**Женщина**({id:int, id\_матери:int, id\_отца:int})

**Мужчина**({id:int, id\_матери:int, id\_отца:int})

*\*Можно считать, что каждый человек имеет id матери и отца, но он не обязан иметь ребенка.*

## Task 2.3

**Атрибут**({название:string, ключевой\_или\_нет:bool, имя\_сущности:int})

*\*Ключ составной, так как два одинаковых атрибута могут быть у разных сущностей.*

**Сущность**({название:string})

**Связь**({название:string, тип:string, название сущности1:string, название сущности2:string, роль:string})

*\*То есть связь определяется двумя сущностями*

## Task 3

### Task3.1

**City**({region:string, name:string})

**Station**({name:string, city\_region:string, city\_name:string, #tracks:int })

*\*Станция определяется ее именем (считаем, что в разных городах станции все равно не повторяются, иначе – надо было бы сделать составной ключ)*

**Train**({TrainNr:int, length:int, start\_station\_name:string, end\_station\_name:string})

**Connected**({prev\_station\_name:string, trainNr:int, arrival:time, departure:time, next\_station\_name:string})

*\*Считаем, что любой поезд один раз уезжает с конкретной станции (было бы логично так делать), поэтому пара prev\_station\_name и trainNr будет ключом (так же можно было взять пару trainNr и next\_station).*

### Task 3.2

**Station**({statNr:int, name:string})

**Room**({roomNr:int, #Beds:int, stationNr:int})

*\*Комната должна определяться не только ее номером, но и местом.*

**Patient**({patientNr:int, name:string, disease:string, PersNr:int, roomNr:int})

**Admission**({patientNr:int, roomNr:int, from:string, to:string})

*\*Для каждого admission только 1 пациент, поэтому он и ключ.*

**StationPersonell**({PersNr:int, #Name:string, stationNr:int})

**Caregiver**({persNr:int, qualification:string, Name:string, StatNr:int})

**Doctor**({persNr:int, area:string, rank:string, Name:string, StatNr:int })