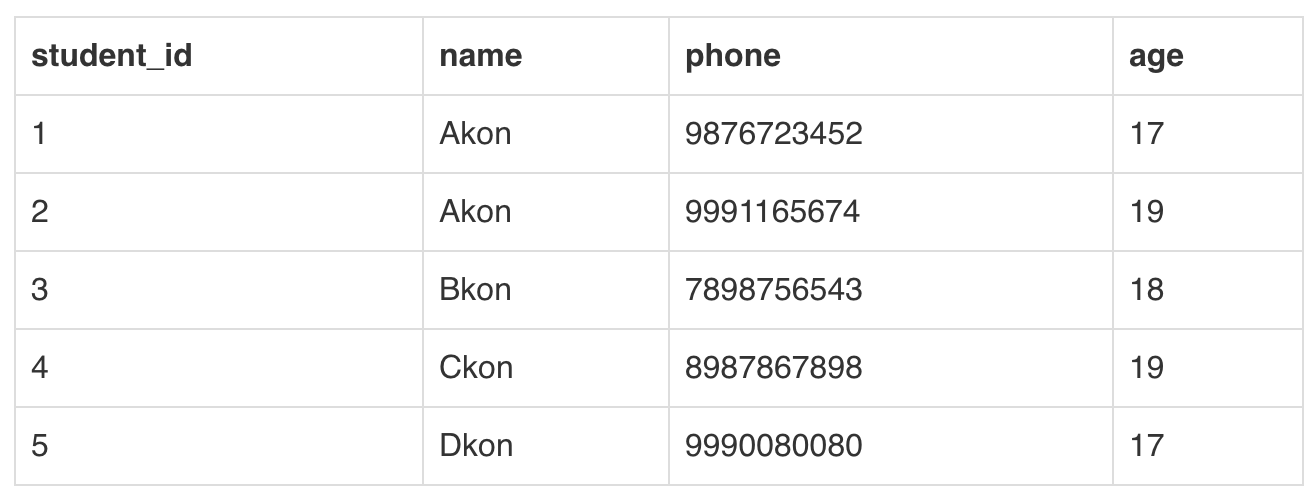
Нормализация таблиц

Определения



**Суперключ** (superkey) - подмножество атрибутов (столбцов), которые уникально идентифицируют каждую строку таблицы.

Пример.

* {student\_id}
* {student\_id, name}
* {student\_id, phone}
* {student\_id, age}
* {student\_id, name, phone}
* {student\_id, name, age}
* {student\_id, name, phone, age}
* {student\_id, phone, age}
* {student\_id, age}
* {phone}
* {phone, name}
* {phone,age}
* {phone, name, age}

**Потенциальный ключ** (candidate key) - минимальное подмножество атрибутов таблицы, уникально идентифицирующих строуку.

Т.е. потенциальный ключ – суперключ минимальной длины.

**Первичный ключ** (primary key) – один из потенциальных ключей, выбранный в качестве основного ключа

**Составной ключ** – ключ, состоящий из нескольких атрибутов

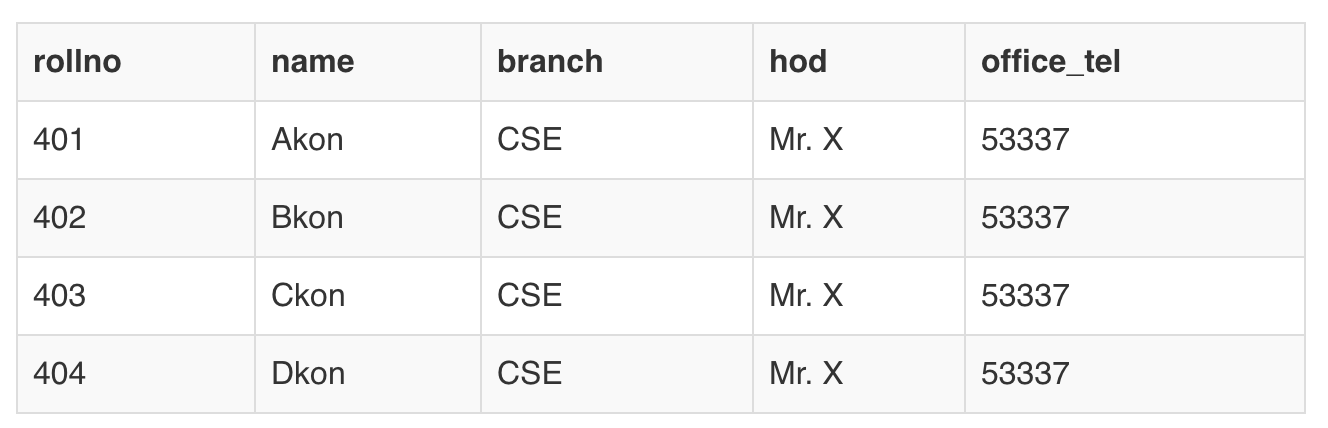
**Неключевой атрибут** – атриубут таблицы, не входящий в состав потенциального ключа.

Нормализация – процесс преобразования таблицы БД к нормальным формам.

Нормальная форма – это свойство таблицы БД, характеризующее его с точки зрения избыточности, потенциально приводящей к логическим ошибках выборки и изменения данных.

**Проблемы хранения данных в ненормализованных таблицах**

**Student**



**Аномалия вставки**

Данные о студенте не могут быть вставлены до выбора специальности.

(либо соответствующие атрибуты имеют NULL значение, если не заданы ограничения)

**Аномалия изменения**

При необходимости изменения данных об одном преподавателе необходимо вносить изменения во множество строк таблицы. При ошибочном пропуске одной из строк возникнет противоречие в данных.

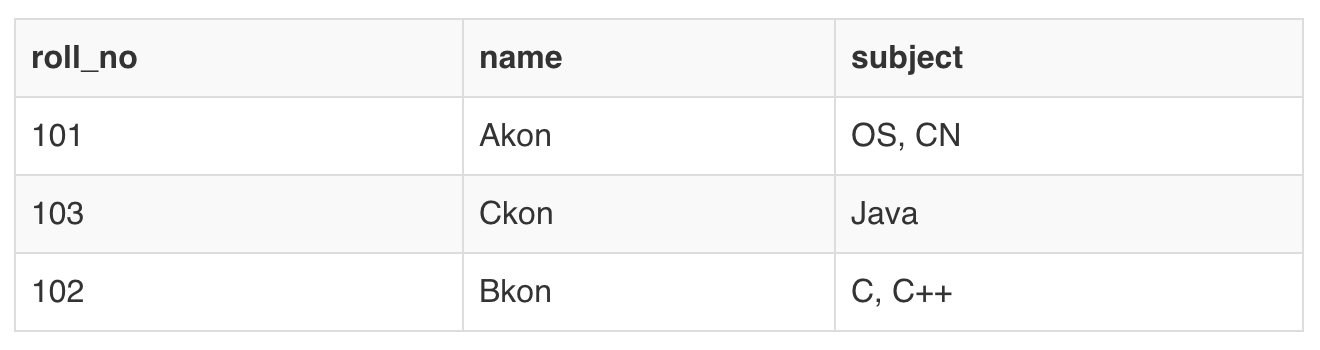
**Аномалия удаления**

Таблица Student хранит информацию о студенте и направлении. При удалении информации о студенте, может исчезнуть информация о направлении.

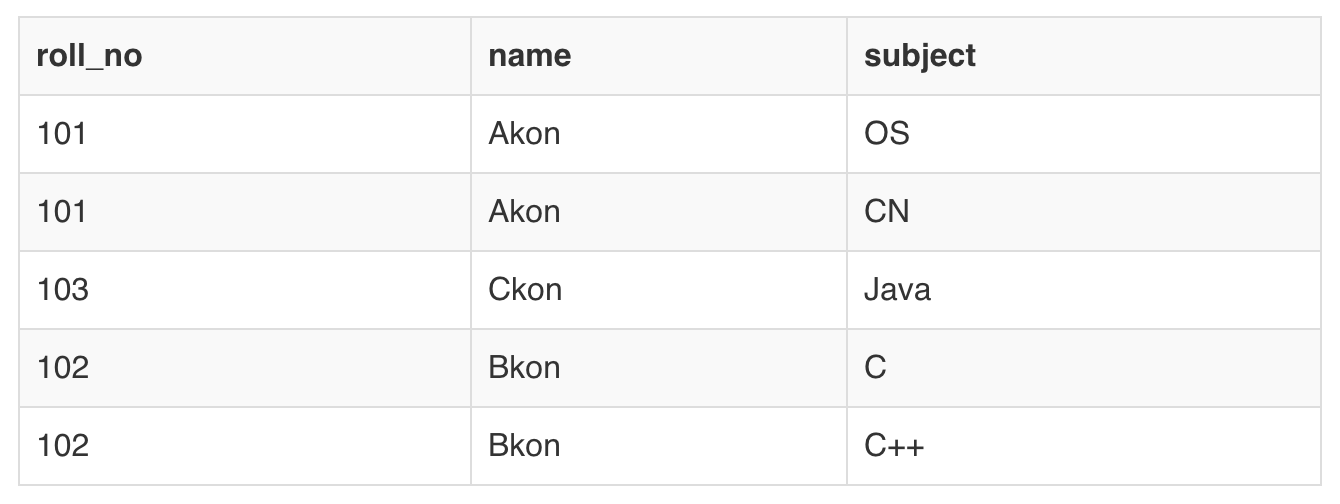
**Первая нормальная форма**

Таблица находится в 1НФ если и только если

* значения атрибутов атомарны
* *значения в строках принадлжат одному домену (типу)*
* *порядок строк не имеет значения*
* *нет атрибутов с одним именем*



=>



**Вторая нормальня форма**

Если даны два атрибута *X* и *Y* некоторого отношения, то говорят, что***Y* функционально зависит от *X***, если в любой момент времени каждому значению *X* соответствует ровно одно значение *Y*.

Обозначается:

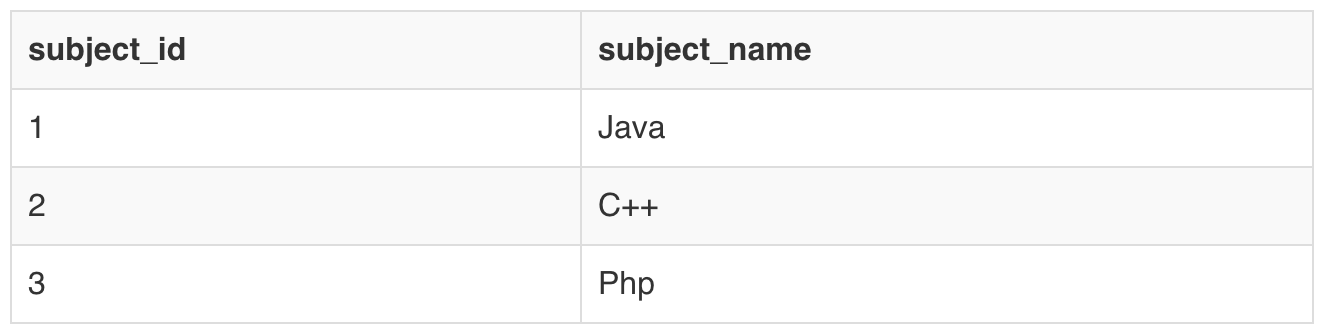
X->Y



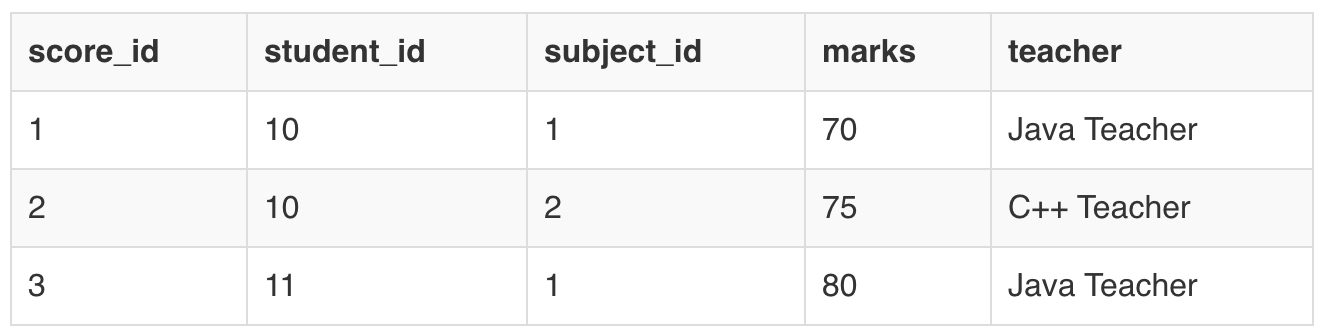
ФЗ: {student\_id}-> {name, reg\_no,branch,address}

Неключевой атрибут **функционально полн**о зависит от составного ключа если он функционально зависит от всего ключа в целом, но не находится в функциональной зависимости от какого-либо из входящих в него атрибутов.

Subjects



Scores



Потенциальные ключи Scores:

score\_id,

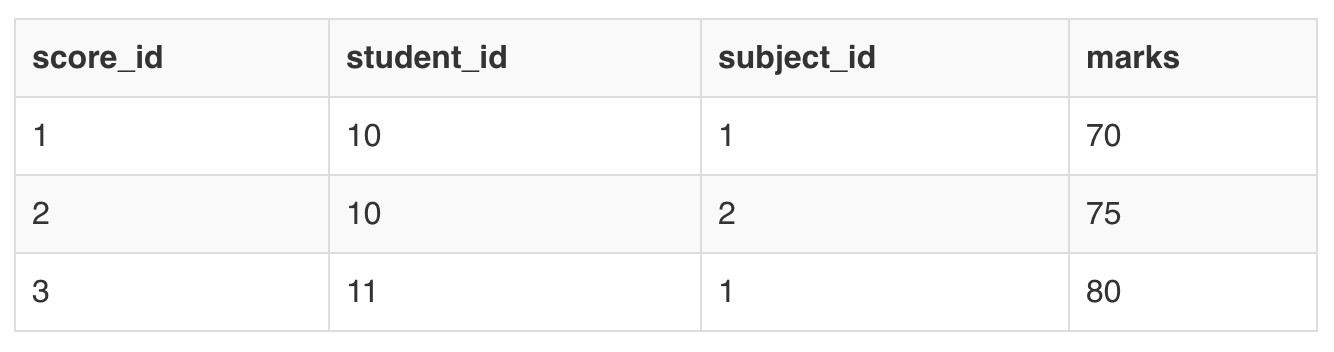
{student\_id, subject\_id}

ФЗ {student\_id, subject\_id} -> {teacher} не является функционально полной

Отношение находится во 2НФ тогда и только тогда, когда

* находится в 1НФ
* каждый неключевой атрибут функционально полно (неприводимо) зависит от потенциального ключа.

Пример.

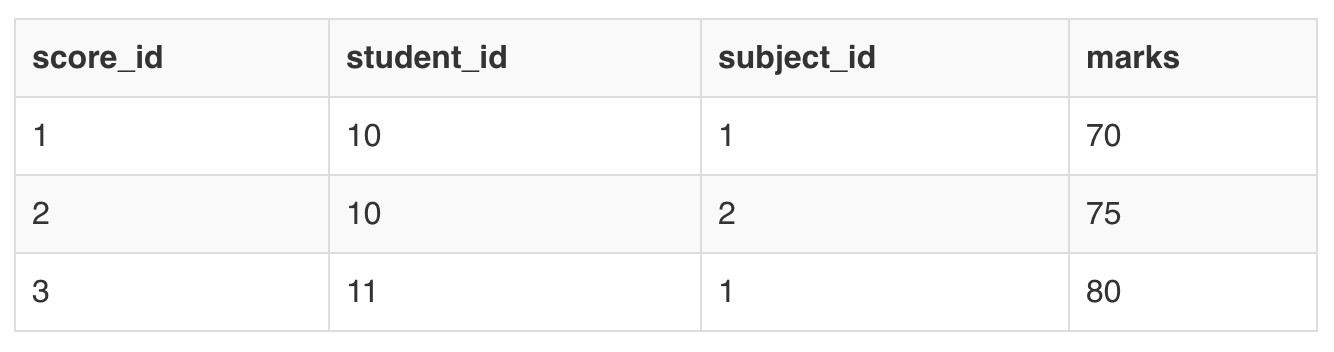


Для приведения в 2НФ произведено вынесение атрибута в другую таблицу.

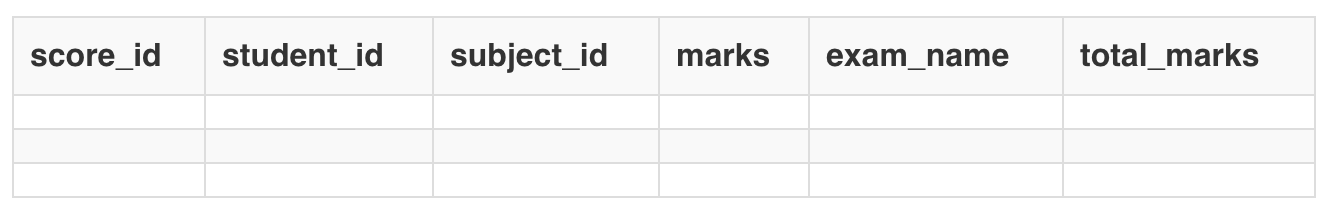
**Третья нормальная форма**

Пусть X, Y, Z - три атрибута некоторого отношения. При этом **X --> Y** **и Y --> Z**, но обратное  
соответствие отсутствует, т.е. **Z -/-> Y** и **Y -/-> X**. Тогда  Z транзитивно зависит от X.

Score



Score\_cf



exam\_name – форма контроля (зачет/экзамен)

total\_marks - шкала оценивания (5 или “зачет”)

ФЗ

{student\_id, subject\_id} -> {exam\_name}

{exam\_name}->{total\_marks}

{exam\_name} -/->{student\_id, subject\_id}

{student\_id, subject\_i d} -> {total\_marks} – транзитивная функциональная зависимость

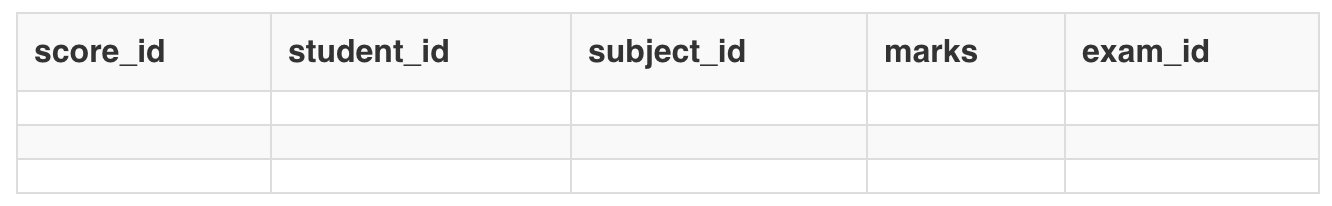
**Есть ФЗ между неключевыми атрибутами.**

Отношение находится в 3 НФ тогда и только тогда, когда

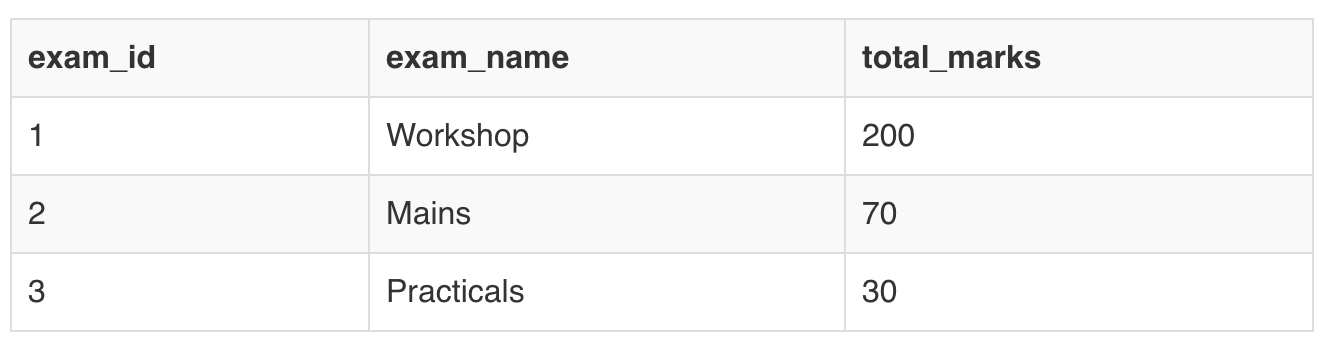
* Отношение находится в 2НФ
* ни один неключевой атрибут R не находится в транзитивной функциональной зависимости от потенциального ключа.

(Т.е. каждый неключевой атрибут нетранзитивно зависит от потенциального ключа)

Score



Exam



Вынесение {exam\_name} и {total\_marks} в отдельную таблицу.

**Нормальная форма Бойса-Кодда**

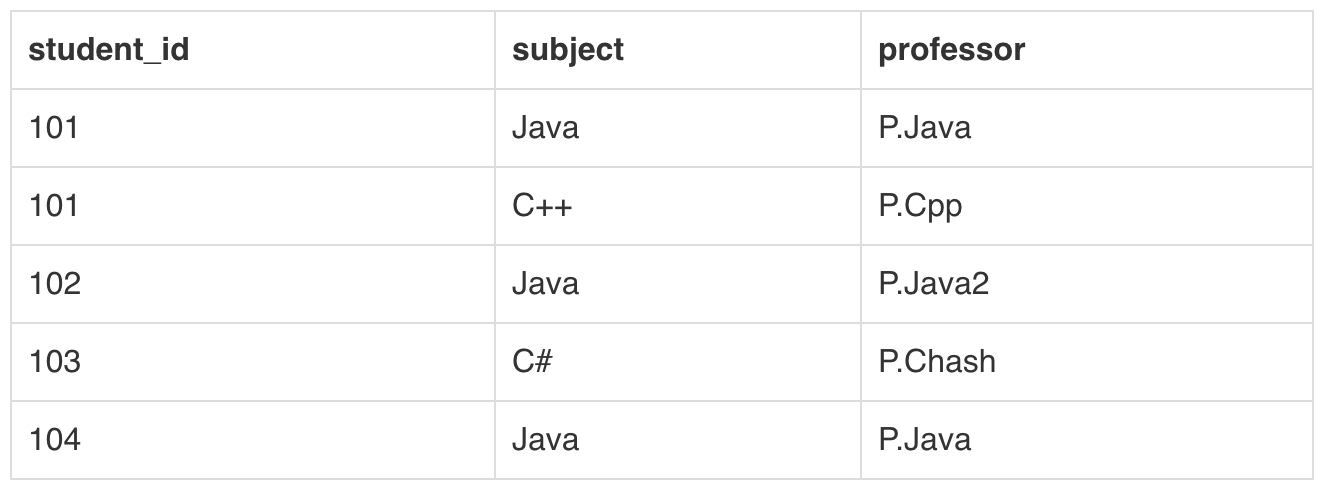
Тривиальная ФЗ - правая (зависимая) часть является подмножеством её левой части (детерминанта).

{student\_id, subject\_id}->{subject\_id}

Неприводимая ФЗ - ни один атрибут не может быть опущен из её детерминанта без нарушения зависимости (детерминант неизбыточен)

{student\_id, subject\_id, marks}->{exam\_id}

Enrollment



Бизнес-правила:

* Студент может быть записан на несколько курсов
* Каждый курс может читать несколько преподавателей
* Преподаватель читает один курс

Потенциальный ключ:

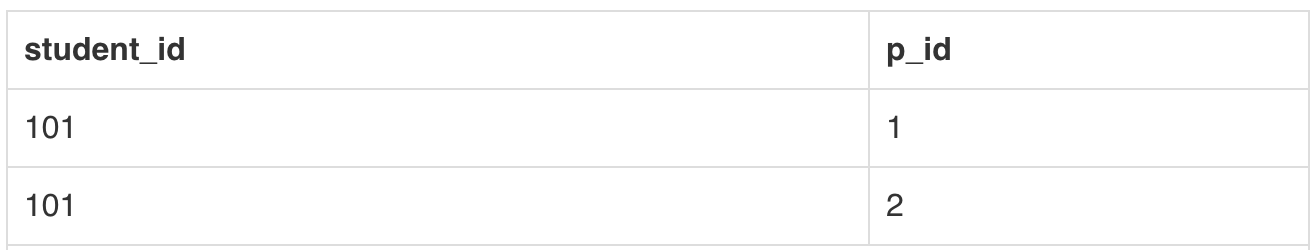
{student\_id,subject}

ФЗ

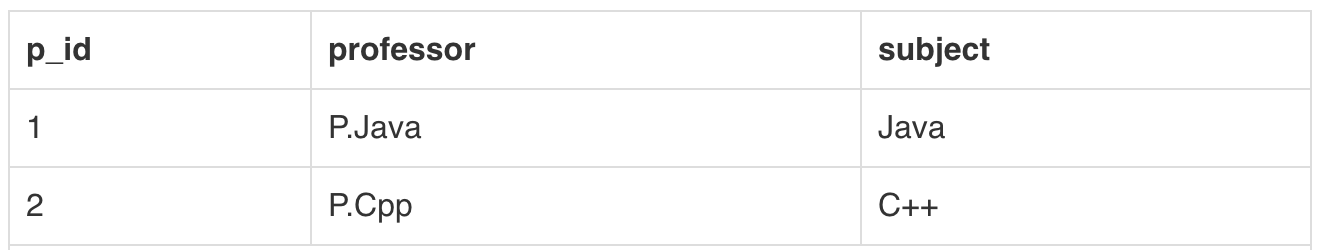
{professor}->{subject}

Переменная отношения находится в НФБК тогда и только тогда, когда каждая её нетривиальная и неприводимая слева функциональная зависимость имеет **в качестве** своего **детерминанта** некоторый **потенциальный ключ**.

Student



Professor



Декомпозиция.