

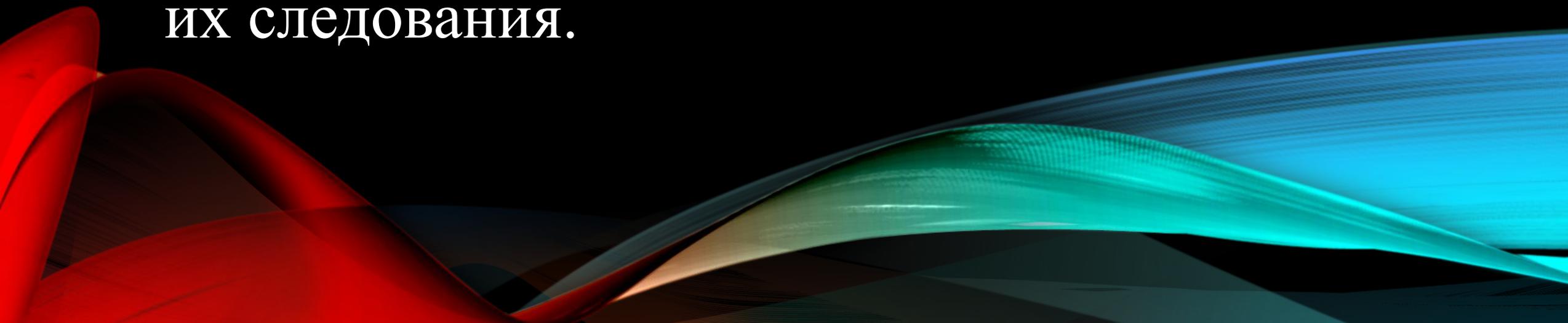
# ЛЕКЦИЯ 6

## Реляционная модель данных

# Реляционное отношение

*Реляционная база данных* — это конечный (ограниченный) набор отношений. Отношение используется для представления объектов, а также для представления связей между объектами. Отношение — это двумерная таблица, имеющая уникальное имя и состоящая из строк и столбцов, где строки соответствуют записям, а столбцы — атрибутам. Каждая строка в таблице представляет некоторый объект реального мира или соотношения между объектами.

*Атрибут* — это поименованный столбец отношения. Свойства объекта, его характеристики определяются значениями атрибутов. Порядок следования атрибутов не влияет на само отношение, оно имеет один и тот же смысл при любом порядке их следования.



Пусть имеется отношение  $r$ . Схемой отношения  $r$  называется конечное множество имен атрибутов  $\{A_1, A_2, \dots, A_n\}$ . Заголовки столбцов отношения содержат имена его атрибутов и, следовательно, все вместе отражают его схему.

Схема отношения ПРЕПОДАВАТЕЛЬ может быть представлена так:

**Таб ном преп, Фамилия, Должность**

## ПРЕПОДАВАТЕЛЬ

Таб\_ном\_преп

Фамилия

Должность

Тогда заголовок отношения ПРЕПОДАВАТЕЛЬ примет вид:

Таб\_ном\_преп

Фамилия

Должность

Отношение строится с учетом ряда факторов. Каждому имени атрибута  $A_i$ ,  $1 \leq i \leq n$  ставится в соответствие множество допустимых для соответствующего столбца значений. Это множество  $D_i$  называется *доменом* данного имени атрибута. В самом общем виде домен определяется заданием некоторого базового типа данных, к которому относятся элементы домена, и произвольного логического выражения, применяемого к элементу типа данных. Если вычисление этого логического выражения дает результат «истина», то элемент данных является элементом домена.

Отношения изображаются в виде таблиц, где имена атрибутов выносятся в шапку таблицы. Каждая строка отношения является множеством значений, взятых по одному из домена каждого имени атрибута. Домены являются произвольными непустыми конечными или счетными множествами и образуют множество:

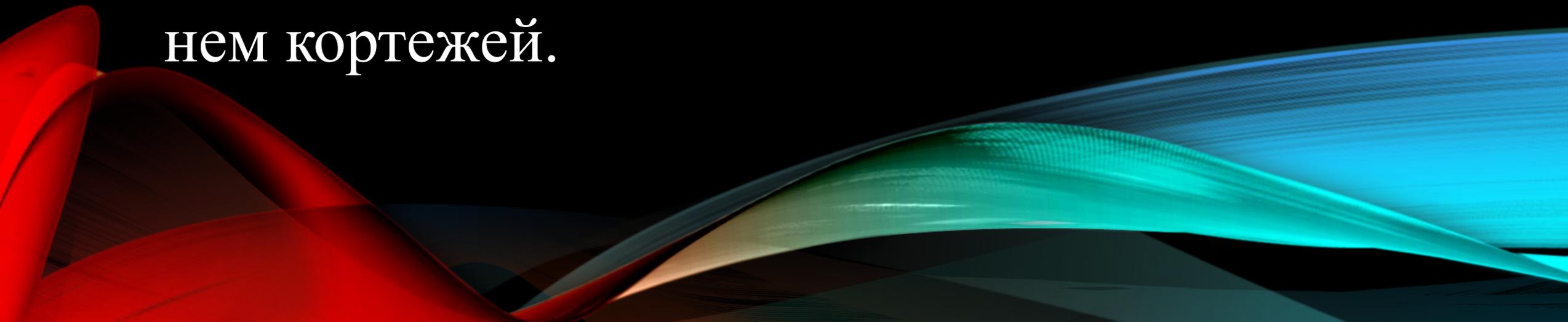
$$\mathbf{D} = \mathbf{D}_1 \cup \mathbf{D}_2 \cup \dots \cup \mathbf{D}_n.$$

Отношение  $r$  со схемой  $R$  - это конечное множество отображений  $\{t_1, t_2, \dots, t_p\}$  из  $R$  в  $D$ . Причем каждое отображение  $t \in r$  должно удовлетворять следующему ограничению:

$t(A_i)$ , принадлежит  $D_i$ ,  $1 \leq i \leq n$ .

Эти отображения называются *кортежами*. Каждый кортеж отношения отображает экземпляр объекта, а атрибут отношения отображает атрибут объекта. Кортежи могут располагаться в любом порядке, при этом отношение будет оставаться тем же самым, а значит иметь тот же самый смысл.

Множество кортежей называются *телом отношения*. Тело отношения отражает состояние объекта, поэтому во времени оно постоянно меняется. Тело отношения характеризуется *кардинальным числом*, которое равно количеству содержащихся в нем кортежей.



Домены

Первичный ключ

Имя  
отношения

СОТРУДНИК

Отношение

Р#	ФИО	Должность	Стаж
Р1	Иванов С.И.	Заведующий	20
Р2	Петров М.Т.	Инженер	5
Р3	Сидоров Р.А.	Инженер	7
Р4	Чистов К.Л.	Лаборант	1

кортежи

Атрибуты  
степени

Одной из главных характеристик отношения является его степень. *Степень отношения* определяется количеством атрибутов, которое в нем присутствует.

В данном отношении есть строка заголовков столбцов и тело отношения. В строке заголовков определены четыре атрибута, следовательно, степень отношения равна ?. Атрибуты имеют свои уникальные имена: Р#, ФИО, Должность, Стаж. Каждый атрибут определен на своем домене.

# СВОЙСТВА И ВИДЫ ОТНОШЕНИЙ

- × Отношение имеет имя, которое отличается от имен всех других отношений.
- × Отношение представляется в виде табличной структуры. Имя таблицы соответствует имени отношения, имена столбцов — именам атрибутов, а строки таблицы — кортежам.
- × Каждый атрибут имеет уникальное имя, его значения берутся из одного и того же домена.

- ✘ Каждый компонент кортежа является простым, атомарным значением, не состоящим из группы значений. Это не позволяет заменять значение атрибута другим отношением, что привело бы к сетевому или иерархическому отношению.
- ✘ Упорядочение атрибутов теоретически несущественно, однако оно может влиять на эффективность доступа к кортежам.
- ✘ Все строки (кортежи) должны быть различны.
- ✘ Теоретически порядок следования кортежей не имеет значения, но этот порядок влияет на эффективность доступа к кортежам.

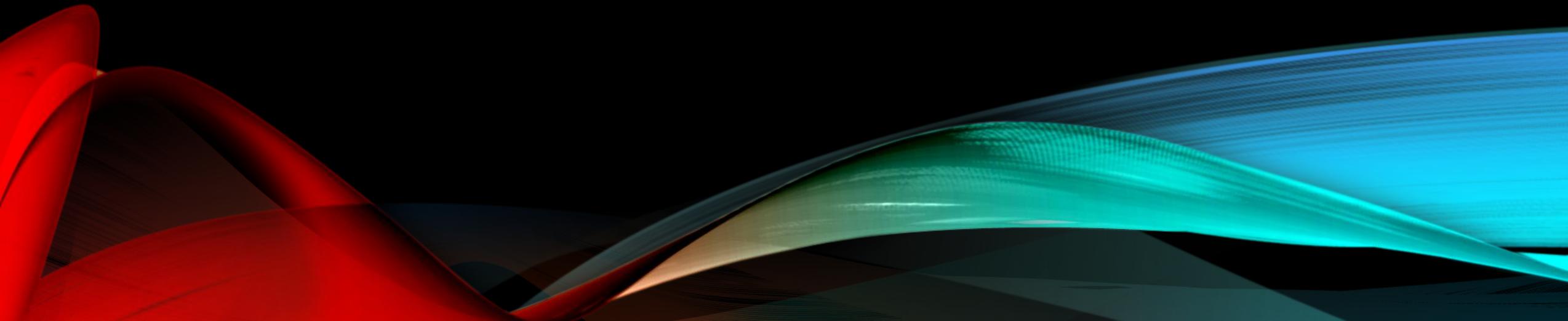
Особое место в ряду свойств занимает требование того, чтобы каждый атрибут имел только простые, атомарные значения. Иными словами, необходимо строить такие отношения, у которых в каждом кортеже каждый атрибут может иметь только единственное значение. Отношение, удовлетворяющее этому условию, называется *нормализованным*. Таким образом, с точки зрения реляционной модели отношение всегда нормализовано. Этим оно отличается от математического отношения, которое вовсе не обязано быть нормализованным.

В реляционной теории встречаются несколько видов отношений, но не все они поддерживаются реальными системами. Различают:

- × именованное отношение – это переменная отношения, определенная в СУБД посредством специальных операторов;
- × базовое отношение – это именованное отношение, являющееся частью базы данных;

- × производное отношение — это отношение, определенное посредством реляционного выражения через базовые отношения;
- × представление — это именованное виртуальное производное отношение, представленное в системе исключительно через определение в терминах других именованных отношений;
- × снимки — это отношения, подобные представлениям, но они сохраняются, доступны для чтения и периодически обновляются;

- × результат запроса — это неименованное производное отношение, получаемое в результате запроса, которое для сохранения необходимо преобразовать в именованное отношение;
- × хранимое отношение — это отношение, которое поддерживается в физической памяти.



# РЕЛЯЦИОННЫЕ КЛЮЧИ

В отношении могут существовать несколько одиночных или составных атрибутов, которые однозначно идентифицируют кортеж отношения. Это - *суперключи*.

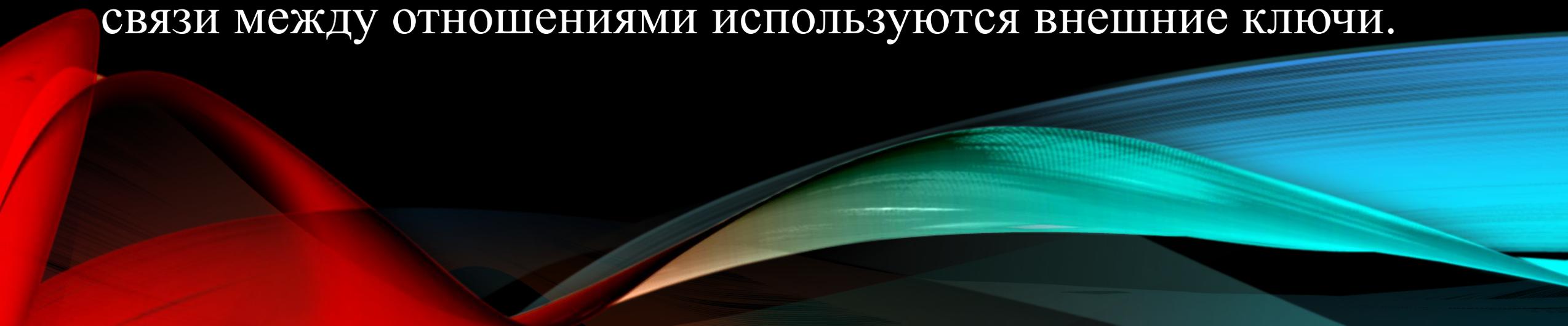


В особую категорию необходимо выделить *потенциальные ключи*. Говорят, что множество атрибутов  $K=(A_i, A_j, \dots, A_k)$  отношения  $R$  является потенциальным ключом  $R$  тогда и только тогда, когда удовлетворяются два независимых от времени условия:

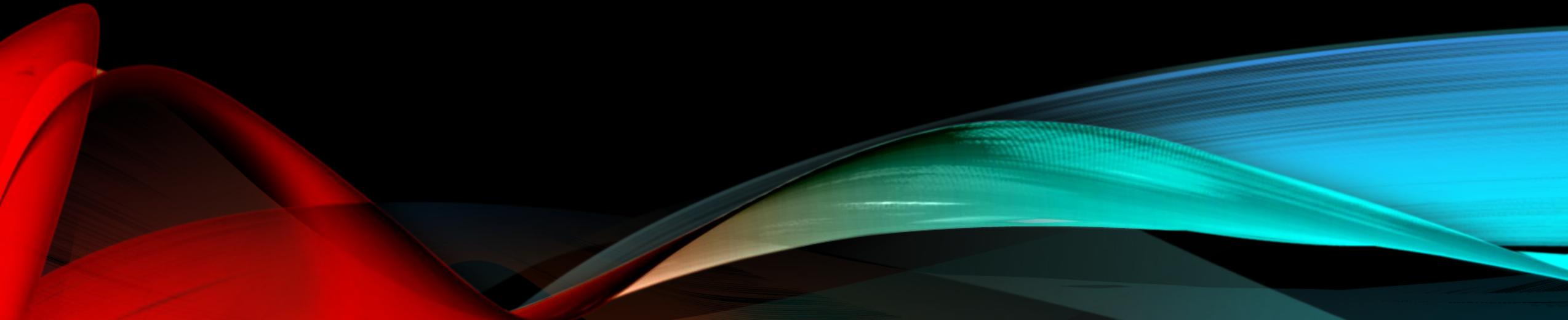
- × уникальность: в произвольный заданный момент времени никакие два различных кортежа  $R$  не имеют одного и того же значения для  $A_i, A_j, \dots, A_k$ ;
- × минимальность: ни один из атрибутов  $A_i, A_j, \dots, A_k$  не может быть исключен из  $K$  без нарушения уникальности.

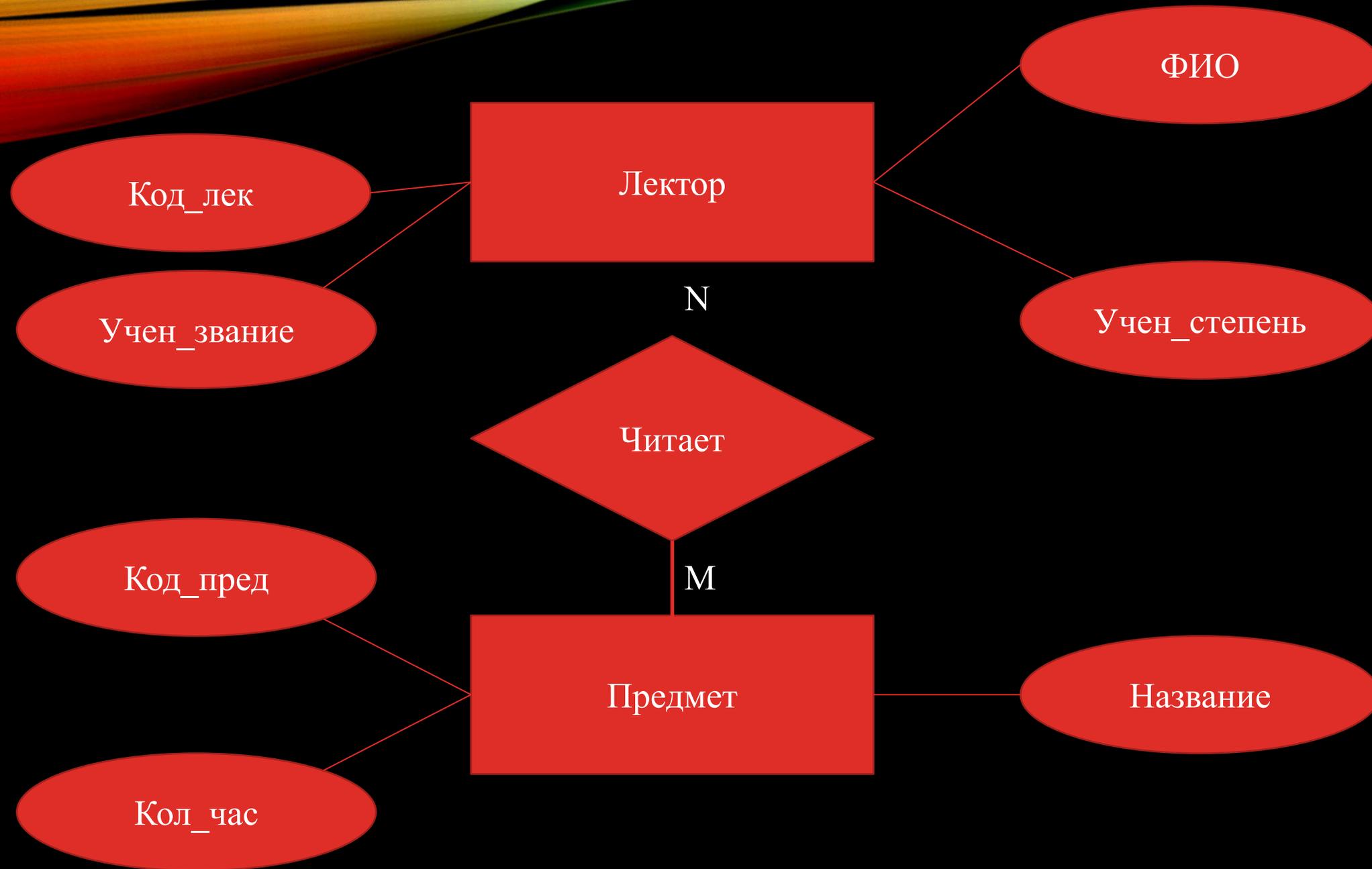
Отношение может иметь несколько потенциальных ключей. Ключ, содержащий два и более атрибута, называется *составным ключом*.

Один из возможных ключей принимается за его *первичный ключ*. Обычно первичным ключом назначается тот возможный ключ, которым проще всего пользоваться при повседневной работе. Остальные возможные ключи, если они есть, называются *альтернативными ключами*. Для индикации связи между отношениями используются внешние ключи.



*Внешний ключ* — это набор атрибутов одного отношения, являющийся потенциальным ключом другого отношения. Причем, атрибуты внешнего ключа не обязательно должны иметь те же имена, что и атрибуты ключа, которым они соответствуют.







Код_лек	ФИО	Учен_степень	Учен_звание
01	Иванов С.И.	к.т.н.	Доцент
02	Петров М.Т.	к.т.н.	Доцент
03	Сидоров Р.А.	к.т.н.	Доцент
04	Чистов К.Л.	д.т.н.	Профессор

ЛЕКТОР

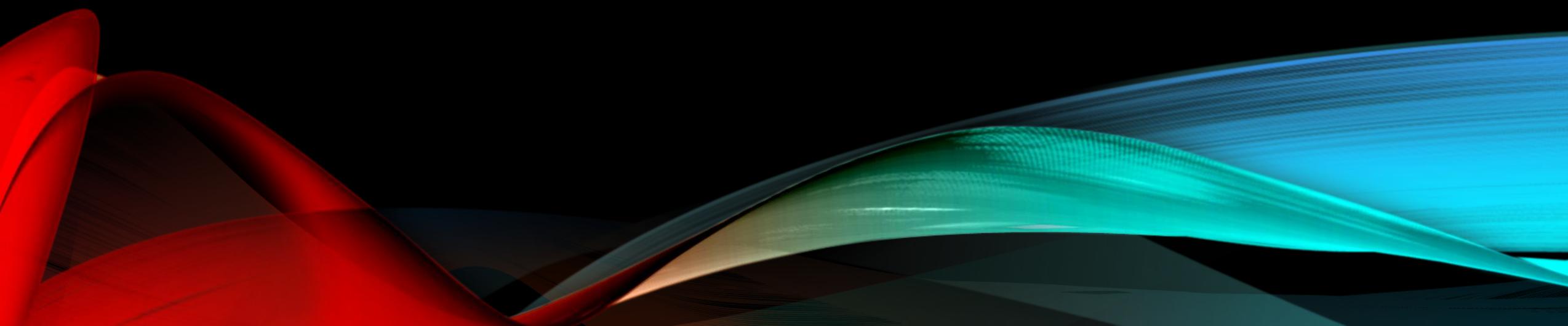
Код_пред	Код_лек
01	02
02	03
03	09
04	07
05	08
06	00

ЧИТАЕТ

Код_пред	Наз_пред	Кол_час
01	ИБ	54
02	БСБД	102
03	ООП	36
04	ЯП	54
05	ЭиС	54
06	ОАиП	36

ПРЕДМЕТ

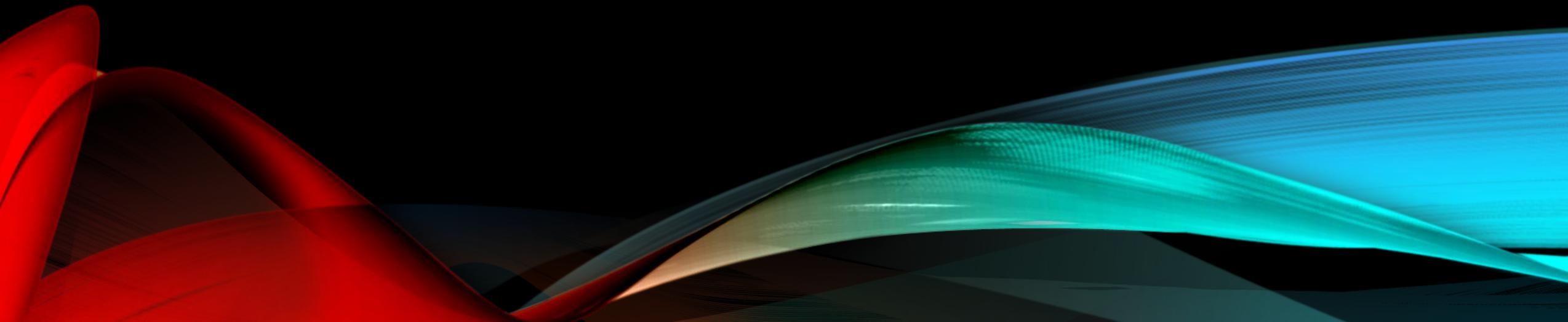
ЛЕКТОР — 4-х арное отношение, с первичным ключом Код\_лек, с кардинальным числом равным четырем; атрибуты определены на следующих доменах: Код\_лек — {целые 1..9}, ФИО — {возможные фамилии и инициалы}, Уч\_степень — {Канд\_техн\_наук, Докт\_техн\_наук, Не\_имеет\_степени}, Уч\_звание — {Доцент, Профессор, Не\_имеет\_звания}.



ПРЕДМЕТ – тернарное отношение, с первичным ключом Код\_пред, с кардинальным числом равным шести; атрибуты определены на следующих доменах: Код\_пред — {символьный}, Назв\_пред — {ИБ, БСБД, ООП, ЯП, ЭиС, ОАиП}, Кол\_во\_час — {целые 54, 102, 36}.



ЧИТАЕТ – бинарное отношение, с составным первичным ключом Код лек, Код пред, с кардинальным числом равным шести, в котором присутствуют первичные ключи только читающих лекторов и первичные ключи только читаемых предметов.



# ОБНОВЛЕНИЕ ОТНОШЕНИЙ

Для обновления отношений необходимо иметь возможность выполнять следующие операции:

- × добавление кортежа;
- × удаление кортежа;
- × изменение кортежа.

*Операция добавления* для отношения  $r$  ( $A_1, A_2, \dots A_n$ ) имеет вид:

$ADD (r; A_1=d_1, A_2=d_2, \dots A_n=d_n).$

Если порядок атрибутов фиксирован, возможна более короткая запись:

$ADD (r; d_1, d_2, \dots d_n).$

Выполнение этой операции может стать невозможным в ряде случаев:

- × добавляемый кортеж не соответствует схеме определенного отношения;
- × некоторые значения кортежей не принадлежат соответствующим доменам;
- × описанный кортеж совпадает по ключу с кортежем, уже находящимся в отношении.

*Операция удаления* предназначена для удаления кортежей. Она может быть записана так:

$\text{DEL (r; A}_1=\mathbf{d}_1, \mathbf{A}_2=\mathbf{d}_2, \dots \mathbf{A}_n=\mathbf{d}_n),$

или для упорядоченных атрибутов:

$\text{DEL (r; } \mathbf{d}_1, \mathbf{d}_2, \dots \mathbf{d}_n).$

Для удаления некоторого кортежа часто достаточно указать значение некоторого ключа:

$\text{DEL (r; ключ).}$

*Операция изменения* предназначена для модификации части кортежа. Для отношения  $r$  ее можно при  $\{c_1, c_2, \dots, c_p\} \in \{A_1, A_2, \dots, A_n\}$  определить так:

СН  $(r; A_1=d_1, A_2=d_2, \dots, A_n=d_n;$

$c_1=e_1, c_2=e_2, \dots, c_p=e_p)$ .

Если  $K=\{B_1, B_2, \dots, B_m\}$  является ключом, то запись данной операции может быть сокращена:

СН  $(r; B_1=d_1, B_2=d_2, B_m=d_m;$

$c_1=e_1, c_2=e_2, \dots, c_p=e_p)$ .

Возможные ошибки в данном случае те же, что и у предыдущих операций:

- ✗ указанный кортеж не существует,
- ✗ изменения имеют неправильный формат,
- ✗ используемые значения не принадлежат соответствующим доменам.