**БД 01.10.25**

**Нормализация БД**

Нормализация основана на анализе функциональных зависимостей между атрибутами отношения

Функциональная зависимость: X → Y

Подмножество атрибутов X – детерминант функциональной зависимости.

Пример:

Сотрудники (код, ФИО, пол, дата рождения).

Код → ФИО

Код → дата рождения

Код → пол

Код → (ФИО, дата рождения)

Каждому значению кода соответсвует одно значение. ФИО и дата рождения взаимно независимые.

**Математические свойства функциональных зависимостей (правила Армстронга):**

1. Рефлективность – если y являеться подмножеством x то x определяет y, при этом такая зависимость – тривиальная. Y c X => x → y

2. Дополнение – A → B => AC → BC

3. Транзитивность – A → B и B → C => A → C.

4. Самоопределение – X → X

5. Декомпозиция и композтция:

A → BC => A → B, A → C

A → B и C → D => AC → BA

6. Теорема всеобщего объединения – если A → B и C → D => A объединение (C – B) → BD

Множество всех функциональных зависимостей – **замыкание**.

**Непреводимым** – множество зависимостей в котором не одна зависимость не может быть выведена из другой.

**Свойства непреводимых зависимостей:**

1. В каждой части зависимости должн быть только один атрибут

2. Из левой части зависимости нельзя удалить ни одного атрибута без потери этой зависимости

**Нежелательные зависимости.** Некоторые зависимости входящие в фз язвляються не желательными, так как приводять к дублированию информации, что создает проблемы обновления данных и предстовляет угрозу целостности бд.

Пример:

Сотрудники (код, ФИО, пол, дата рождения, возрастная категория). Добовление **вк** приведет к новой зависимости.

(Дата рождения, пол) → Возрастная категория

Можно онтнести к нежелательным: много дубликатов, нужно выполнять перещет.

**Декомпозия без потерь (теорема Хеза)**

**Декомпозиция отношения** – это взятие одной или несколько проэкций исходного отношения, так чтобы эти проэкции содержали возможно с повторениями все атрибуты исходного отношения. В декомпозиции не должны быть потеряны данные.

Пусть дано отношение R(A, B, C) – если имееться функциональная зависимость атрибутов, то проэкции R1(A, B) и R2(A, C) образуют декомпозицию без потерь. При этом информация потеряна не будет.

**Нормализация** – обратимый пошаговый процесс декомпозиции отношений, с устранением нежелательных **фз**.

**Формы нормализации:** (каждая форма лучще предыдущей)

1. Отношение находиться в 1 форме если: Значение его всех атрибутов атомарны и остутсвуют повторяющиеся группы атрибутов.

Доходы сотрудников (код, фио, доход за январь…, доход за декабрь...)

2. Первичный ключь может быть составным атрибутом.

**Полная фз** – неключевой атрибут функционально полно зависит от составного ключа, если он он зависит от всего ключа в целом, но не находиться в зависимости от подмножества атрибутов составного ключа.

3. x → y, y → z, z транзитивно зависим от x. Отношение в 3 когда оно в 2 и каджый не ключевой атрибут нетразитивно зависит от первичного ключа

Нормальная форма (BCNF) – отношение в bcnf если оно в 3 форме, и в ней отсутвует зависимости атрибутов первичного ключа от неключевых атрибутов. Ситуация когда отношение в 3 форме но не в bcnf возникает при условии, что отношение имеет, 2 или более возможных ключа, которые состовные и имеют общие атрибуты.

4. Касаеться того где есть повторяющиеся наборы данных. Нарушение 4 формой в таблицах связках, в которых более 2 сущностей.

Пример:

Расписание (день недели, номера пары, аудитория, преподователь, предмет)

(день недели, пара, аудитория)

(день недели, пара, преподователь)

(день недели, пара, предмет)

Отношение в 4 форме если оно в bcnf и в нем отсутвуют многозначные зависимости.

5. Отношение в 5 форме тогда и только тогда, когда любая зависимость по соединению в нем определяеться только его потенциальными ключами.

**Денормализация**

Недостатки:

1. Неэкономное использование диска

2. Прблемы с обновлением данных в виду нарушения принциа каждый факт в одном месте

3. Проблемы контроля целостности.

Недостаток нормализации – замедление работы СУБД при выполнении запросов на выборку данных. В нормализованой бд практичиски каждый запрос требует соединения данных из нескольких таблиц. А соединение таблиц требует значительных ресурсов.

Денермализация – это процесс модификации структуры таблиц нормализованной бд с целью повышения производительности, засчет допущения некоторой управляемой избыточности данных.