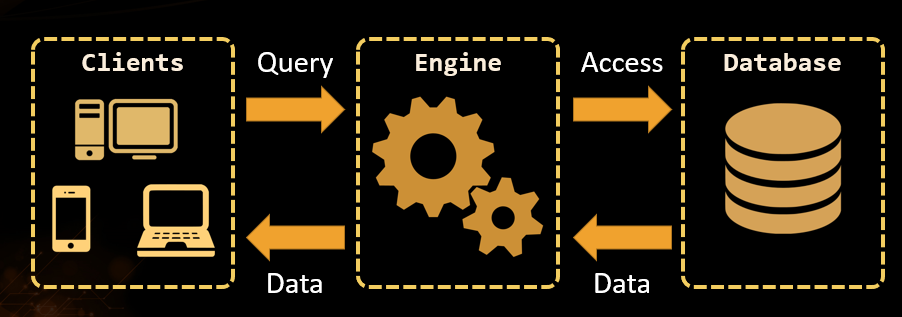
Тема 9: Бази от данни

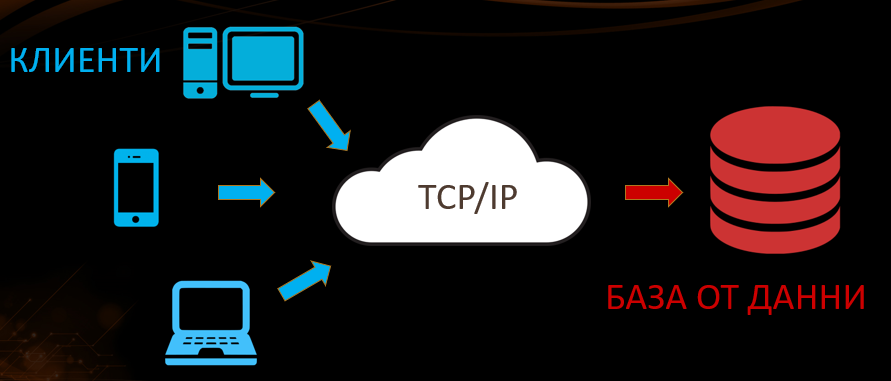
**План-тезис:** Въведение в базите от данни. Типове данни. Основни команди. Моделиране на релационни бази от данни. Заявки за извличане и промяна на данни. Сложни заявки за извличане на данни. Съединения на таблици (SQL JOIN). Агрегиращи функции. Групиране на данни. Скаларни функции, транзакции, съхранени процедури, тригери.

**Въведение в Бази от данни:**

БД е организирана колекция от информация; Налага правила на съдържащите се дании; Система за Управление на Релационна База от Данни (СУРБД) предоставя инструменти за управление; Потребителя няма пряк достъп до съхранените данни.

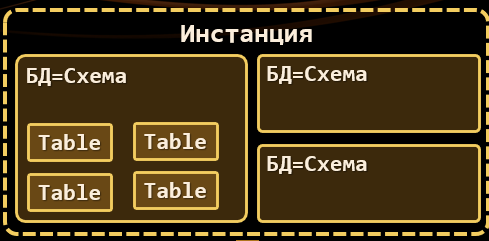
SQL server използва Client-Server модел за комуникация





Архитектура: Логическо и Физическо хранилище.

Логическо: Инстанция(на SQL Server), База от данни(схема), таблица.



Физическо: Файлове с данни и записи, страници с данни.

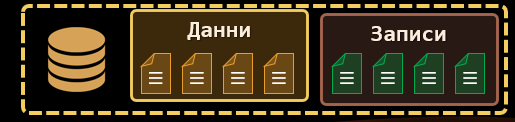
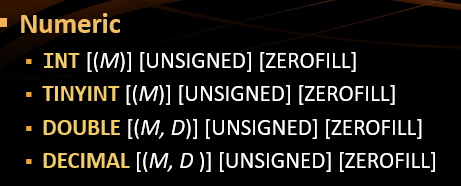
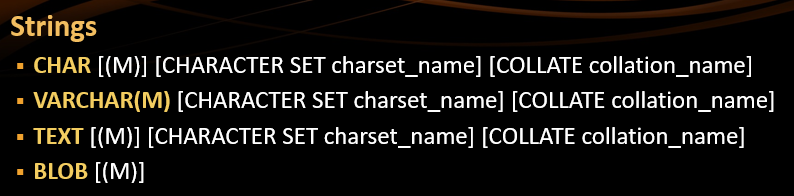


Таблица: Основен градивен елемент на всяка база данни. Всеки ред от една таблица е запис, а колоните определят типа данни.

**Типове данни в БД:**

****

****

****

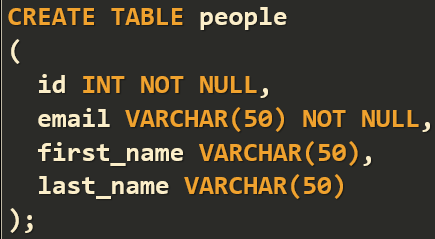
**Основни команди:**

За комуникация се използва SQL(Structured Query Language). Декларативен език. Разделен на четири части: Data definition (описва структурата на данните), Data manipulation (записва и чете данни), Data control (определя кой има достъп до данните), Transaction control (пакетни операции и позволяване на възстановяването).

Свързани таблици: разделяме данните и въвеждаме релации между таблиците, за да се избегне повтаряне на информация. Връзката е установена чрез външен ключ(Foreign Key) в едната таблица, сочещ към първичния ключ(Primary Key) в другата.

DDL(Data Definition Language):

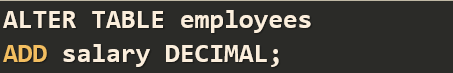




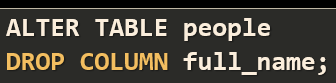
Промяна на таблица: ALTER TABLE



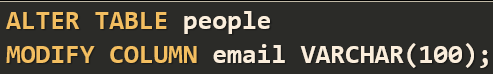
Добавяне на колона:



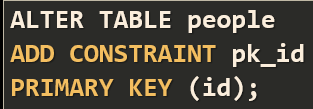
Изтриване на колона:



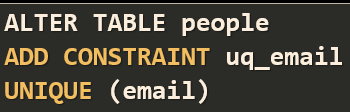
Промяна на типа данни на съществуваща колона:



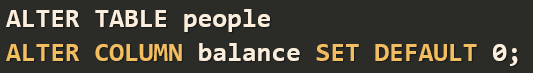
Добавяне на първичен ключ към съществуваща колона:



Добавяне на уникално ограничение:



Стойност по подразбиране:



Изтриване:

Изтриването на структури се нарича dropping. Може ключове, ограничения, таблици, бази. Изтриване на данни е truncating. Не могат да бъдат отменени.

Изтриване на записи:



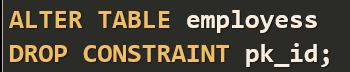
Изтриване на данни и структура:

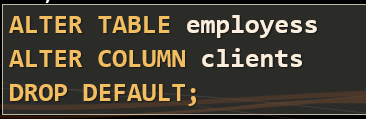


Премахване на база данни:



Премахване на ограничения:





**Моделиране на релационни бази от данни:**

Проектиране на бази данни:

1. Идентифициране на видовете данни
2. Идентифициране на колоните в таблицата
3. Задаване на основен ключ(обикновено допълнителна колона, int, auto\_increment)
4. Идентифициране и моделиране на отношенията
5. Задаване на ограничения
6. Наливане на информация

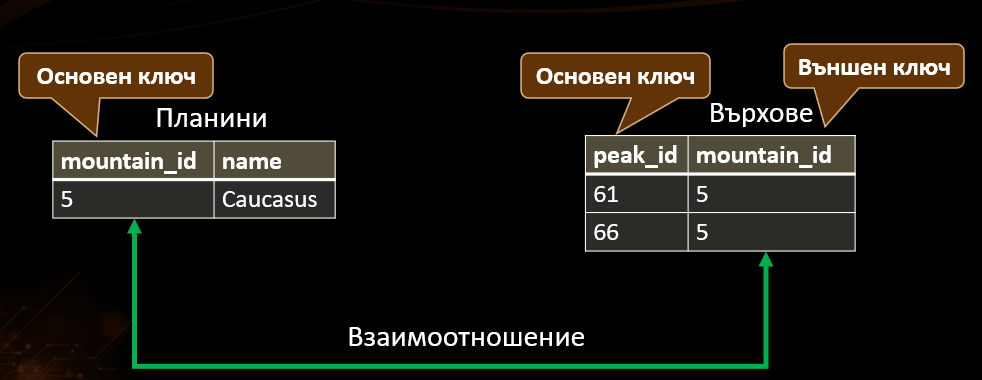
Взаимоотношения и видовете им:

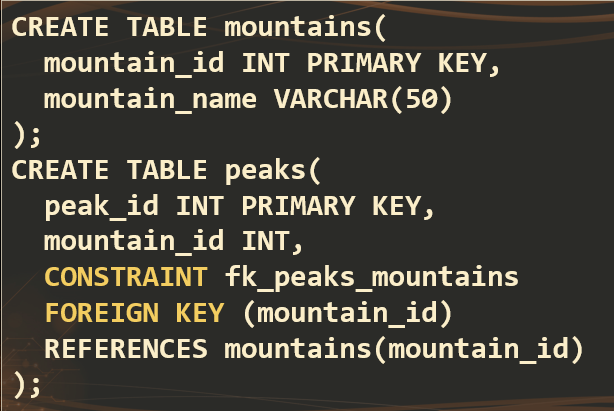
Базирани са на PK-FK връзки.

Видове:

1. Един към много
2. Много към много
3. Един към един

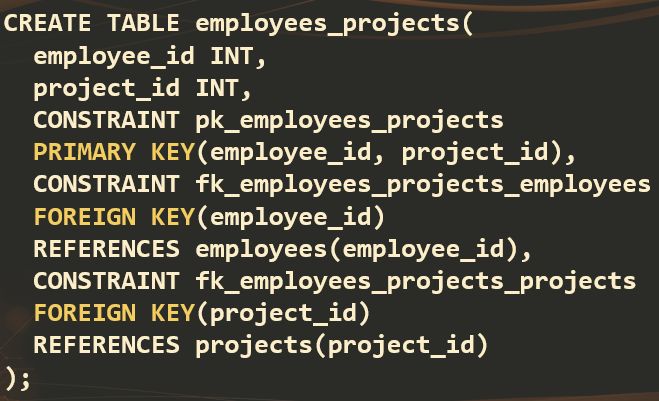
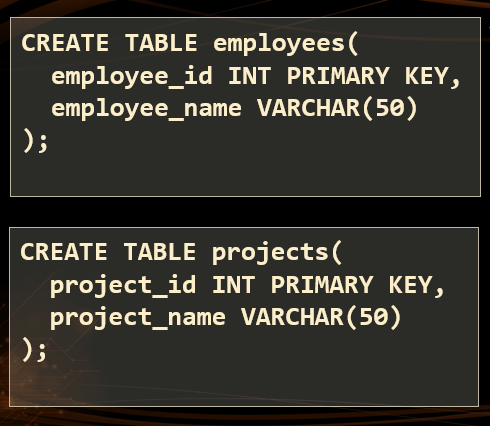
Един към много/Много към един (One to many)



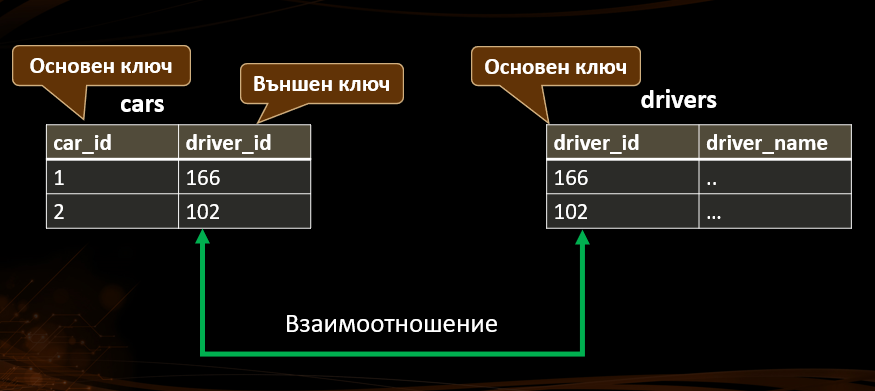


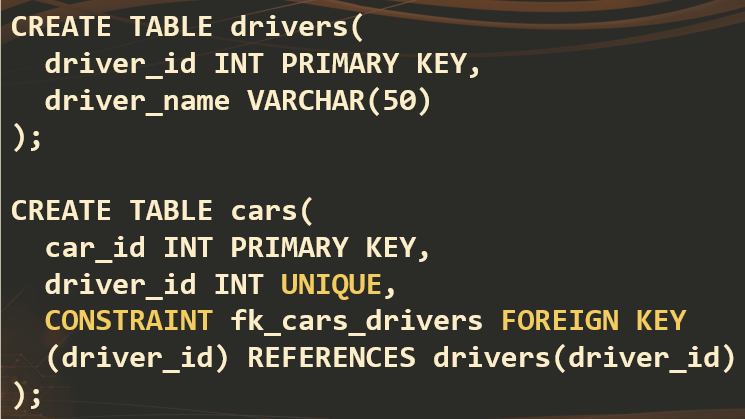
Много към много(Many to many)





Един към един (One to one)





E/R диаграма(Релационна схема)

Схема на всяка от таблиците, показва релациите между таблици и други елементи от базата(ограничения). Описва структурата

Ограничения:

Задават правила за данните в дадена таблица.

Свойства:

Primary Key - първичен ключ

Auto\_Increment - автоматично увеличаване

Unique - уникални стойности

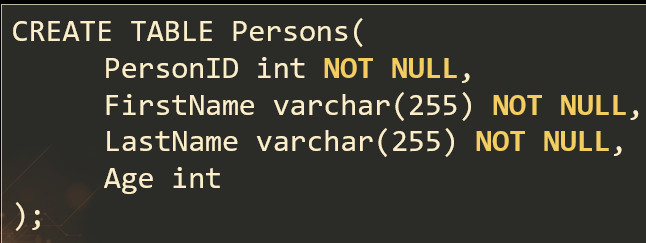
Default - стойност по подразбиране

Foreign key- външен ключ

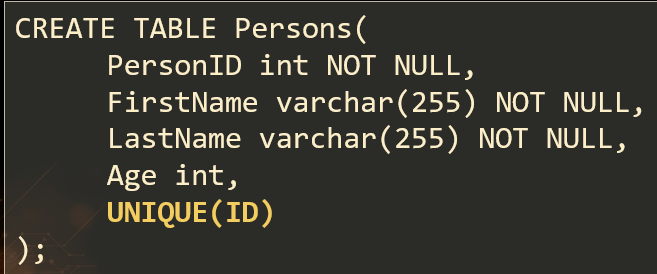
Not null - не позволява клетката да е празна

Check - гарантира, че всички данни в колоната удовлетворяват дадено условие

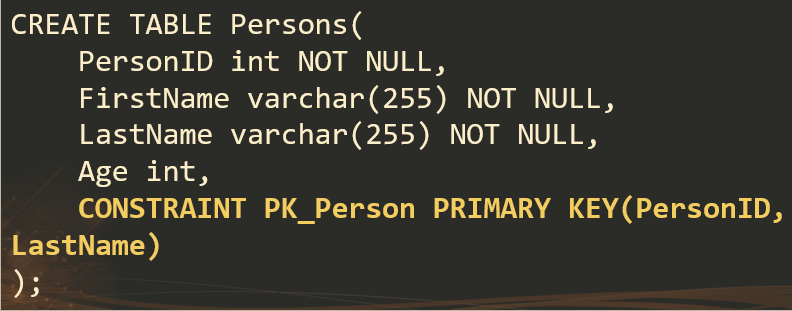
NOT NULL



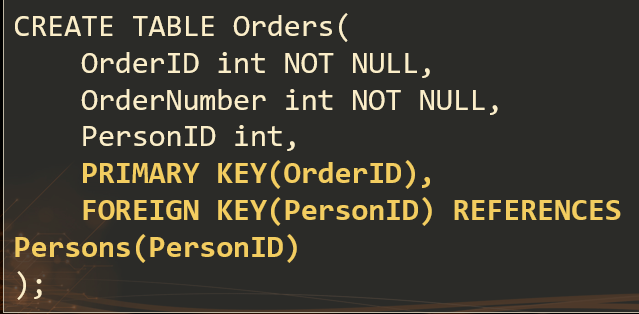
UNIQUE



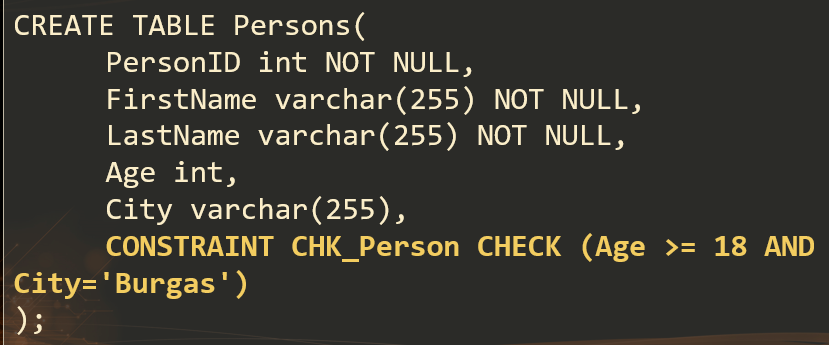
PRIMARY KEY



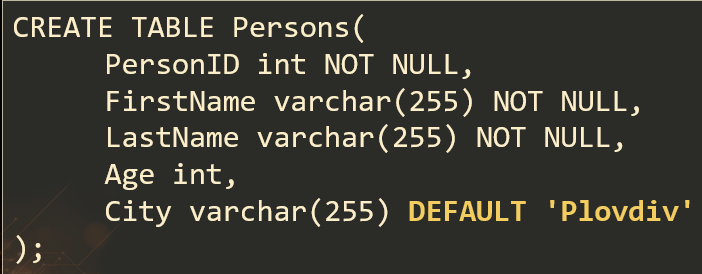
FOREIGN KEY



CHECK

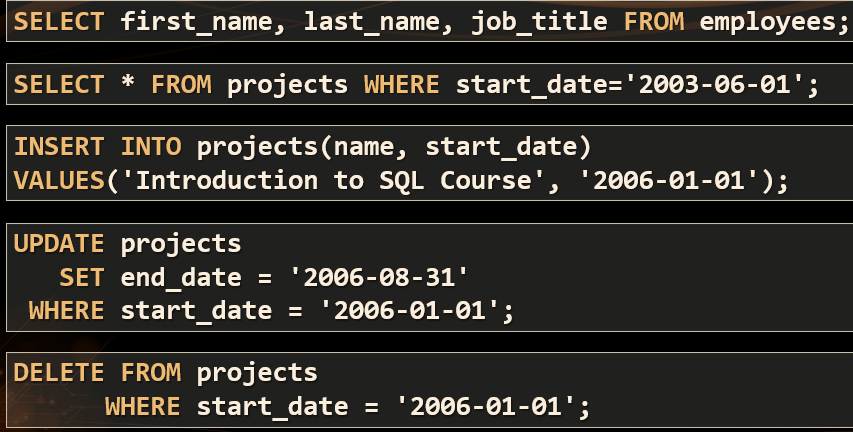


DEFAULT



**Заявки за извличане на данни**

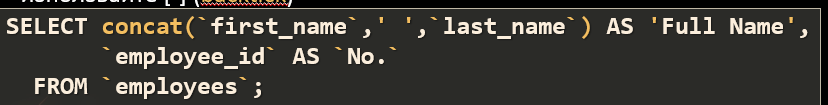
SELECT, INSERT, UPDATE, DELETE



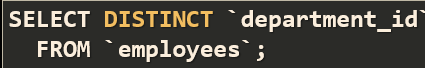
Псевдоним: Изписва id, вместо employee\_id.



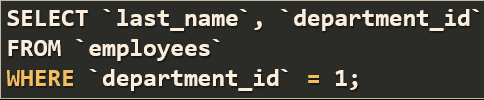
CONCAT: Събира колони или символни низове.



DISTINCT: Премахва дублиращи се записи



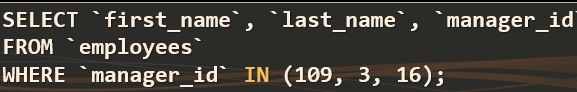
WHERE: Филтриране по указано условие



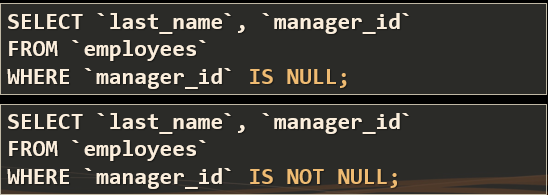
Други условия за сравнения: NOT, OR, AND, BETWEEN, IN/NOT IN





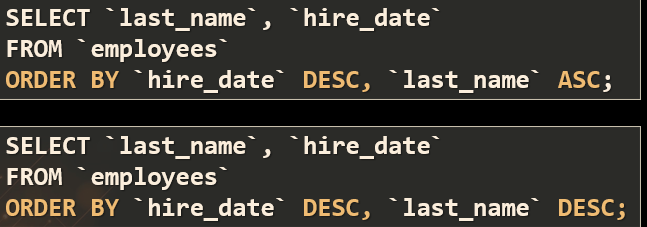


Сравнение с NULL

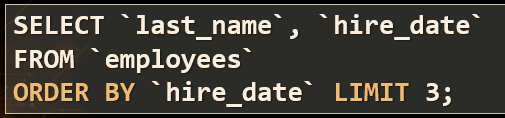


**Сложни заявки за извличане на данни**

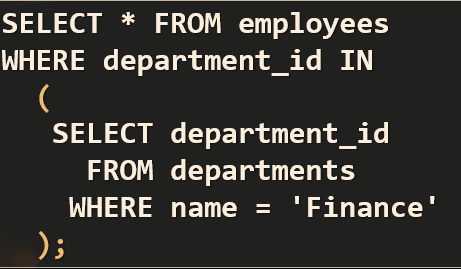
ORDER BY - Сортира(АSC - възходящ, DESC - низходящ)



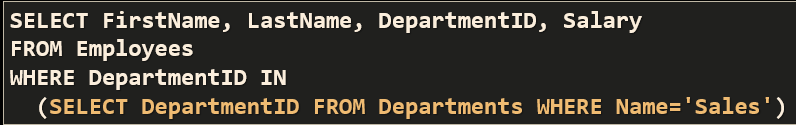
LIMIT- Ограничава броя записи.



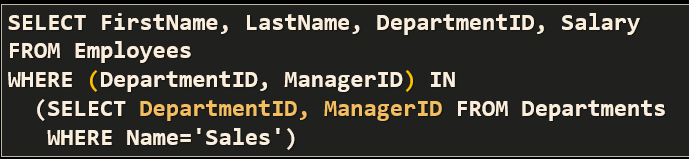
Подзаявки: вложени заявки. Видове: скаларни(връща една стойност), колонни(връща колона), таблични(връща таблица)



IN - Проверка за наличие. Изискват колонна подзаявка



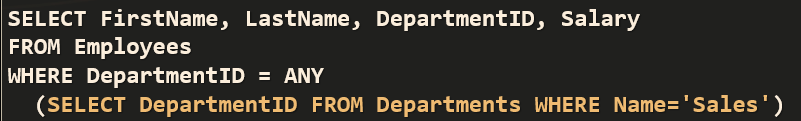
\* Ако подзаявката връща таблица.



ALL- дали условието е в сила за всички

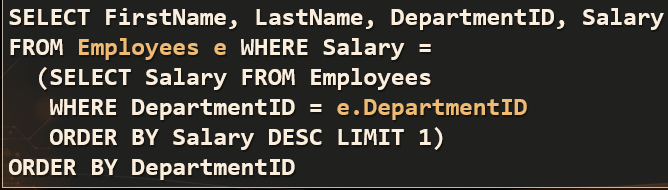
ANY - дали условието е в сила за поне 1

SOME = ANY



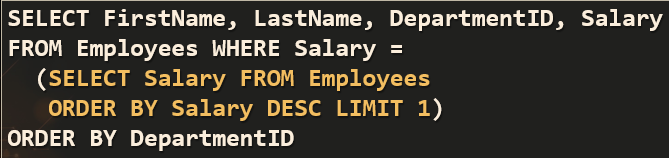
Взаимно свързани заявки

Неща от външния SELECT могат да бъдат споменати във вътрешния SELECT



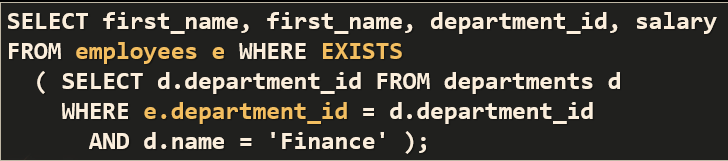
Необвързани заявки

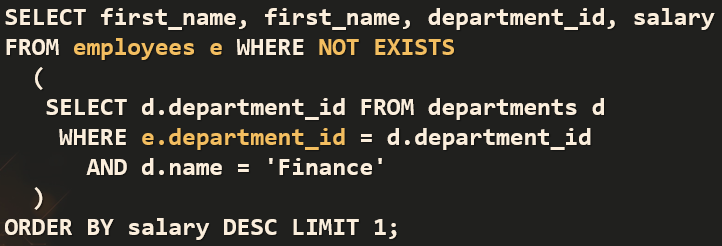
Външния SELECT може да се използва самостоятелно.



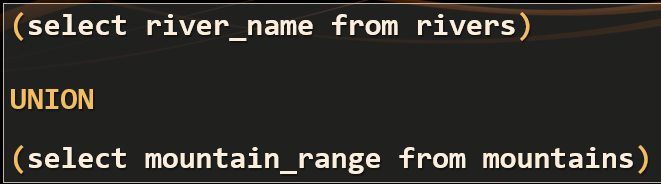
EXISTS/NOT EXISTS

EXISTS е вярно ако подзаявката връща записи, NOT EXIST - ако е празна подзаявка. Ползват се във взаимно свързани заявки.





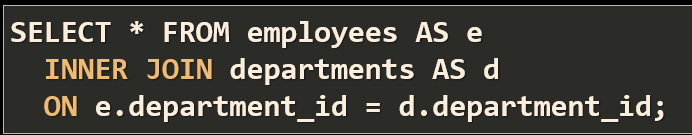
UNION - Обединява заявки от различни таблици. Колоните в двете заявки трябва да са с един и същи брой



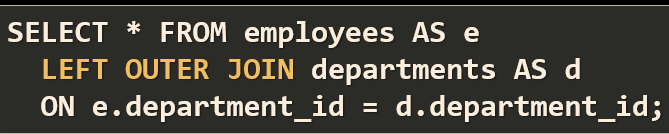
**Съединения на таблици (SQL JOIN)**

Видове JOIN: INNER, LEFT и RIGHT, FULL OUTER, CROSS

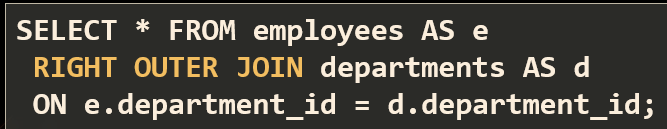
INNER JOIN



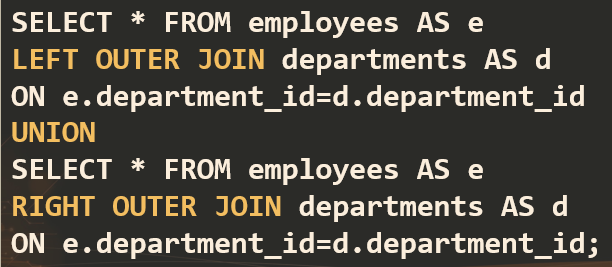
LEFT OUTER JOIN



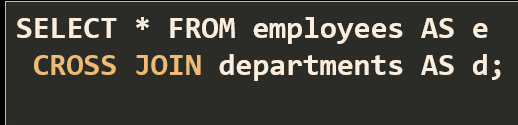
RIGHT OUTER JOIN



FULL OUTER JOIN

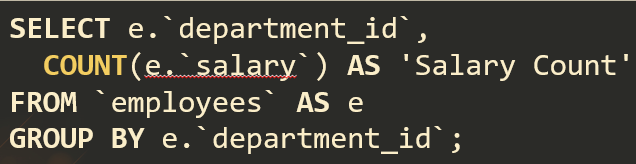


CROSS JOIN

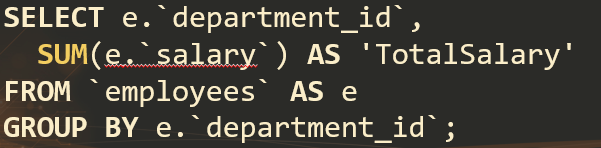


**Агрегиращи функции.**

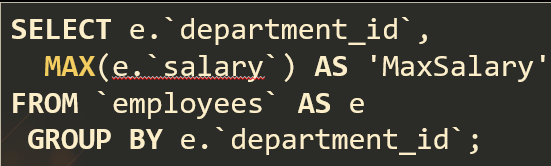
COUNT - брои всичко стойности различни от NULL в една или повече колони, според критерий.



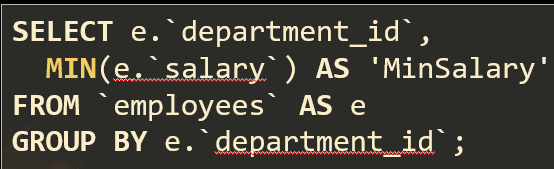
SUM- сумира всички стойности в колоната.



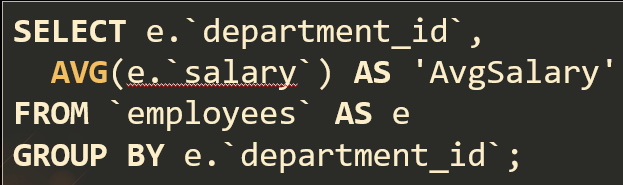
МАX - максимална стойност в колоната



MIN - минималната стойност в колоната

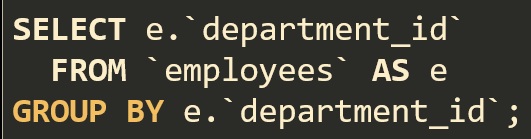


AVG - средна стойност в колоната

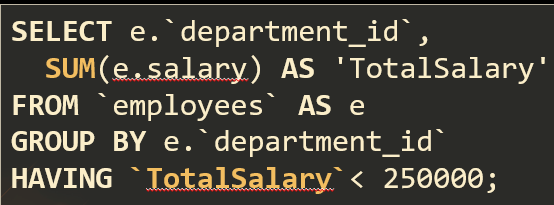


**Групиране на данни:**

Събира информация в отделни групи според общо свойство.

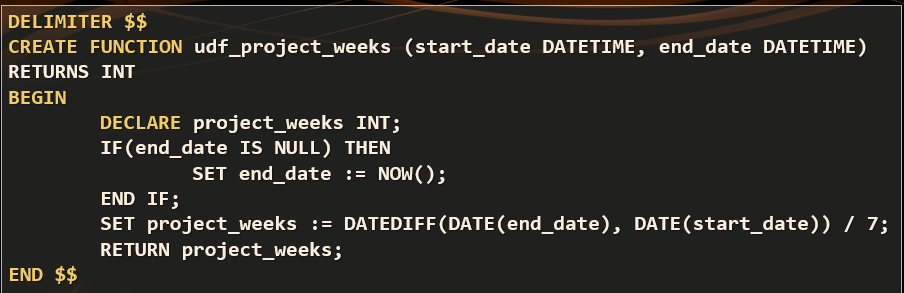


При филтриране се използва HAVING, след агрегирането. HAVING извършва филтриране преди да се случи агрегирането.

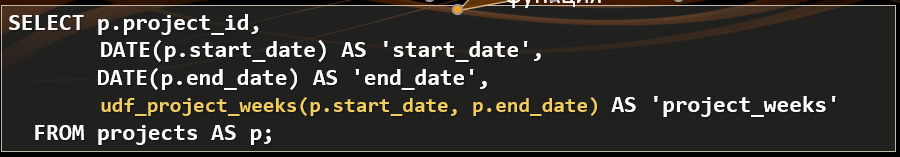


**Скаларни функции, транзакции, съхранени процедури, тригери.**

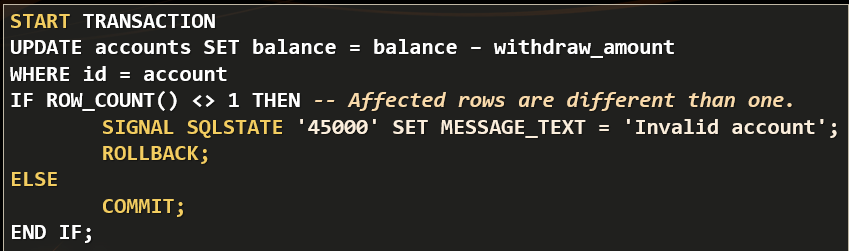
Функции: Приемат параметри, връщат стойност.



Изпълнение

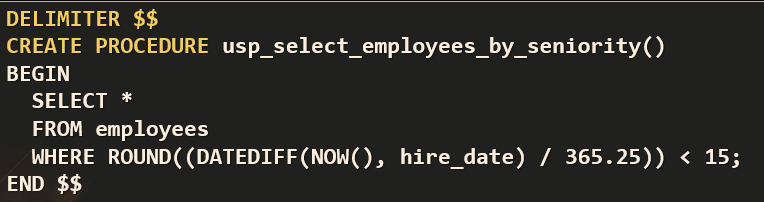


Транзакции: поредица от действия изпълнявани като цялост. Или се изпълняват всички успешно или нито едно от тях се изпълнява.



(Промените се съхраняват след COMMIT, Отменят се по всяко време с ROLLBACK)

Съхранени процедури: Капсулират повтаряща се логика, приемата входни параметри, връщат изходни параметри.



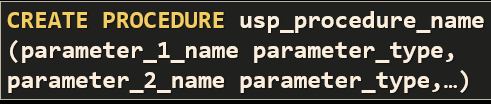
Изпълнение:



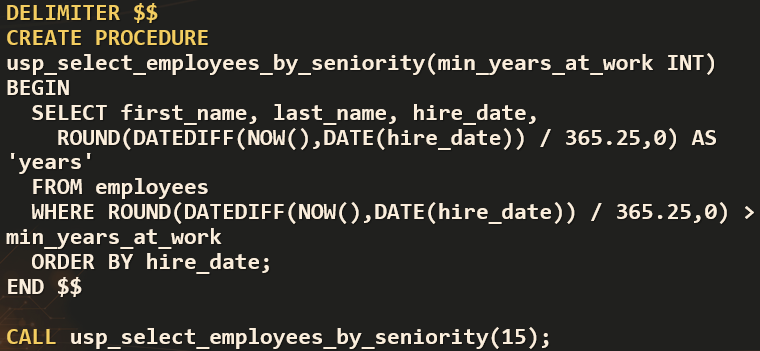
Изтриване:



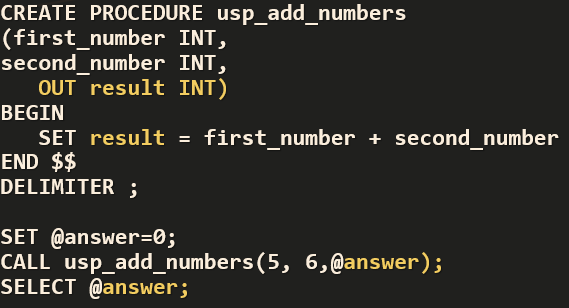
Дефиниране на параметризирани процедури:



Параметризирани съхранени процедури



Връщане на стойности



Спусъци(Triggers): Приличат на съхранени процедури, не ги извикваме изрично, изпълняват се като определена заявка върху съдържанието на таблицата.

