

Въведение в Базите от данни



Съдържание

Управление на
данните. Кога
се нуждаем от
БД?

Машина на
Базата от
данни.



Съхранение или Управление?

1. Съхранението на данни не е основната причина да се използват бази от данни.
2. Обичайното съхранение поражда въпроси, свързани с размера, лекотата на актуализиране, точността, съкращенията/излишеството.

Бази от данни и СУРБД.

БД е организирана колекция от информация.

Налага правила на съдържащите се данни.

Релационното съхранение е първо предложено от Едгар Код през 1970 г.

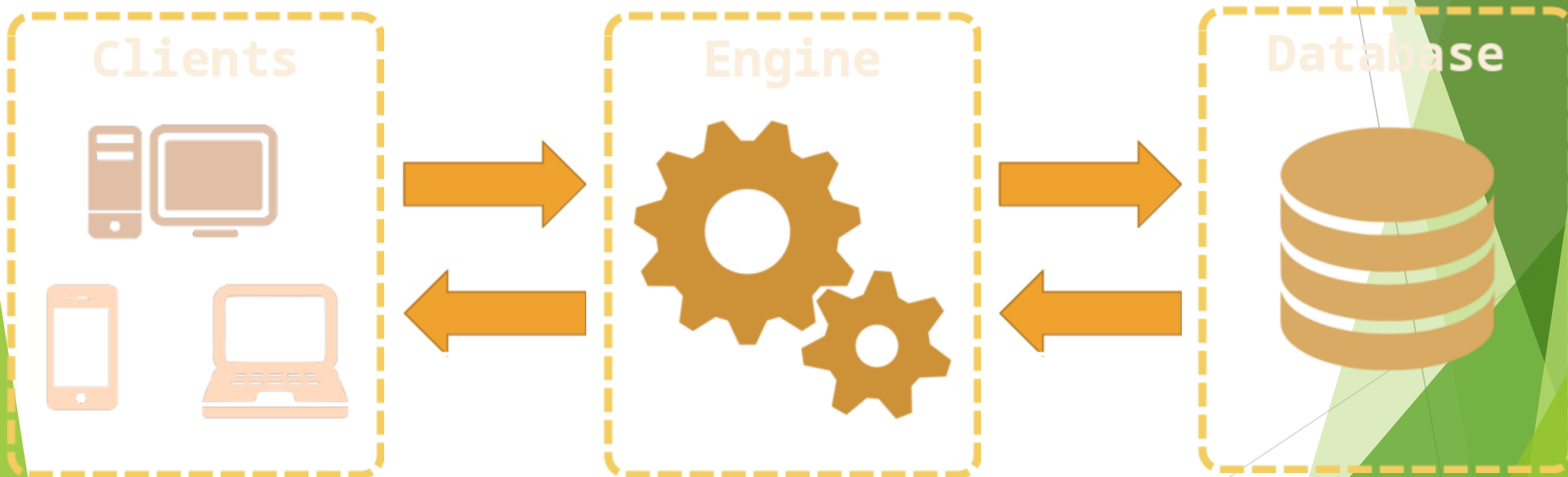
Системата за Управление на Релационна База от Данни предоставя инструменти за управление на база данни.

Тя прави разбор на заявките от страна на потребителя и взема необходимите мерки.

Потребителят няма пряк достъп до съхранените данни.

Поток на Базата от данни

SQL Server използва модел, наречен Клиент-Сървър.



SQL

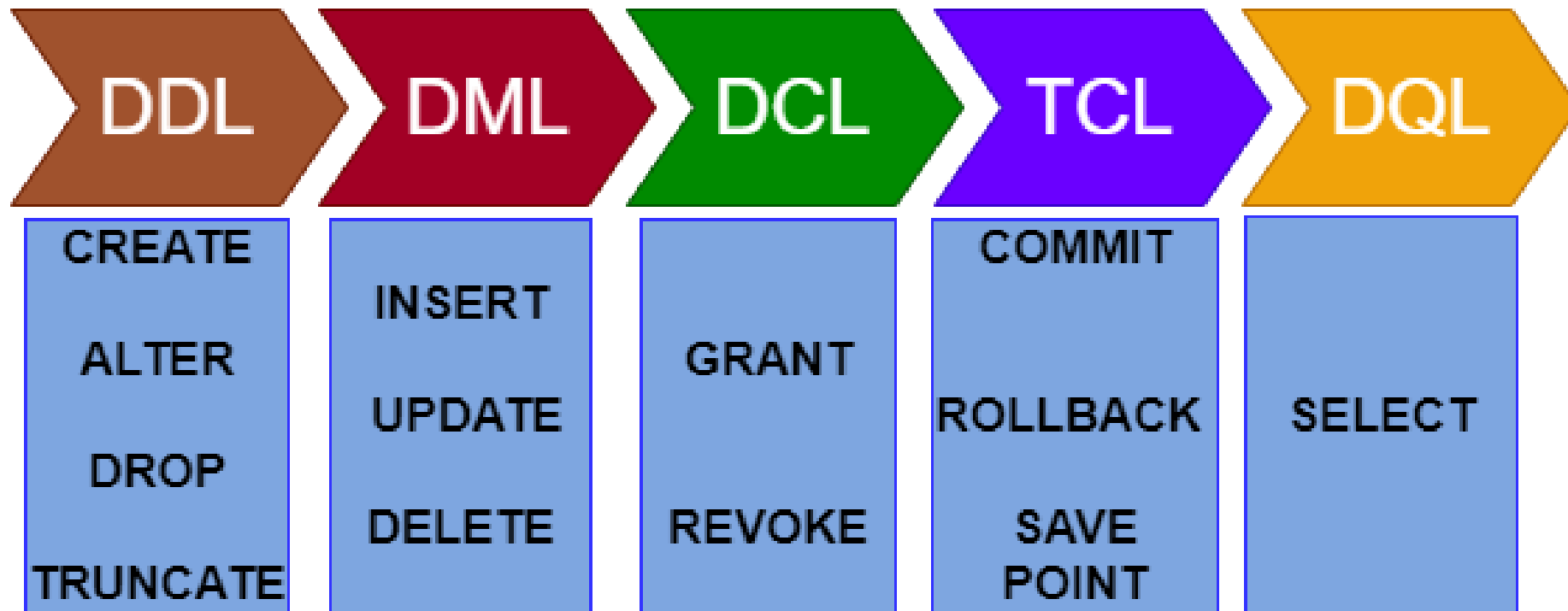


Въведение в SQL. Основни команди

SQL - структуриран език за заявки

- За комуникация с машината ние използваме SQL.
 - Декларативен език
- Логически разделен на четири части:
 - Data Definition - описва структурата на нашите данни
 - Data Manipulation - записва и чете данни
 - Data Control - определя кой има достъп до данните
 - Transaction Control - пакетни операции и позволяване на възстановяването

SQL COMMANDS



Типове данни

- **String Data Types**

- Char(n) - Fixed width character string (max 8000 characters)
- Varchar(n) - Variable width character string (max 8000 characters)
- Nvarchar(n) - Variable width Unicode string (max 4000 characters)
- Text - Variable width character string (max 2GB of text data)

- **Numeric Data Types**

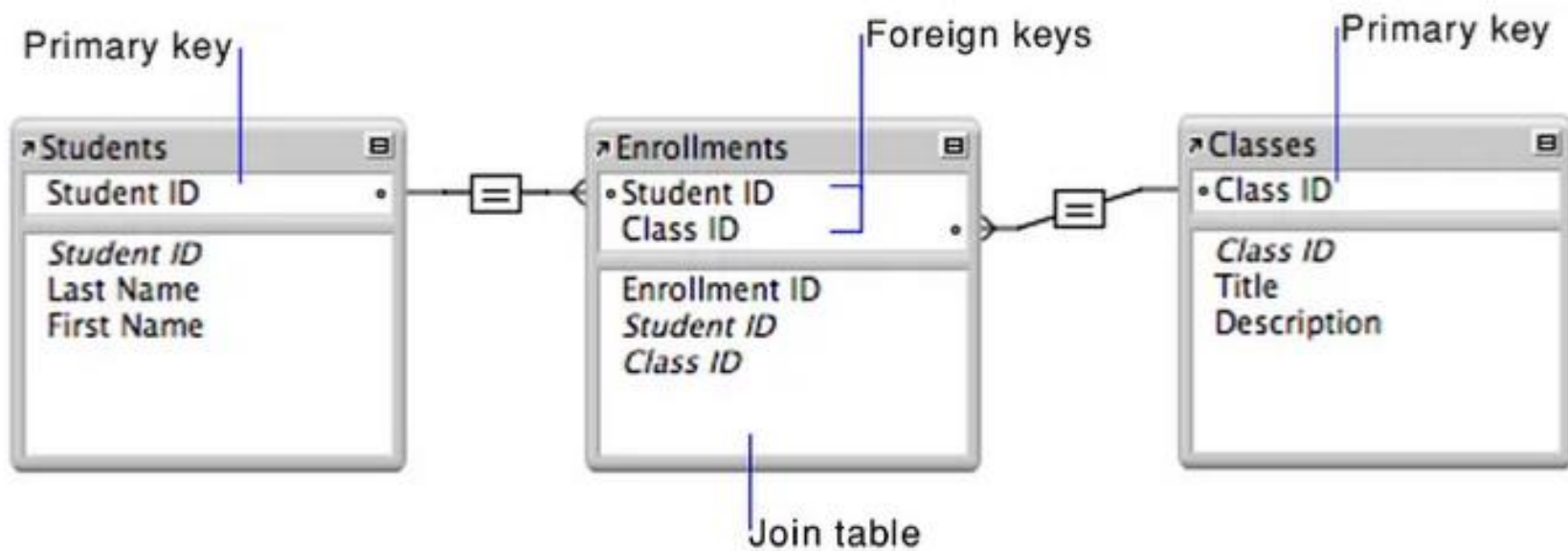
- Tinyint - Allows whole numbers from 0 to 255 (1 byte)
- Smallint - Allows whole numbers between -32,768 and 32,767 (2 bytes)
- Int - Allows whole numbers between -2,147,483,648 and 2,147,483,647 (4 bytes)
- Bigint - Allows whole numbers between -2^{63} and $2^{63}-1$ (8 bytes)
- Decimal(p,s) - p: maximum total number of digits (between 1 and 38); s: maximum number of digits stored to the right of the decimal point (between 0 and p)

Типове данни (2)

- **Data and Time Data Types**
 - Datetime - From January 1, 1753 to December 31, 9999
 - Date - Store a date only. From January 1, 0001 to December 31, 9999
 - Time - Store a time only to an accuracy of 100 nanoseconds

Защо се разделят свързани данни?

order_id	date	customer	product	s/n	price
00315	07/16/2016	David Rivers	Oil Pump	OP147-0623	69.90
00315	07/16/2016	David Rivers	Accessory Belt	AB544-1648	149.99
00316	07/17/2016	Sarah Thorne	Wiper Fluid	WF000-0001	99.90
00317	07/18/2016	Michael Walters	Oil Pump	OP147-0623	69.90



Свързани таблици - разделяме данните и въвеждаме връзки между таблиците, за да се избегне повтаряне на информация. Връзката е установена чрез външен ключ в едната таблица, сочещ към първичния ключ в другата.

T-SQL заявки

- Можем да общуваме с машината на базата от данни, използвайки T-SQL.
- Заявките предоставят по-голям контрол и гъвкавост.
- За да създадем База от Данни с T-SQL:

Име на Базата
от Данни

```
CREATE DATABASE employees
```

Създаване на таблица със T-SQL

```
CREATE TABLE people  
(  
  id INT NOT NULL,  
  email VARCHAR(50) NOT NULL,  
  first_name VARCHAR(50),  
  last_name VARCHAR(50)  
)
```

Име на
таблицата

Персонални
Свойства

Име на колона

Тип данни

Извличане на записи със T-SQL

- За да получите цялата информация от таблица:

Име на таблица

```
SELECT * FROM people
```

- Можете да ограничите колоните и броя на записите:

```
SELECT TOP 5 first_name, last_name  
FROM people
```

Брой записи

СПИСЪК ОТ
КОЛОНИ

Персонализирани свойства на колони

- Първичен ключ `id INT NOT NULL PRIMARY KEY`
- Автоувеличаване `id INT IDENTITY(1,1) PRIMARY KEY`
- Уникални ограничения - неповтарящи се стойности в цялата таблица
`email VARCHAR(50) UNIQUE`
- Стойности по подразбиране
`balance DECIMAL(10,2) DEFAULT 0`

Промяна на таблици с T-SQL

- Добавяне на първичен ключ към съществуваща колона

```
ALTER TABLE people  
ADD CONSTRAINT pk_id  
PRIMARY KEY (id)
```

Име на
ограничение

Име на колона (повече от
една за композитни ключ)

- Добавяне на уникално ограничение

```
ALTER TABLE people  
ADD CONSTRAINT uq_email  
UNIQUE (email)
```

- Задаване на стойност по подразбиране

```
ALTER TABLE people  
ALTER COLUMN balance SET DEFAULT 0
```


Изтриване

- Изтриването на структури се нарича dropping (премахване, падане, сваляне).
 - Можете да премахвате ключове, ограничения, таблици, дори и цялата База от Данни.
 - Изтриване на всички данни в таблица се нарича truncating (отрязване).

И двете действия не могат да бъдат отменени - използвайте ги с повишено внимание!

Премахване и отрязване

- За да изтриете всички записи в таблица:

```
TRUNCATE TABLE people
```

Име на
таблица

- За да премахнете таблица (да изтриете данните и структурата):

```
DROP TABLE people
```

- За премахване на цялата База от Данни:

```
DROP DATABASE employees
```

Име на база
данни

Премахване и отрязване (2)

- За да премахнете ограничаващо правило от колона:

```
ALTER TABLE people  
DROP CONSTRAINT pk_id
```

- За да премахнете стойност по подразбиране:

```
ALTER TABLE people  
ALTER COLUMN age  
DROP DEFAULT
```

Задачи

• Задача 1: Създайте таблица People

- **id** - unique number for every person. (Not null, auto incremented)
- **name** - full name of the person will be **no more than 200 ascii characters**. (Not null)
- **height** - In meters. Real number precise up to **2 digits** after floating point. (Allow nulls)
- **weight** - In kilograms. Real number precise up to **2 digits** after floating point. (Allow nulls)
- **gender** - Possible states are **m** or **f**. (Not null)
- **birthdate** - (Not null)
- **biography** - detailed biography of the person it can contain **max allowed Unicode characters**. (Allow nulls)

Направете ИД първичен ключ. Попълните таблицата с 5 записа. Изпратете вашите CREATE и INSERT заявки до системата за проверка като стартирате заявките и проверите базата от данни (Run queries & check DB).

Задачи

- **Задача 2: БД - Коли под наем** - С помощта на SQL заявки създайте база от данни **car_rental** със следните таблици:

- **categories** (id, category, daily_rate, weekly_rate, monthly_rate, weekend_rate)
- **cars** (id, plate_number, make, model, car_year, category_id, doors, picture, car_condition, available)
- **employees** (id, first_name, last_name, title, notes)
- **customers** (id, driver_licence_number, full_name, address, city, zip_code, notes)
- **rental_orders** (id, employee_id, customer_id, car_id, car_condition, tank_level, kilometrage_start, kilometrage_end, total_kilometrage, start_date, end_date, total_days, rate_applied, tax_rate, order_status, notes)

Задайте най-подходящи типове данни за всяка колона. Задайте първичен ключ за всяка таблица. Попълнете всяка таблица с 3 записа. Изпратете заявката си до системата за проверка като стартирате заявките и проверите базата от данни (Run queries & check DB)..