

M U N I
M E D

Institut
biostatistiky
a analýz

Databázové systémy v biomedicíně

Lekce I - Úvod

About me

Daniel Klimeš

- Vzdělání: Obecná biologie
- PGS: onkologie
- Specializace: Screeningové projekty v onkologii, Národní zdravotní registry
- Databáze MS SQL, PostgreSQL, ORACLE
- Zaměstnání: Datové centrum ÚZIS

□ klimes@iba.muni.cz

E5447 Databázové systémy v biomedicině

Lectures

Každou středu/wednesday od 10:00 – do 11:40

Teoretická přednáška → navazující praktické cvičení

Praktická část/practical tasks : PostgreSQL,
Domácí cvičení/homework: PostgreSQL

Zakončení: zápočet/credit – domácí úkol/homework
zkouška/exam – praktický **test**, časový limit

Rozsah předmětu

- Základy SQL
 - SELECT, UPDATE, DELETE, INSERT
- Funkce a operátory
- Základy agregace dat
 - GROUP BY, HAVING, AVG, SUM, ...
- Práce s více tabulkami
 - JOIN, UNION
- Zanořené dotazy
- Window funkce
 - Rank, lag, lead, OVER
- Vyhledávání v textu
 - Like, regulární výrazy
- Umělá inteligence a SQL
 - ChatGPT

Kdy zpracovávat data v databázi

1. Data jsou primárně v databázi uložena
2. Zpracováváme objemná data v řádu sto tisíc záznamů a více
3. Zpracování dat plánujeme provádět opakovaně
4. S daty bude pracovat více uživatelů
5. Způsob zpracování potřebujete dokladovat
6. Nechcete se zbláznit z excelu

Relační databáze x NoSQL databáze

Význam databáze pro analytika

Databáze
SQL

Statistický SW
Statistika for Windows, R,
SPSS, SAS, MS Excel

- **Předzpracování dat**
- **Čištění dat**
- Popisná analýza
- Filtrování
- Propojení dat
- Agregace dat

**Matematicko - statistický
skriptový SW**
R, Matlab, Maple, programovací jazyk

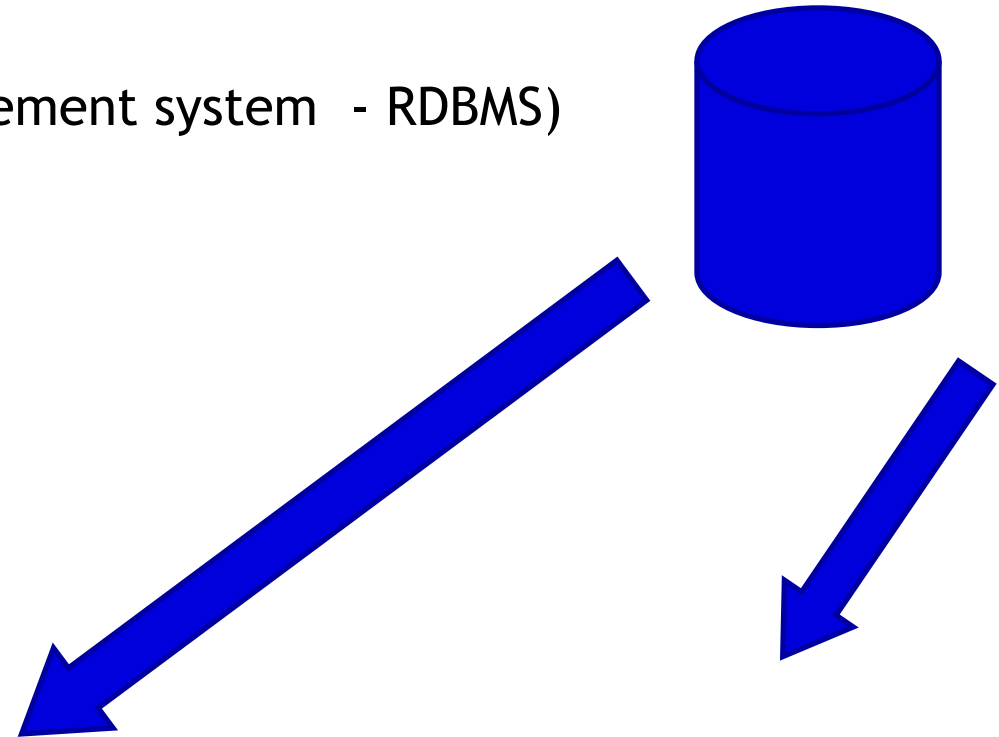
Databázové systémy

Relační databáze (Relational database management system - RDBMS)
Relace/relation - termín z relační algebry

Základ: **tabulka/table**

- sloupec/column = atribut/parametr
- řádek/row = popsáný objekt

Databáze = systém provázaných tabulek



PatientID	FirstName	LastName
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

PatientID	ExaminationDate	Result
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

Databázové systémy - Produkty

Dle dostupnosti

Komerční

ORACLE*
MS SQL*
DB2
MS ACCESS
FOX PRO

Freeware

MySQL
PostgreSQL
Firebird

* Okleštěné verze jsou k dispozici zdarma

Dle počtu uživatelů

Jednouživatelské/single user

MS ACCESS
FOX PRO

Víceuživatelské/multiuser

ORACLE
MS SQL
DB2
MySQL
PostgreSQL
Firebird

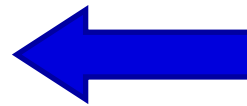
https://en.wikipedia.org/wiki/List_of_relational_database_management_systems

Tabulka/Table

Definovaná struktura, do které se vkládají záznamy

Definují se sloupce/columns

- jméno/name
- datový typ /datatype
 - text
 - číslo/number
 - datum/date
 - BLOB
- doplňující vlastnosti/other properties



PatientID	ExaminationDate	Result
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

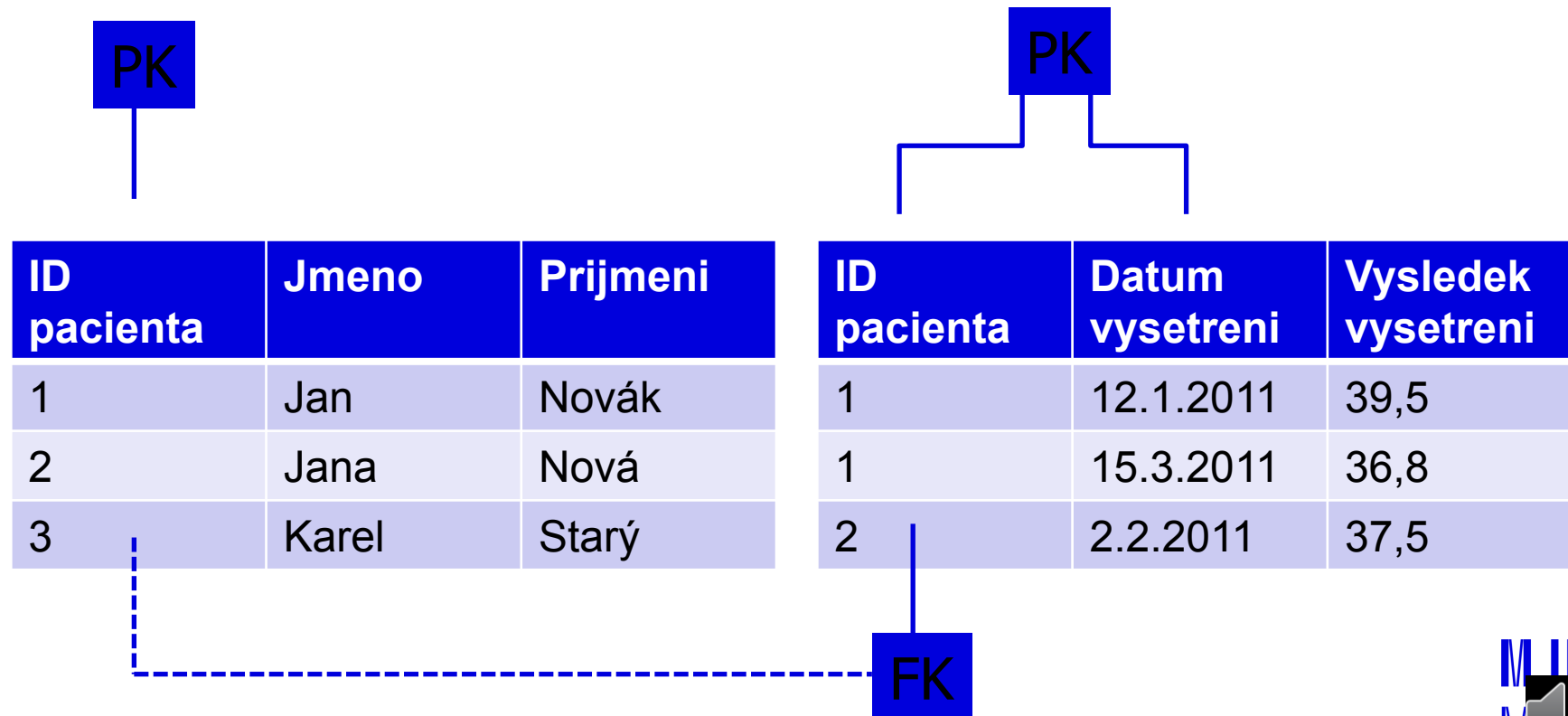
Definice tabulky předchází načtení dat
A table must be created before data import

Klíče/keys

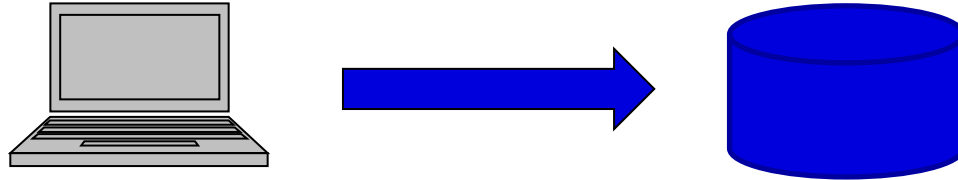
Vybrané sloupce se označují jako klíče (keys)

Primární klíč (primary key - PK)- 1 až n sloupců jednoznačně identifikující řádek (unique values)

Cizí klíč (foreign key - FK) - identifikuje nadřazený řádek v rodičovské tabulce



Přístup do databáze/Access to a database



Klient/client = SW umožňující ověření uživatele a spuštění řídicích příkazů
Ovladač/Driver = komponenta klienta

Řídící příkazy/commands = Structured Query Language - SQL

- DDL = data definition language
 - vytváření, změna, rušení objektů (tabulka, index, pohled, ...)
 - CREATE / ALTER / DROP
- DML = data manipulation language
 - SELECT - získávání dat z databáze
 - INSERT - vkládání dat do databáze
 - DELETE - mazání dat v databázi
 - UPDATE - změna/aktualizace dat
- transakční příkazy / commands for transactions
 - COMMIT - potvrzení transakce
 - ROLLBACK - odvolání transakce

SQL

SQL jazyk (Structured Query Language)

- case insensitive
- klíčová slova /keywords – pro názornost VELKÝM písmem/uppercase
- názvy objektů (tabulek, sloupců) / name of objects
 - pouze alfanumerické znaky / only alphanumeric
 - první znak písmeno / first letter
 - omezená délka (ORACLE 32 znaků) / length limit
- operátory / operators
- funkce / function
- různá rozšíření v jednotlivých DB produktech
- SQL příkazy – ve skriptu ukončeny defaultně středníkem (;)
- komentáře / comments odděleny - - nebo v bloku /* komentar */



PRAKTICKÉ CVIČENÍ



Určete primární a cizí klíč v tabulkách

Lékař

ID pracoviště	ID lékaře	Jméno lékaře	Příjmení lékaře
1	1	Petr	Moudrý
1	2	Josef	Mladý
2	3	Jana	Malá
2	4	Petr	Mladý

Pracoviště

ID pracoviště	Název pracoviště
1	Interna
2	Chirurgie
3	Neurologie
4	Urologie
5	Gynekologie

Cvičení

Instalace PostgreSQL

<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>

Instalace klienta

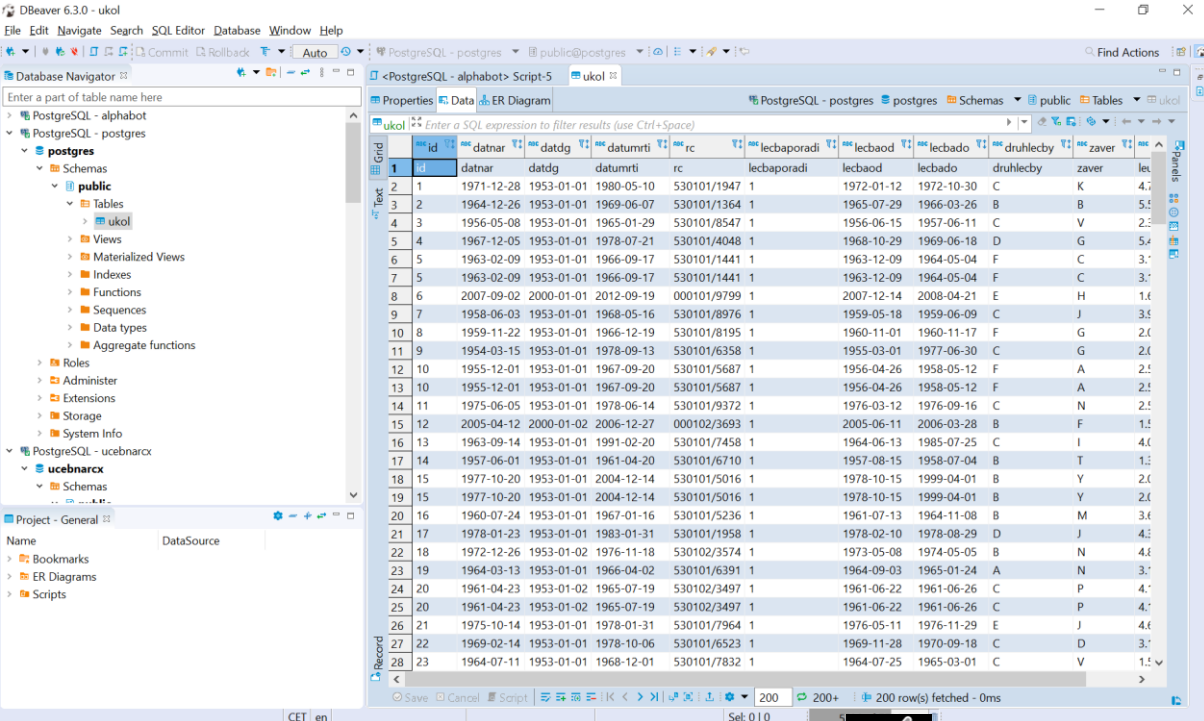
Klient/client

- **pgAdmin – specifický pro postgres**
- Přihlášení k serveru/connection
- Nová databáze/new database - matbi
- Schéma/schema – public

- **Dbeaver**
- Univerzální

<https://dbeaver.io/download/>
Community edition

server je 147.251.145.6
db:ucebnarxc
studentucebna / RCX2019ucebna



DBeaver 6.3.0 - ukol

PostgreSQL - postgres public@postgres

Database Navigator

PostgreSQL - alphabot

PostgreSQL - postgres postgres Schemas public Tables ukol

id	datnar	datdg	datumrti	rc	lecbaporadi	lecbao	lecbado	druhleby	zaver	let
1	1971-12-28	1953-01-01	1980-05-10	530101/1947	1	1972-01-12	1972-10-30	C	K	4.2
2	1964-12-26	1953-01-01	1969-06-07	530101/1364	1	1965-07-29	1966-03-26	B	B	5.5
3	1956-05-08	1953-01-01	1965-01-29	530101/8547	1	1956-06-15	1957-06-11	C	V	2.2
4	1967-12-05	1953-01-01	1978-07-21	530101/4048	1	1968-10-29	1969-06-18	D	G	5.4
5	1963-02-09	1953-01-01	1966-09-17	530101/1441	1	1963-12-09	1964-05-04	F	C	3.1
6	1963-02-09	1953-01-01	1966-09-17	530101/1441	1	1963-12-09	1964-05-04	F	C	3.1
7	2007-09-02	2000-01-01	2012-09-19	000101/9799	1	2007-12-14	2008-04-21	E	H	1.6
8	1958-06-03	1953-01-01	1968-05-16	530101/8976	1	1959-05-18	1959-06-09	C	J	3.3
9	1959-11-22	1953-01-01	1966-12-19	530101/8195	1	1960-11-01	1960-11-17	F	G	2.1
10	1954-03-15	1953-01-01	1978-09-13	530101/6358	1	1955-03-01	1977-06-30	C	G	2.1
11	1955-12-01	1953-01-01	1967-09-20	530101/5687	1	1956-04-26	1958-05-12	F	A	2.1
12	1955-12-01	1953-01-01	1967-09-20	530101/5687	1	1956-04-26	1958-05-12	F	A	2.1
13	1975-06-05	1953-01-01	1978-06-14	530101/9372	1	1976-03-12	1976-09-16	C	N	2.1
14	2005-04-12	2000-01-02	2006-12-27	000102/3693	1	2005-06-11	2006-03-28	B	F	1.1
15	1963-09-14	1953-01-01	1991-02-20	530101/7458	1	1964-06-13	1985-07-25	C	I	4.4
16	1957-06-01	1953-01-01	1961-04-20	530101/6710	1	1957-08-15	1958-07-04	B	T	1.1
17	1977-10-20	1953-01-01	2004-12-14	530101/5016	1	1978-10-15	1999-04-01	B	Y	2.1
18	1977-10-20	1953-01-01	2004-12-14	530101/5016	1	1978-10-15	1999-04-01	B	Y	2.1
19	1960-07-24	1953-01-01	1967-01-16	530101/5236	1	1961-07-13	1964-11-08	B	M	3.6
20	1978-01-23	1953-01-01	1983-01-31	530101/1958	1	1978-02-10	1978-08-29	D	J	4.1
21	1972-12-26	1953-01-02	1976-11-18	530102/3574	1	1973-05-08	1974-05-05	B	N	4.8
22	1964-03-13	1953-01-01	1966-04-02	530101/6391	1	1964-09-03	1965-01-24	A	N	3.1
23	1961-04-23	1953-01-02	1965-07-19	530102/3497	1	1961-06-22	1961-06-26	C	P	4.1
24	1961-04-23	1953-01-02	1965-07-19	530102/3497	1	1961-06-22	1961-06-26	C	P	4.1
25	1975-10-14	1953-01-01	1978-01-31	530101/7964	1	1976-05-11	1976-11-29	E	J	4.4
26	1969-02-14	1953-01-01	1978-10-06	530101/6523	1	1969-11-28	1970-09-18	C	D	3.1
27	1964-07-11	1953-01-01	1968-12-01	530101/7832	1	1964-07-25	1965-03-01	C	V	1.1
28	1964-07-11	1953-01-01	1968-12-01	530101/7832	1	1964-07-25	1965-03-01	C	V	1.1



SQL - SELECT

SELECT * FROM tabulka; -- všechny řádky i sloupce tabulky / all rows , all columns

/* vybrané sloupce, všechny řádky /selected columns */

SELECT sloupec1, sloupec2, sloupec1 + sloupec2 AS soucet FROM tabulka;

/* všechny sloupce, vybrané řádky / selected rows, all columns */

SELECT * FROM tabulka WHERE sloupec1 = 1;

SELECT * FROM tabulka WHERE sloupec2 = 'Jan' ; -- text do apostrofů

SELECT * FROM tabulka
WHERE sloupec1 = 1 AND sloupec2 > 10 AND sloupec3 < sloupec 4

SQL - SELECT

--Setřídění výstupu (ORDER BY)

```
SELECT * FROM tabulka ORDER BY sloupec1; -- vzestupné třídění / ascending
```

```
SELECT * FROM tabulka ORDER BY sloupec2 DESC; -- sestupné třídění / descending
```

```
SELECT * FROM tabulka ORDER BY sloupec1, sloupec2 DESC --kombinace
```

SQL - SELECT

--Sumární výstupy = Agregační funkce

SELECT COUNT(*) FROM tabulka -- počet řádků v tabulce / number of rows

/* suma, aritmetický průměr, minimum, maximum */

SELECT SUM(sloupec1), AVG(sloupec2), MIN(sloupec3), MAX(sloupec4)
FROM tabulka

SELECT COUNT(*), sloupec1 FROM tabulka - nelze / error

Cvičení 2 / Task 2

Spuštění prvního skriptu (skript1.sql) / Run script

Tabulka/Table **STUDENT**

- Zobrazte celý obsah tabulky / select all data from table
- Zobrazte jen jméno a příjmení / select only firstname and lastname columns
- Setříd'te výstup podle studia, jména / order by study
- Kolik má tabulka řádků? / how many rows are in the table?
- Vyberte pouze svůj záznam (své UČO) / select only your UCO row
- Minimum, maximum a průměrná hodnota sloupce *UCO*?

Min, max , average of UCO

- Minimum, maximum z jména / min, max of firstname
- Kolik máme v seznamu mužů? / How many men are in the table?