

TXT CSV JSON XML

DATABÁZOVÉ SYSTÉMY

SQL

Martin Šechný

2022

verzia 1.4

hlavné zdroje <https://sqlzoo.net>
<https://www.w3schools.com>

názov Databázové systémy
autor © 2020 – 2022, Mgr. Martin Šechný, martin.sechny @ shenk.sk
URL <https://www.shenk.sk/skola/informatika/Databazove-systemy.pdf>
licencia CC-BY-SA 4.0



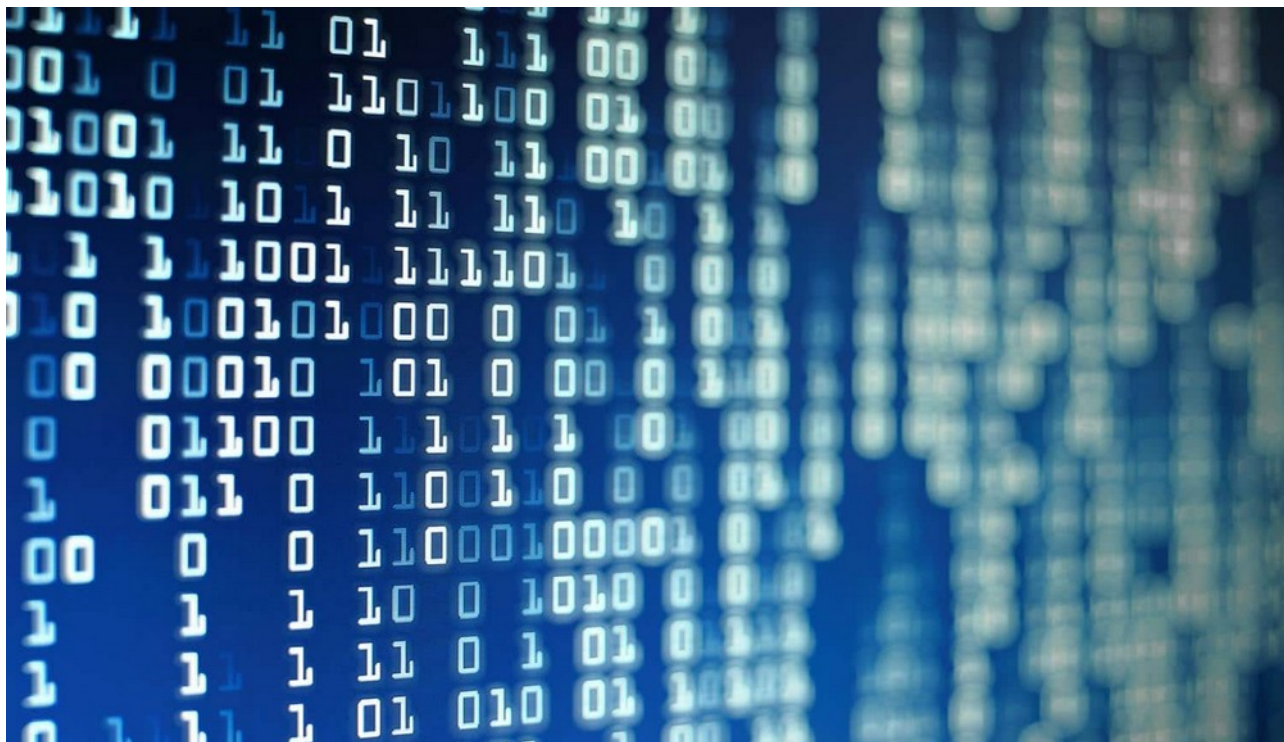
<http://sk.creativecommons.org>
<http://creativecommons.org/licenses/by-sa/4.0/>

Obsah

Úvod	4
1 Základné pojmy	5
1.1 Jednotky v informatike	6
1.2 Tabuľka a tabuľkový editor/procesor	7
2 Relačná databáza a jazyk SQL	9
2.1 SELECT	9
2.2 WHERE	9
2.3 ORDER BY	9
2.4 GROUP BY, HAVING	9
2.5 SELECT IN SELECT	9
2.6 JOIN	10
2.7 UNION	11
2.8 CREATE TABLE	11
2.9 INSERT	11
2.10 ALTER TABLE	11
2.11 DROP TABLE	11
2.12 UPDATE	12
2.13 DELETE	12
2.14 CREATE DATABASE, CREATE USER, GRANT	13
2.15 DROP DATABASE, DROP USER	14
2.16 TRANSACTION	14
2.17 TRIGGER	14
2.18 CONSTRAINT	15
2.19 Projektová úloha	15
3 Dáta v aplikáciách	16
3.1 Ochrana osobných údajov (GDPR)	16
3.2 PHP a dáta (SQL)	16
3.3 PHP a dáta (TXT)	20
3.4 Dátové formáty TXT a CSV	21
3.5 Dátový formát JSON	23
3.6 Dátový formát XML	25
3.7 Prehľad portálov otvorených dát	28
3.8 Analýza dát v tabuľkovom procesore	31
3.9 Vizualizácia číselných dát	33
3.10 Veľké dáta	35
3.11 Java a dáta (TXT, CSV, JSON, XML)	36
3.12 Java a dáta (SQL, JDBC)	38
4 Objektový návrh aplikácie	39
4.1 UML	40
4.2 Návrh aplikácie (UML, SQL)	44
4.3 Implementácia aplikácie (SQL)	44
4.4 Implementácia aplikácie (PHP alebo Java)	44
5 Dátová bezpečnosť	45
5.1 Zálohovanie dát	45
5.2 Zálohovanie databázy	45
5.3 Bezpečnosť a spoľahlivosť dát databázového servera	45
5.4 Oprávnenia k dátam	45
5.5 Šifrovanie dát a podpisovanie dát	45
Záver	49
Zdroje obrázkov	50
Register	51

Úvod

Počítač spracúva dáta. Tie sú najčastejšie digitálne. Analógové dáta sa obvykle najprv prevedú na digitálne dáta a potom sa ďalej spracujú počítačom.



Obrázok 1: Digitálne dáta

Dáta môžu pochádzať z rôznych zdrojov a môžu byť použité na rôzne účely. Dáta by mali byť užitočné. Hospodárstvo postavené na dátach vytvára hodnotu pre digitálnu spoločnosť.

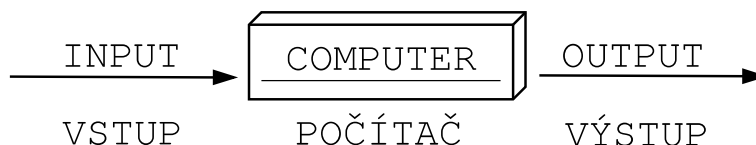


Obrázok 2: Zdroje dát

1 Základné pojmy

V literatúre alebo na internete¹ je možné nájsť rôzne definície základných pojmov z informatiky a špeciálne z databázových systémov:

- ✓ **IT** (*information technology*) – informačná technológia, priemyselný odbor zaoberajúci sa výrobou počítačov a počítačovým spracovaním dát.
- ✓ **IKT/ICT** (*information and communications technology*) – informačná a komunikačná technológia (synonymum k IT, zdôrazňuje integráciu počítačovej techniky a telekomunikácií).
- ✓ **Počítač** (*computer*) – (elektronický, analógový, digitálny) stroj s vlastnou pamäťou, spracúvajúci vstupy podľa programu, odovzdávajúci výstupy.



Obrázok 3: Počítač

- ✓ **Údaje, dáta** (*data*) – postupnosť znakov (symbolov) určená na spracovanie počítačom.
- ✓ **Informácia** (*information*) – údaj, ktorý má hodnotu pre používateľa, znižuje mieru neistoty vo výskyte náhodných udalostí. Informácia ako veličina je číselné vyjadrenie množstva takých dát. Jednotka informácie je 1 bit (b), 1 bajt (B) = 8 b.
- ✓ **Znalosť** (*knowledge*) – zložená informácia z viacerých elementárnych informácií, informácia s kontextom, súhrn aplikovaných poznatkov.
- ✓ **DB – databáza** (*database*) – množina štruktúrovaných údajov uložená v počítačovom systéme, ktorý umožňuje automatizované spracovanie týchto údajov.
- ✓ **DBS – databázový systém** (*database system*) – komplexný softvér na automatizovanú správu štruktúrovaných údajov. Databázový systém zahŕňa všetky spravované databázy a databázový riadiaci systém. Databázový systém obvyčajne používa jazyk SQL.
- ✓ **DBMS – databázový riadiaci systém** (*database management system*) – počítačový program na automatizovanú správu štruktúrovaných údajov. Najznámejšie systémy: Oracle DB, MySQL, MariaDB, PostgreSQL, IBM DB2, MS SQL Server.
- ✓ **SQL** (*structured query language*) – jazyk pre prácu s relačnými databázami.
- ✓ **Relačná databáza** (*relational database*) má dáta štruktúrované do prepojených tabuliek:
 - entita (tabuľka)
 - doména, atribút (stĺpec tabuľky) je premenná, má dátový typ (int, char, varchar...)
 - záznam, pole (riadok tabuľky)
 - relácia (vzťah medzi tabuľkami vytvorený pomocou primárneho a cudzieho kľúča)
- ✓ **E-R diagram** (entitno-relačný diagram v jazyku UML, tabuľkovo-objektová štruktúra dát).

Pri vkladaní dát do databázy možno využiť obmedzenie (*constraint*), spúšťač (*trigger*), transakciu, procedúru (podprogram v programovacom jazyku, napr. Oracle PL/SQL).

- ✓ **Algoritmus** (*algorithm*) – jednoznačný postup zložený z krokov, postup riešenia úlohy. Algoritmus môže obsahovať sekvenciu (jednoduchú postupnosť) krokov, podmienku, vetvenie, cyklus, skok, volanie iného algoritmu a návrat. Algoritmus môže byť konečný alebo nekonečný.
- ✓ **Program** (*program*) – algoritmus zapísaný v programovacom jazyku, zdrojový kód.

¹ Slovenská terminologická databáza, kategória IT, JÚLŠ SAV, <https://terminologickyportal.sk/wiki/Kategória:It>

A teraz na začiatok niekoľko jednoduchých otázok:

- Čo je počítač?
- Aké dáta obsahuje databáza?
- Čo je databázový systém?
- Čím sa graficky znázorní štruktúra dát v relačnej databáze?
- Koľko bitov je 10 bajtov?
- Koľko bajtov je 64 bitov?
- Čo je program?
- Na čo sa používa SQL?

1.1 Jednotky v informatike

Základná jednotka v informatike je **bit** (*b*, *binary digit*). Pomocou bitu vieme zakódovať jeden z dvoch možných stavov. Ďalšia jednotka **bajt** (*B*, *byte*) sa obyčajne používa na kódovanie jedného znaku z 256-znakovej abecedy, kde použijeme 8 bitov, lebo $256=2^8$. Preto 1 bajt má 8 bitov:

$$1 \text{ B} = 8 \text{ b}$$

V praxi sa môžeme stretnúť aj s nesprávnym označením b – bajt.

Násobky jednotiek k, M, G, T a ďalšie sú obyčajne násobkami jednotiek v desiatkovej sústave. Ale v informatike nie je vhodný násobok 1000, lebo informatika používa dvojkovú sústavu. Najbližší vhodný násobok je $1024=2^{10}$. Kilo v informatike sa má písať vždy s veľkým K = 1024. Aby sa predišlo nejasnosti, zaviedlo sa nové označenie² (aspoň písomne, lebo teda na výslovnosť je ťažké):

$$1 \text{ Ki (kibi, kilo binary)} = 2^{10} = 1024$$

$$1 \text{ Mi (mebi, mega binary)} = 2^{20} = 1024 \text{ Ki}$$

$$1 \text{ Gi (gibi, giga binary)} = 2^{30} = 1024 \text{ Mi}$$

$$1 \text{ Ti (tebi, tera binary)} = 2^{40} = 1024 \text{ Gi}$$

V praxi sa používajú oba spôsoby písania násobkov. Treba si dávať pozor na nesprávne použitie, kde napr. K by malo znamenať 1024, ale hodnota je počítaná s násobkom 1000. Pri jednotkách informaticko-fyzikálnych násobok má byť vždy 1000:

$$1 \text{ Kbps} = 1 \text{ kbps} = 1000 \text{ bps} = 1000 \text{ b/s}$$

Pre zvládnutie tohto učebného textu je vhodné poznať dvojkovú číselnú sústavu:

- Preveďte desiatkové číslo 1000 do dvojkovej sústavy.
- Preveďte 8-bitové dvojkové číslo 10 101 010 do desiatkovej sústavy.
- Koľko bajtov je 155 kilobitov?
- Koľko bitov je 36 MiB?
- Ako dlho potrvá kopírovanie 1 TiB disku cez pripojenie s rýchlosťou 20 MB/s?
- Aké najväčšie desiatkové číslo sa dá zapísať do 64-bitového dvojkového tvaru?

² Norma IEC 60027-2, http://sk.wikipedia.org/wiki/IEC_60027-2, http://sk.wikipedia.org/wiki/Binárny_prefix, http://en.wikipedia.org/wiki/Binary_prefix

1.2 Tabuľka a tabuľkový editor/procesor

Relačná databáza má dáta uložené v tabuľkách. Tabuľka je usporiadanie dát do riadkov a stĺpcov. Dáta sú spravidla textové a číselné. Dáta sa usporiadávajú do tabuľky pomocou textového editora alebo tabuľkového editora. Prvý riadok tabuľky (hlavička tabuľky) obsahuje označenie stĺpcov (domén, atribútov, premenných).

Tabuľka 1: Tabuľka s komplexnými dátami

meno a priezvisko	adresa
Ernest Hašter	Nová ulica 10, 000 00 Mesto
Júlia Krehká	Stará ulica 8, 000 00 Mesto

- ✓ Textový editor a tabuľkový editor formátujú text bez štýlu. Rámčeky nie sú súčasťou tabuľky. Rámčeky môžu byť dodatočne zobrazené programom pre ľahšie čítanie dát.
- ✗ Textový procesor a tabuľkový procesor formátuje text so štýlom. Rámčeky sú súčasťou štýlu tabuľky. Tabuľkový procesor umožňuje vkladať do tabuľky okrem textu a čísel aj vzorce a funkcie. Tabuľka so štýlmi, vzorcami, funkciami nie je vhodná pre relačnú databázu.

Dobrá databáza má dáta rozdelené na najmenšie zmysluplné časti – atomické dáta. Je lepšie rozdeliť meno a priezvisko, aby sa osoba dala vyhľadať len podľa mena alebo podľa priezviska.

Tabuľka 2: Tabuľka s atomickými dátami

meno	priezvisko	ulica	číslo	psč	obec
Ernest	Hašter	Nová ulica	10	00 000	Mesto
Júlia	Krehká	Stará ulica	8	00 000	Mesto

Programátor by mal uprednostniť angličtinu pri návrhu databáz, aby zvýšil použiteľnosť svojho produktu na iný účel a v inom prostredí. Angličtina je preferovaný jazyk v IT praxi.

Tabuľka 3: Tabuľka s atomickými dátami a kľúčom

id	firstname	surname	street	number	zip	city
1	Ernest	Hašter	Nová ulica	10	00 000	Mesto
2	Júlia	Krehká	Stará ulica	8	00 000	Mesto

Je užitočné každému riadku priradiť **id** (identifikátor), neprázdnu jedinečnú hodnotu, ktorá zaberá čo najmenej bajtov, aby sa dalo úsporne a jednoznačne ukazovať na jeden konkrétny riadok (záznam). Taký atribút je kľúčový a obyčajne má číselné hodnoty počítané od 1.

Tabuľka 4: Tabuľka s dátovými typmi

id int	firstname char(20)	surname char(20)
1	Ernest	Hašter
2	Júlia	Krehká

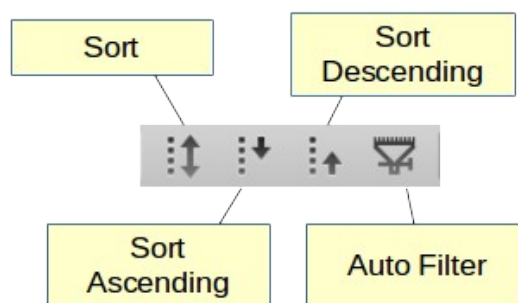
Každý stĺpec (atribút) môže mať hodnoty len jedného dátového typu.

Celé číslo **int** (*integer*) je najčastejším číselným dátovým typom. Text **char(n)** alebo **varchar(n)** alebo **string** je znakový dátový typ (*character*) s pevným alebo variabilným počtom znakov, s maximom **n**.

Ďalšie dátové typy: **decimal** alebo **float** alebo **double** je reálne (desatinné) číslo, **date** je dátum, **time** je čas, **timestamp** je dátum a čas (časová pečiatka), **file** alebo **binary** alebo **blob** je súbor (binárne dáta).

Dobrá tabuľkový editor alebo tabuľkový procesor, rovnako ako databázový systém, poskytuje základné databázové operácie s tabuľkou:

- Zobrazit'/skryť vybrané stĺpce.
- Vyhľadať (filtrovať) riadky podľa hodnoty daného stĺpca.
- Usporiadať riadky abecedne/číselne podľa stĺpca, a to vzostupne alebo zostupne.
- Spočítať riadky v tabuľke.
- Vykonať číselné alebo znakové funkcie na hodnotách v stĺpci.



Obrázok 4: LibreOffice Calc: usporiadať a filtrovať

Použité usporiadanie a použitý filter:

	A	B	C	D	E
1	Empid	Name	Title	Department	Salary
2	100	Thomas	All		\$5,000.00
3	200	Jason	Top 10	ology	\$5,500.00
4	300	Raj	Standard Filter...	ology	\$7,000.00
5	400	Nisha	- empty -	ing	\$9,500.00
6	500	Randy	- not empty -	ology	\$6,000.00
7			DBA		
8			Developer		
9			Manager		
10			Sysadmin		
11					

Obrázok 5: LibreOffice Calc: použité usporiadanie a použitý filter

Otázky a úlohy:

- V tabuľkovom editore/procesore (napr. LibreOffice Calc) vytvorte tabuľku pre zákazníkov obchodu (e-shop).
- V tabuľkovom editore/procesore vytvorte tabuľku pre tovar obchodu (e-shop).
- Navrhните, ako zaznamenávať predaj v obchode (e-shop), aký každý zákazník mohol nakúpiť niekoľko položiek (rôzny tovar, rôzne množstvo).
- V tabuľkovom editore/procesore vyhľadajte (filtrujte) tovar s cenou 1 €.
- V tabuľkovom editore/procesore usporiadajte zákazníkov abecedne podľa mena.
- V tabuľkovom editore/procesore usporiadajte tovar podľa ceny od najdrahšieho.
- Koľko zákazníkov je zapísaných v tabuľke?
- Koľko tovarov má cenu nad 10 €?
- Zobrazte zákazníka s číslom 2 a počet ním nakúpených položiek.
- Aká je suma predaja v € za celý obchod?
- Má obchod (e-shop) evidovať aj počet kusov každého tovaru na sklade?

2 Relačná databáza a jazyk SQL

SQL (*structured query language*) je jazyk pre prácu s relačnými databázami. SQL je jednoduchý jazyk. Má malú sadu kľúčových slov (príkazov). Je to dátový typ počítačového jazyka – popisuje dáta. SQL má premenné, podmienky, vetvenia, ale tie sa dajú využiť len obmedzene. SQL nie je algoritmický programovací jazyk, nemá univerzálne procedúry alebo funkcie, cykly, skoky. SQL môže byť rozšírený o procedurálny jazyk, napr. Oracle PL/SQL, PHP, Python, Java.

Na webe je niekoľko kvalitných návodov – SQL tutoriálov:

<https://sqlzoo.net>

<https://www.w3schools.com/sql/default.asp>

<https://www.tutorialspoint.com/mysql/index.htm>

<https://www.mysqltutorial.org>

<https://dev.mysql.com/doc/refman/8.0/en/tutorial.html>

2.1 SELECT

<https://www.w3schools.com/sql/default.asp>

https://www.w3schools.com/sql/sql_intro.asp

https://www.w3schools.com/sql/sql_syntax.asp

https://www.w3schools.com/sql/sql_select.asp

2.2 WHERE

https://sqlzoo.net/wiki/SELECT_basics

https://sqlzoo.net/wiki/SELECT_names

https://sqlzoo.net/wiki/SELECT_from_WORLD_Tutorial

https://sqlzoo.net/wiki/SELECT_from_Nobel_Tutorial

https://www.w3schools.com/sql/sql_where.asp

2.3 ORDER BY

https://sqlzoo.net/wiki/SELECT_from_Nobel_Tutorial

https://www.w3schools.com/sql/sql_orderby.asp

https://sqlzoo.net/wiki/Window_LAG

2.4 GROUP BY, HAVING

https://sqlzoo.net/wiki/SUM_and_COUNT

https://www.w3schools.com/sql/sql_groupby.asp

https://www.w3schools.com/sql/sql_having.asp

2.5 SELECT IN SELECT

https://sqlzoo.net/wiki/SELECT_within_SELECT_Tutorial

Úloha:

- Použite <https://sqlzoo.net>, vypíšte kontinent a počet krajín na každom kontinente, ktorých meno krajiny sa začína na druhé písmeno z vášho mena. Meno nech je uvedené v príkaze celé. Kontinenty nech sú usporiadané podľa celkového počtu obyvateľov kontinentu, od najväčšieho.

2.6 JOIN

https://sqlzoo.net/wiki/The_JOIN_operation

https://sqlzoo.net/wiki/More_JOIN_operations

https://sqlzoo.net/wiki/Using_Null

https://sqlzoo.net/wiki/Self_join

https://sqlzoo.net/wiki/Subquery_and_JOIN

https://sqlzoo.net/wiki/Subquery_and_JOIN2

https://www.w3schools.com/sql/sql_join.asp

https://www.w3schools.com/sql/sql_join_inner.asp

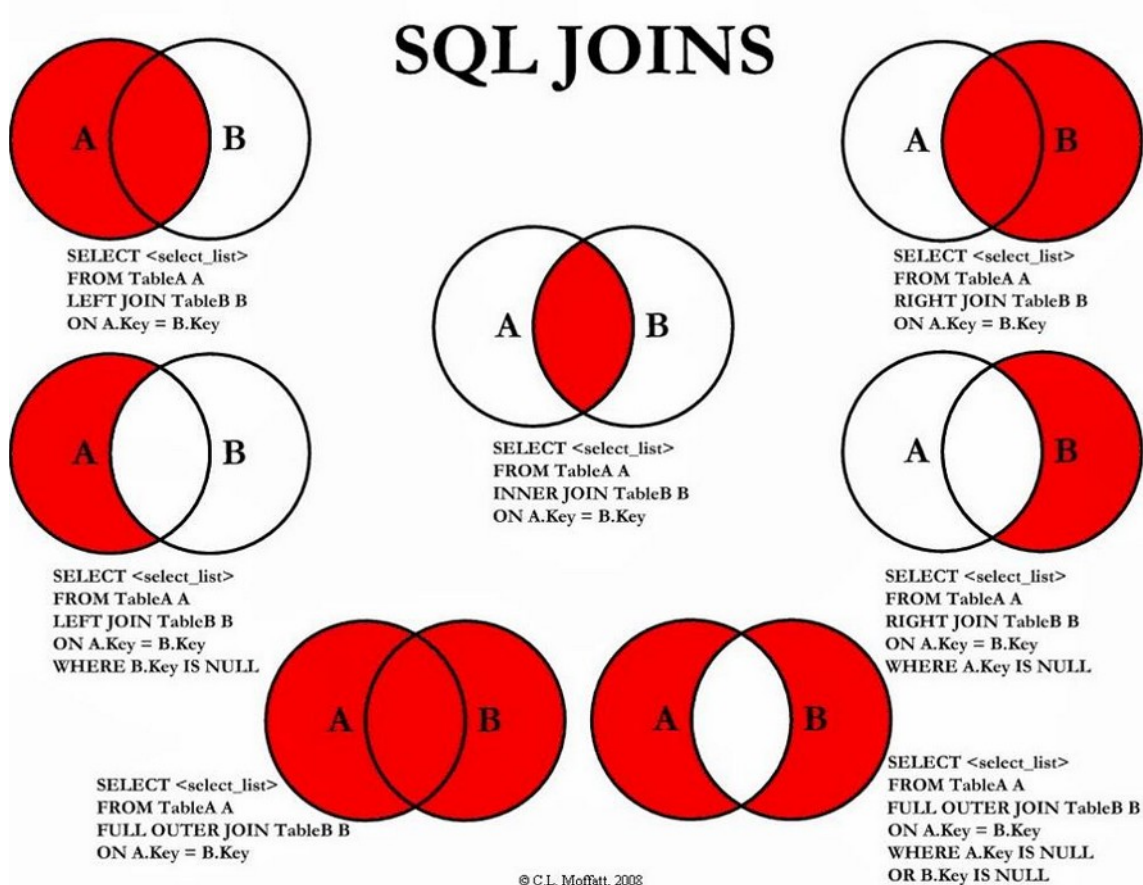
https://www.w3schools.com/sql/sql_join_left.asp

https://www.w3schools.com/sql/sql_join_right.asp

https://www.w3schools.com/sql/sql_join_full.asp

https://www.w3schools.com/sql/sql_join_self.asp

https://sqlzoo.net/wiki/Running_Total



Obrázok 6: SQL joins

Pre kreslenie E-R diagramov použite papierový zošit alebo niektorú aplikáciu, napr.: Dia³. Zvoľte kategóriu značiek: UML. Viac o E-R diagramoch a UML diagramoch je v kapitole 4.1.

Úlohy:

- Použite <https://sqlzoo.net>, napíšte príkaz, ktorý nájde názov najstaršieho filmu, v ktorom hrali Uma Thurman a Ethan Hawke. Je to pekný film.
- V databáze <https://sqlzoo.net> chýba sci-fi film, v ktorom hral Harrison Ford. Režisér je ten istý ako vo filme Alien. Film má pokračovanie, ktoré nedávno bolo v kinách. Aký je názov filmu a od koho je filmová hudba?
- Pre ktorého surrealistického režiséra robí Angelo Badalamenti aj skupina Rammstein filmovú hudbu?

2.7 UNION

https://sqlzoo.net/wiki/Combine_queries

https://www.w3schools.com/sql/sql_union.asp

2.8 CREATE TABLE

https://sqlzoo.net/wiki/CREATE_TABLE

<https://sqlzoo.net/wiki/DROP>

https://www.w3schools.com/sql/sql_create_table.asp

https://www.w3schools.com/sql/sql_primarykey.asp

https://www.w3schools.com/sql/sql_foreignkey.asp

<http://sqlfiddle.com>

2.9 INSERT

https://sqlzoo.net/wiki/INSERT_.._VALUES

https://www.w3schools.com/sql/sql_insert.asp

https://www.w3schools.com/sql/sql_insert_into_select.asp

<http://sqlfiddle.com>

Úloha:

- Vytvorte SQL skript pre databázu: Svoj aktuálny týždenný rozvrh, ktorý obsahuje aj skutočný čas obeda. Databáza má minimálne 1 tabuľku, môže mať viac.

Poznámka: SQL skript je textový súbor s príponou **.sql**, v ktorom je správna postupnosť príkazov CREATE, INSERT, SELECT a iných.

2.10 ALTER TABLE

<https://sqlzoo.net/wiki/ALTER>

https://www.w3schools.com/sql/sql_alter.asp

<http://sqlfiddle.com>

2.11 DROP TABLE

https://sqlzoo.net/wiki/CREATE_and_DROP_Reference

https://www.w3schools.com/sql/sql_drop_table.asp

<http://sqlfiddle.com>

³ <http://dia-installer.de>

2.12 UPDATE

Vkladaný údaj je konštanta -> jednoduchý príkaz.

Vkladaný údaj je výsledok z vnoreného príkazu SELECT -> niekedy nastáva problém.

https://sqlzoo.net/wiki/INSERT_SELECT

<https://sqlzoo.net/wiki/UPDATE>

https://www.w3schools.com/sql/sql_update.asp

<http://sqlfiddle.com>

Príklad na CREATE, INSERT:

```
CREATE TABLE games(yr INT, city VARCHAR(10));
CREATE TABLE world(country VARCHAR(10), capital VARCHAR(10));

INSERT INTO games VALUES(2000, 'Sydney');
INSERT INTO games VALUES(2004, 'Athens');
INSERT INTO world VALUES('Slovakia', 'Bratislava');
```

Príklad na UPDATE + konštanta:

```
UPDATE games SET city='Bratislava' WHERE yr=2000;
```

Príklad na UPDATE + SELECT na rôznych tabuľkách:

```
UPDATE games SET city=(SELECT capital FROM world WHERE country='Slovakia')
WHERE yr=2000;
```

Príklad na UPDATE + SELECT na rovnakej tabuľke:
(nefunguje, lebo nemožno čítať a zapisovať naraz)

```
UPDATE games SET city=(SELECT city FROM games WHERE yr=2004) WHERE yr=2000;
```

Tabuľku treba premenovať (aj tak nefunguje):

```
UPDATE games x SET x.city=(SELECT y.city FROM games y WHERE y.yr=2004)
WHERE x.yr=2000;
```

Treba zmeniť poradie v príkaze:

```
UPDATE games x, (SELECT city FROM games WHERE yr=2004) AS y SET x.city=y.city
WHERE x.yr=2000;
```

Alebo zmeniť SELECT IN SELECT -> SELECT JOIN:

```
UPDATE games x JOIN games y SET x.city=y.city WHERE x.yr=2000 AND y.yr=2004;
```

2.13 DELETE

<https://sqlzoo.net/wiki/DELETE>

https://www.w3schools.com/sql/sql_delete.asp

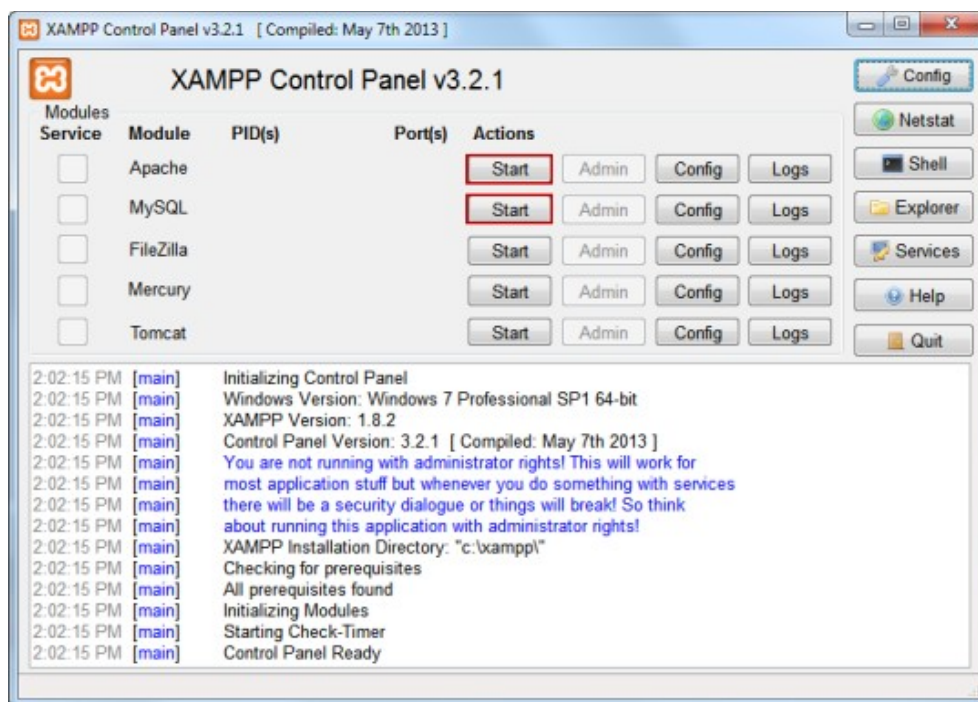
<http://sqlfiddle.com>

2.14 CREATE DATABASE, CREATE USER, GRANT

https://www.w3schools.com/sql/sql_create_db.asp

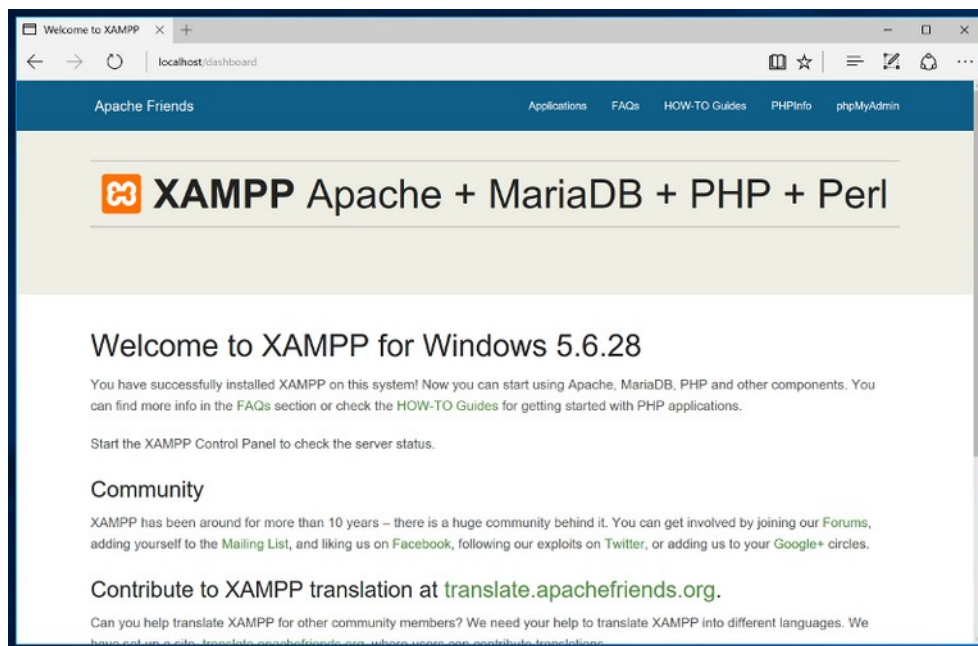
Potrebujeme nainštalovaný databázový server a vhodný nástroj na správu databáz.

- ✓ **XAMPP**⁴ – balík najmä pre MS Windows.
- ✓ **WAMP**⁵ – podobný balík najmä pre MS Windows.
- ✓ **MAMP**⁶ – podobný balík najmä pre Apple macOS:



Obrázok 7: XAMPP control panel: Start: Apache, MySQL

Apache a MySQL sa zmenia na zeleno. Do webového prehliadača zadáme adresu **localhost**.⁷



Obrázok 8: XAMPP: localhost

⁴ <https://www.apachefriends.org/download.html>

⁵ <https://www.wampserver.com/en/download-wampserver-64bits/>

⁶ <https://www.mamp.info/en/downloads/>

⁷ Skype musí byť vypnutý, lebo používa TCP port 80.

Napravo hore je odkaz na **phpMyAdmin**, alebo zadáme adresu: **localhost/phpmyadmin**:



Obrázok 9: phpMyAdmin: login

Inštalácia XAMPP vo Windows vytvorí administrátora MySQL (*root*) bez hesla. Verejne prístupný počítač má mať heslo pre každého používateľa. Návod pre vytvorenie novej databázy a používateľa tejto databázy:

[http://webvaultwiki.com.au/\(S\(iungfwenvz2f5c55pioeinil\)X\(1\)\)/Create-Mysql-Database-User-Phpmyadmin.ashx](http://webvaultwiki.com.au/(S(iungfwenvz2f5c55pioeinil)X(1))/Create-Mysql-Database-User-Phpmyadmin.ashx)

Návod pre vytvorenie databázy a používateľa v SQL príkazovom riadku:

<https://www.mysqltutorial.org/mysql-create-user.aspx>

Úloha:

- Vytvorte databázu zamestnancov firmy s 2 tabuľkami, kde každý zamestnanec je v aspoň jednom oddelení firmy.

2.15 DROP DATABASE, DROP USER

https://www.w3schools.com/sql/sql_drop_db.asp

2.16 TRANSACTION

<https://sqlzoo.net/wiki/Transactions>

Transakcia je skupina databázových operácií, ktoré musia byť vykonané úspešne ako celok. Finančná (banková) transakcia sa skladá aspoň z dvoch operácií: odčítanie danej hodnoty z jedného účtu a pripočítanie tejto hodnoty na druhý účet.

Úloha:

- Vytvorte databázu pre obchod s aspoň tromi tabuľkami: tovar, sklad, pokladňa. Pri nákupe v obchode sa čítajú dáta o tovare, odpočítajú sa kupované kusy zo skladu, pripočíta sa suma do pokladne. Nákup musí byť naprogramovaný ako transakcia.

2.17 TRIGGER

<https://www.w3resource.com/mysql/mysql-triggers.php>

Úloha:

- V databáze pre obchod pridajte kontrolu (akciu) pre záporný stav zásob.

2.18 CONSTRAINT

[https://sqlzoo.net/wiki/ALTER TABLE ... ADD constraint](https://sqlzoo.net/wiki/ALTER_TABLE_..._ADD_constraint)

https://www.w3schools.com/sql/sql_constraints.asp

Úloha:

- V databáze pre obchod pridajte obmedzenia pre hodnoty stavu zásob.
- V databáze pre obchod pridajte prednastavené hodnoty.

2.19 Projektová úloha

Na tejto úlohe si overíme vedomosti z predošlých kapitol a tvorivosť.

Navrhните relačnú SQL databázu na ľubovoľnú tému. Databáza nech spĺňa tieto podmienky:

- Databáza má aspoň 3 tabuľky.
- Každá tabuľka má aspoň 3 stĺpce (atribúty).
- Každá tabuľka má aspoň 10 riadkov (záznamov).
- Dáta v tabuľkách sú atomické (rozdelené do stĺpcov na najmenšie zmysluplné časti).
- V databáze sú použité aspoň 3 rôzne dátové typy.
- Databáza má nastavené vhodné kódovanie znakov (*utf8_general*).
- V databáze sú aspoň 2 prepojenia tabuliek pomocou primárnych a cudzích kľúčov.

Riešenie projektu môže byť overené na ľubovoľnom databázovom softvéri. Má tieto časti:

- Názov databázy a stručný slovný popis témy (max. 5 viet).
- E-R diagram (*UML class diagram*) vo forme obrázku, vytvorenom vhodným nástrojom.
- SQL skript (*database-name.sql*) na vytvorenie, naplnenie a zobrazenie databázy (CREATE DATABASE, CREATE TABLE, INSERT, SELECT).

3 Dáta v aplikáciách

3.1 Ochrana osobných údajov (GDPR)

- ✓ **Osobný údaj** je akýkoľvek údaj, ktorý čiastočne alebo úplne jednoznačne identifikuje (alebo môže identifikovať) konkrétnu osobu. Osobné údaje sú napríklad: meno, priezvisko, adresa, telefón, e-mail, používaná IP adresa, dátum narodenia, vek, pohlavie, všeobecný identifikátor (rodné číslo), fotografia, biometrické údaje, charakteristika osoby.

Aplikácia má zbierať len tie osobné údaje, ktoré sú nevyhnutné pre jej účel. Musí sa rešpektovať právo na súkromie používateľa. Osobné údaje sa smú spracúvať len na základe zákonného právneho základu⁸ (podľa osobitného zákona, so súhlasom osoby, pre plnenie zmluvy, pre oprávnený záujem spracovateľa, pre verejný záujem, na ochranu života, zdravia, majetku). Prísnejšiu ochranu majú osobitné kategórie osobných údajov, napr. rasa, etnikum, politické názory, náboženstvo a údaje biometrické, genetické, sexuálne, zdravotné.

- ✓ **GDPR** (*general data protection regulation*) je všeobecné nariadenie o ochrane osobných údajov platné pre EÚ⁹.

3.2 PHP a dáta (SQL)

- ✓ **Statická webová stránka** – webový server iba poskytuje na disku uložené súbory HTML/XHTML, CSS, JS a ďalšie multimedialne formáty pre webového klienta.
- ✓ **Dynamická webová stránka** – webový server vytvára alebo mení HTML dokument na požiadanie od klienta. Napríklad cestovný poriadok, obchodný register, sociálne siete.
- ✓ **PHP** (*personal home page, hypertext preprocessor*) – programovací jazyk pre webový server, v aktuálnej verzii 7.

Pre dynamickú stránku potrebujeme programovací jazyk na strane servera. Takých je veľa: PHP, Python, Perl, JavaScript, Java, Ruby, ASP, ASP.NET atď. Nám stačí poznať len princípy.

Ďalej potrebujeme softvér pre webový server, ktorý poskytuje webové stránky pre klientov a vie pracovať so zvoleným programovacím jazykom. Najčastejšie používaným webovým serverom je Apache httpd. Iné webové servery: Apache Tomcat, Nginx, MS IIS. Softvér môžeme inštalovať osobitne, alebo spolu ako jeden balík. Dobrou voľbou je balík XAMPP alebo WAMP alebo MAMP. Obsahuje nástroj **phpMyAdmin**, ktorý sme už použili pri jazyku SQL.

Úloha:

- Vyskúšajte si webový server. Do webového prehliadača zadajte adresu **localhost** a niečo sa zobrazí. Svoje súbory môžete ukladať vo Windows obyčajne do: **C:\XAMPP\htdocs**

Súbor HTML sa musí začínať a končiť týmito riadkami:

```
<!DOCTYPE html>
<html lang="sk">
...
</html>
```

index.html

Súbor PHP sa musí začínať a končiť týmito riadkami:

```
<?php
...
?>
```

index.php

⁸ Zákon č. 18/2018 Z. z. o ochrane osobných údajov,

<https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2018/18/201901>

⁹ Nariadenie Európskeho parlamentu a Rady (EÚ) 2016/679 o ochrane fyzických osôb pri spracúvaní osobných údajov a o voľnom pohybe takýchto údajov,

<https://eur-lex.europa.eu/legal-content/SK/TXT/PDF/?uri=CELEX:32016R0679&from=SK>

Príkaz **echo** dáva výstup vo formáte TXT bez štýlu, aby ho prehliadač zobrazil:

```
<?php                                     index.php
    echo "Ahoj";
?>
```

Webová stránka so štýlom v jazyku PHP používa jazyk HTML ako textový výstup príkazu **echo**:

```
<?php                                     index.php
    echo '<!DOCTYPE html>';
    echo "\n";
    echo '<html lang="sk">';
    echo "\n";
    echo '...';
?>
```

Značka **\n** je zalomenie riadku v zdrojovom kóde HTML zobrazenom v prehliadači. Inak by celý kód bol v jednom riadku. Značka **\t** je tabulátor. Text v príkaze echo musí byť zabalený do úvodzoviek alebo apostrofov. Ak v texte už sú nejaké úvodzovky, použijú sa apostrofy.

Jazyk HTML môže byť vnorený do PHP:

```
<?php ?>                                     index.php
<!DOCTYPE html>
<html lang="sk">
...
</html>
<?php ?>
```

Jazyk PHP môže byť vnorený do HTML:

```
<?php ?>                                     index.php
<!DOCTYPE html>
<html lang="sk">
...
    <?php ... ?>
...
</html>
<?php ?>
```

Upozornenie: Ak prehliadač zobrazí prázdnu PHP stránku, je chyba v kóde – treba pozrieť do error logu Apache! (XAMPP control panel, alebo GNU/Linux: /var/log/).

Keď je webová stránka písaná v jazyku PHP, môžeme ju zložiť z viacerých súborov, aby sa ten istý kód neopakoval vo viacerých súboroch – príkazom **include** alebo **require**:

```
<?php                                     index.php
    include "header.php";
    include "main.php";
    include "footer.php";
?>
```

Návody:

<https://www.w3schools.com/php/default.asp>

<https://www.php.net/manual/en/index.php>

PHP + SQL

Jazyk SQL sa môže vložiť do PHP. Pripojenie PHP k databáze je zabezpečené vhodnou funkciou. Balík XAMPP je dobrou voľbou, lebo obsahuje Apache, PHP, MySQL/MariaDB, phpmyadmin.

PHP sa môže používať procedurálne/funkcionálne:

```
<?php                                                                    index.php
    $host='localhost';
    $dbuser='user';
    $password='password';
    $db='database';
    $dbconnection=mysqli_connect($host, $dbuser, $password, $db);
    if($dbconnection) {
        mysqli_set_charset($dbconnection, "utf8");
        $query="SELECT * FROM table";
        if(($result=mysqli_query($dbconnection, $query))!=0) {
            $row=implode(mysqli_fetch_assoc($result));
            echo $row;
        } else {
            echo "SQL error";
        }
    } else {
        echo "DB connect error: " . mysqli_connect_error();
    }
    mysqli_close($dbconnection);
?>
```

PHP sa môže používať objektovo:

```
<?php                                                                    index.php
    $host='localhost';
    $dbuser='user';
    $password='password';
    $db='database';
    $dbobject=new mysqli($host, $dbuser, $password, $db);
    if($dbobject->connect_errno==0) {
        $dbobject->set_charset("utf8");
        $query="SELECT * FROM table";
        if($result=$dbobject->query($query)) {
            $row=implode($result->fetch_assoc());
            echo $row;
            $result->free;
        } else {
            echo "SQL error";
        }
    } else {
        echo "DB connect error: " . $dbobject->connect_error;
    }
    $dbobject->close();
?>
```

Textová odpoveď SQL v \$result alebo \$row môže byť pole.

PHP súbor je obvyčajne čitateľný z webu. Ak obsahuje heslo, je to nebezpečné. Heslo má byť v súbore, ktorý nie je prístupný z webu, ale len lokálne na disku pre Apache. Prístup k súboru môže byť chránený cez .htaccess, alebo v operačnom systéme (chown, chmod).

```
<?php
    $host='localhost';
    $dbuser='user';
    $passowrd='password';
    $db='db';
?>
```

config.php

```
<?php
    require "config.php";
    ...
?>
```

index.php

Návody:

https://www.w3schools.com/php/php_mysql_select.asp

<https://www.php.net/manual/en/book.mysql.php>

<https://www.php.net/manual/en/book.mysql.php>

Úloha:

- Vytvorte webovú stránku pomocou PHP pre polročné zadanie z SQL (alebo podobné). Textový výstup z SQL SELECT, ktorý sa dal pozrieť v phpmyadmin alebo sqlfiddle nech je teraz HTML tabuľka v prehliadači.

3.3 PHP a dáta (TXT)

Úloha:

- Vytvorte nejaký TXT súbor a zobrazte ho pomocou PHP:

```
<?php
    $textfile="subor.txt";
    $text=file($textfile);
    echo $text;
?>
```

main.php

Ak prekáža znak konca riadku, použite:

```
$text=file($textfile, FILE_IGNORE_NEW_LINES);
```

Ak sa zobrazí: *Array*, tak premenná *\$text* neobsahuje len jeden riadok, ale viac riadkov, čiže pole.

```
<?php
    $textfile="subor.txt";
    $text=file($textfile);
    foreach($text as $line) {
        echo "$line" . "<br />";
    }
?>
```

main.php

Úloha:

- Vytvorte jednoduchú tabuľku s textom a číslami v tabuľkovom procesore (napr. LibreOffice Calc) a uložte súbor vo formáte CSV.
- Vytvorte jednoduchú webovú stránku v PHP a zobrazte CSV súbor vo forme HTML tabuľky s rámčekmi.

CSV súbor je pole riadkov. Riadok je pole po stĺpcoch. Rozložte CSV súbor do riadkov a stĺpcov:

```
<?php
    $textfile="subor.csv";
    $n=3;
    $text=file($textfile);
    foreach($text as $line) {
        $field=explode(",", $line, $n);
        for($i=0; $i<$n; $i++) {
            echo "$field[$i]" . "\t";
        }
        echo "<br />";
    }
?>
```

main.php

3.4 Dátové formáty TXT a CSV

Otázky:

- Zachová dané kódovanie znakov v textovom súbore farbu textu?
- Dá sa súbor **password.txt** zobrazit' vo webovom prehliadači?
- Je CSV súbor vhodným riešením pre prenos rozvrhu hodín jednej triedy z tabuľkového editora do webovej stránky školy?

Zamyslenie:

- Predstavte si, že cez prázdniny bude medzinárodný tábor stredoškolákov. Informácie o tábore budú na webe vo viacerých jazykoch. Vedeli by ste naprogramovať diskusné fórum tak, aby sa správy dali písať zaradom v rôznych jazykoch (napr. slovensky, anglicky, čínsky, arabsky)? Teraz sa budeme venovať kódovaniu znakov a textovým súborom.

Otázka:

- Čo je kódovanie znakov?

Úlohy:

- Pozrite si tabuľku znakov ASCII na nasledujúcom obrázku. Na webe vyhľadajte informácie o kódovaní UTF-8. Porovnajte ASCII a UTF-8. Prečo je UTF-8 lepšie?

ASCII control characters			ASCII printable characters					
00	NULL	(Null character)	32	space	64	@	96	`
01	SOH	(Start of Header)	33	!	65	A	97	a
02	STX	(Start of Text)	34	"	66	B	98	b
03	ETX	(End of Text)	35	#	67	C	99	c
04	EOT	(End of Trans.)	36	\$	68	D	100	d
05	ENQ	(Enquiry)	37	%	69	E	101	e
06	ACK	(Acknowledgement)	38	&	70	F	102	f
07	BEL	(Bell)	39	'	71	G	103	g
08	BS	(Backspace)	40	(	72	H	104	h
09	HT	(Horizontal Tab)	41	)	73	I	105	i
10	LF	(Line feed)	42	*	74	J	106	j
11	VT	(Vertical Tab)	43	+	75	K	107	k
12	FF	(Form feed)	44	,	76	L	108	l
13	CR	(Carriage return)	45	-	77	M	109	m
14	SO	(Shift Out)	46	.	78	N	110	n
15	SI	(Shift In)	47	/	79	O	111	o
16	DLE	(Data link escape)	48	0	80	P	112	p
17	DC1	(Device control 1)	49	1	81	Q	113	q
18	DC2	(Device control 2)	50	2	82	R	114	r
19	DC3	(Device control 3)	51	3	83	S	115	s
20	DC4	(Device control 4)	52	4	84	T	116	t
21	NAK	(Negative acknowl.)	53	5	85	U	117	u
22	SYN	(Synchronous idle)	54	6	86	V	118	v
23	ETB	(End of trans. block)	55	7	87	W	119	w
24	CAN	(Cancel)	56	8	88	X	120	x
25	EM	(End of medium)	57	9	89	Y	121	y
26	SUB	(Substitute)	58	:	90	Z	122	z
27	ESC	(Escape)	59	;	91	[	123	{
28	FS	(File separator)	60	<	92	\	124	
29	GS	(Group separator)	61	=	93	]	125	}
30	RS	(Record separator)	62	>	94	^	126	~
31	US	(Unit separator)	63	?	95	_		
127	DEL	(Delete)						

Obrázok 10: ASCII table

- V ľubovoľnom textovom editore vytvorte súbor **súbor.txt** s krátkym slovenským textom (napr. text tejto úlohy) a pozrite si ho vo webovom prehliadači. V menu webového prehliadača nájdite možnosť zmeny kódovania a vyskúšajte si, ako sa menia znaky pri zmene kódovania.

- Uložte súbor subor.txt v textovom editore v kódovaní UTF-8 a znovu si ho pozrite vo webovom prehliadači. Mal by sa zobraziť správne pri výbere kódovania UTF-8.

Poznámka: Niektoré textové editory neponúkajú na výber kódovanie. V operačnom systéme MS Windows vyskúšajte napr.: Notepad++, PSPad, LibreOffice/OpenOffice Writer. V operačnom systéme GNU/Linux vyskúšajte napr.: xed, gedit, Bluefish, LibreOffice/OpenOffice Writer.

- V tabuľkovom editore/kalkulátore (napr. LibreOffice/OpenOffice Calc) vytvorte malú tabuľku s textom a číslami, ale bez vzorcov (napr. spolužiaci – meno, vek, výška). Uložte ju vo formáte CSV (tabulka.csv). Súbor zobrazte v textovom editore alebo vo webovom prehliadači.

Poznámka: Textový editor a webový prehliadač nemusia poznať formát CSV, stačí súbor premenovať na tabulka.txt a bude sa dať zobraziť. Je súbor uložený v kódovaní UTF-8?

Vysvetlenie

UTF-8 je štandard pre kódovanie znakov textových súborov. UTF-8 vychádza z kódovania ASCII pre prvých 128 znakov. Ďalšie znaky sú národné (aj s diakritikou) pre takmer všetky jazyky sveta, s variabilnou dĺžkou 1 až 4 B. UTF-8 sa používa aj pri vytváraní webových stránok (napr. HTML).

Formát TXT, tiež označovaný ako MIME type: text/plain, je štandard pre textové súbory pre čistý text bez štýlu. Odporúčané kódovanie je UTF-8, ktoré je kompatibilné s kódovaním ASCII pre prvých 128 znakov. Formát TXT je zobraziteľný a editovateľný v takmer každej aplikácii určenej pre text, preto by sa mal prednostne používať vždy, keď potrebujeme zapísať text bez štýlu.

Formát CSV, tiež označovaný ako MIME type: text/csv, je štandard (RFC odporúčanie¹⁰) pre tabuľkové dáta uložené do textového súboru. Štandard CSV má povinné kódovanie znakov ASCII alebo UTF-8.

Úlohy:

- Vytvorte tabuľku v nejakom tabuľkovom editore/kalkulátore, uložte ho vo formáte CSV, preneste na iný počítač a otvorte v inom tabuľkovom editore/kalkulátore.
- Porovnajte textové súbory vytvorené v operačných systémoch MS Windows a GNU/Linux. V akom znaku sa líšia bez ohľadu na kódovanie?
- Ktoré kódovanie znakov je štandardom pre nové súbory TXT, CSV, HTML?
- Prepíšte nasledujúce údaje do formátu CSV (stačí najjednoduchší zápis):

číslo dňa	deň
1	pondelok
2	utorok

10 https://sk.wikipedia.org/wiki/Comma-separated_values
<https://www.ietf.org/rfc/rfc4180.txt>
<https://tools.ietf.org/html/rfc4180>

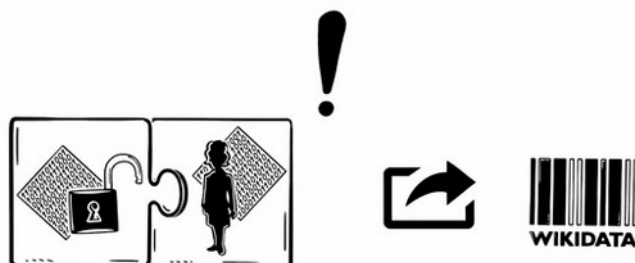
3.5 Dátový formát JSON

- ✓ **JSON** (*JavaScript object notation*) – dátový formát pre štruktúrované dáta uložené v textovom súbore.

Úlohy:

- Pozrite si video:

Open data – explained in a nutshell, 2:03, <https://www.youtube.com/watch?v=c42QNa-rccw>



What is Open Data? County of..., 3:02, <https://www.youtube.com/watch?v=PzWpcVzuwV0>



- Preskúmajte dostupné dáta (datasety) z centrálneho portálu otvorených dát Vlády SR:
<https://data.gov.sk>
- Nájdite webové sídlo vášho mesta alebo obce, zistite, či má zverejnené otvorené dáta. Dobrý príklad: mesto Prešov.
- Preskúmajte dostupné dáta (datasety) z európskeho portálu:
<https://data.europa.eu/euodp/en/home/>
<https://www.europeandataportal.eu/>
- Vyberte si niektorý zaujímavý dataset, pri ktorom je viac dostupných formátov súborov. Na tejto vyučovacej hodine nás zaujímajú formáty: CSV, XLSX, JSON. Obsah súborov sa dá vidieť cez náhľad vo webovom prehliadači. Stiahnite formáty CSV, XLSX, JSON k jednému datasetu a otvorte ich vo vhodných aplikáciách. Zapište názov datasetu.
- V akých aplikáciách ste otvorili formáty CSV, XLSX, JSON?
- Aké kódovanie znakov je použité v textových súboroch?
- Popíšte úryvok z obsahu súboru CSV a úryvok z obsahu súboru JSON.
- Čo ste sa dozvedeli z datasetu?
- Nájdite na disku počítača súbor formátu JSON (*.json), otvorte ho vo vhodnej aplikácii. Zapište názov súboru aj s cestou a stručne opíšte obsah súboru.

Vysvetlenie

Koncept otvorených dát má dôležité uplatnenie v IT a v správe vecí verejných, v otvorenej spoločnosti. Filozofia otvorenej občianskej spoločnosti: čo nie je tajné, je verejné a zverejnené. Dobre pripravené otvorené dáta sú ľahko vyhľadateľné, štruktúrované a ľahko strojovo spracovateľné. Medzi populárne formáty ukladania otvorených dát patria: CSV, JSON, XML.

Formát JSON, tiež označovaný ako MIME type: application/json, sa používa v otvorených dátach, aj v rôznych typoch softvéru na uloženie používateľských dát a nastavení. JSON je textový súbor so štandardným kódovaním znakov UTF-8, so štruktúrou zápisu podobnou ku kódu v programovacom jazyku Javascript.

Úlohy:

- Preskúmajte dostupné dáta (datasets) z portálu Wikidata:
https://www.wikidata.org/wiki/Wikidata:Main_Page
- Aká licencia je použitá pri zverejnení otvorených dát na portáli Wikidata a na iných centrálnych portáloch otvorených dát?
- Zoznámte sa s verejnou licenciou Creative Commons:
<https://sk.creativecommons.org>
- Je verejná licencia nutným predpokladom pre otvorené dáta?
- Je strojová spracovateľnosť dôležitá pre otvorené dáta?
- Používa sa formát JSON aj na ukladanie používateľských dát a nastavení v aplikáciách?

3.6 Dátový formát XML

- ✓ **XML** (*extensible markup language*) – rozšíriteľný značkovací jazyk, dátový formát pre štruktúrované dáta uložené v textovom súbore.

Úlohy:

- Pozrite si video

Anglicky, 2:41, <https://www.youtube.com/watch?v=WXD0UP2aDlw>



- Akým spôsobom ste doteraz riešili problémy pri prenose dokumentov medzi rôznymi verziami aplikačného softvéru alebo medzi rôznymi operačnými systémami, najmä u dokumentov Office?
- Preskúmajte dostupné dáta (datasets) z niektorého centrálného portálu otvorených dát. Vyberte si zaujímavý dataset, pri ktorom je viac dostupných formátov. Na tejto vyučovacej hodine nás zaujíma formát XML, na minulých to boli CSS a JSON. Stiahnite súbory a zapíšte si hypertextový odkaz na dataset. Portály otvorených dát:
Mesto Prešov: <https://egov.presov.sk> (vľavo dole Katalóg OPEN DATA)
Prešovský samosprávny kraj: <https://www.po-kraj.sk/sk/open-data/>
SR: <https://data.gov.sk> (napravo Datasets, potom naľavo dole Formáty)
EÚ: <https://www.europeandataportal.eu/sk>
- Zobrazte súbor XML vo webovom prehliadači. Zobrazte zdrojový kód stránky. Aké kódovanie znakov je použité?
- Prezrite si štruktúru jazyka XML. Čo ste sa dozvedeli z datasetu (o akej téme je dataset)?
- Aká licencia sa často používa pri zverejnení otvorených dát?
- Vo webovom prehliadači zobrazte ľubovoľnú webovú stránku. Zobrazte zdrojový kód stránky. Aké sú rozdiely medzi jazykmi XML, XHTML, HTML?
- Vytvorte veľmi jednoduchú webovú stránku (jeden súbor) v jazyku XHTML 5 tak, aby v ňom neboli XML chyby, ani HTML chyby, ani CSS chyby. Všimnite si, že webový prehliadač umožní zobrazíť iba bezchybný kód podľa štandardu XML.
- Nájdite na disku nejaký XML súbor. Ku ktorej aplikácii patrí?
- V kancelárskom balíku vytvorte ľubovoľný krátky dokument. Uložte súbor v niektorom z týchto formátov: text ODT alebo DOCX, tabuľka ODS alebo XLSX, prezentácia ODP alebo PPTX. Potom v operačnom systéme premenujte tento súbor na **súbor.zip**, rozbaľte súbor a prezrite si, aké súbory sú vo vnútri. Sú tam súbory XML? Zobrazte tieto súbory vo webovom prehliadači. Nájdite súbor, ktorý má uložený obsah pôvodného dokumentu a súbor, ktorý má uložený štýl pôvodného dokumentu.

Vysvetlenie

Medzi populárne formáty ukladania otvorených dát patria: CSV, JSON, XML. Formát XML, tiež označovaný ako *MIME type: text/xml, application/xml*, je štandard pre ukladanie a výmenu štruktúrovaných dát. XML je textový súbor so štandardným kódovaním znakov UTF-8, v ktorom je čitateľná štruktúra dát.

Webové stránky, webové aplikácie a webové služby môžu byť vytvárané/programované vo viacerých jazykoch, aj v jazyku XML. Jazyk XHTML 5 je aplikáciou (podmnožinou) jazyka XML. Jazyk HTML 5 je aplikáciou (podmnožinou) jazyka SGML, čo je predchodca XML. Tieto jazyky slúžia na zápis štruktúry obsahu webového dokumentu. Iné jazyky sú použité na zápis štýlu pre webový dokument.

- ✓ **REST** (*representational state transfer*) je softvérová architektúra pre prepojenie webových služieb pomocou výmeny stavových informácií najčastejšie vo formáte XML. Použitie napríklad: jednotné prihlásenie do viacerých webových služieb jedného poskytovateľa (webmail, chat, disk, mapa).

Formát XML sa používa na ukladanie štruktúrovaných používateľských dát a nastavení v rôznych typoch softvéru, napríklad v dokumentoch ODF (ODT, ODS, ODP) alebo MS OOXML (DOCX, XLSX, PPTX). V štruktúre týchto dokumentov je osobitne uložený obsah, osobitne štýl a osobitne nastavenia.

Úlohy:

- Pozrite si video

Anglicky, 2:13, <https://www.youtube.com/watch?v=MQkRfdKBB58>



- Stručne opíšte použitie XML a JSON pre Android.
- Bonusová nepovinná otázka: Aké máte plány o svojej budúcnosti v IT?

Literatúra:

<https://sk.wikipedia.org/wiki/XML>

<https://en.wikipedia.org/wiki/XML>

<https://sk.wikipedia.org/wiki/XHTML>

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

<https://sk.wikipedia.org/wiki/OpenDocument>

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

XML 1.0 a XHTML 5

Zjednodušene možno napísať: XHTML 5 = XML + HTML 5

Aby prehliadač použil XML parser, lokálny súbor má mať príponu .xml alebo .xhtml a zdrojový kód musí mať XML deklaráciu. Inak sa použije SGML parser a prehliadač číta kód ako HTML, nie ako XHTML. Aby prehliadač čítal webový obsah vo verzii HTML 5, musí zdrojový kód obsahovať HTML deklaráciu. Deklarácia bez čísla verzie znamená poslednú dostupnú verziu, tá je aktuálne 5, staršie verzie sa označujú číslom. Keďže XML deklarácia obsahuje kódovanie znakov, aj HTML kód musí obsahovať meta dáta o rovnakom kódovaní znakov. Dvojmo sa zapisuje ešte jazyk obsahu. Vzory XML, HTML 5, XHTML 5:

```
<?xml version="1.0" encoding="utf-8"?>
<root>
</root>
```

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8" />
    <title>Vzor HTML 5</title>
  </head>
  <body>
    <h1>Vzor HTML 5</h1>
  </body>
</html>
```

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html>
<html lang="sk" xml:lang="sk" xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta charset="utf-8" />
    <title>Vzor XHTML 5</title>
  </head>
  <body>
    <h1>Vzor XHTML 5</h1>
  </body>
</html>
```

Odporúčanie W3C pre tvorbu nových stránok je: použiť štandard HTML 5 alebo XHTML 5, pričom zdrojový kód by sa mal medzi nimi líšiť len o deklaráciu XML a súvisiace atribúty. Ostatné riadky majú byť napísané tak, že prejdú validátorom pre HTML 5 aj pre XHTML 5. Inými slovami: aj keď nejaké HTML pravidlo dovoľuje veľa, iné pravidlo môže protirečiť. Treba vyhovieť všetkým pravidlám. Kód treba skontrolovať vhodným validátorom:

<https://xmlvalidation.com>

<https://validator.w3.org>

<http://jigsaw.w3.org/css-validator/>

Webový prehliadač cez jeho XML parser zobrazí iba bezchybný kód XML alebo XHTML, inak oznámi chybu na danom riadku.

3.7 Prehľad portálov otvorených dát

Úloha:

➤ Prezrite si portály otvorených dát:

Mesto Prešov: <https://egov.presov.sk> (vľavo dole Katalóg OPEN DATA)

Prešovský samosprávny kraj: <https://www.po-kraj.sk/sk/open-data/>

SR: <https://data.gov.sk> (napravo Datasets, potom naľavo dole Formáty)

EÚ: <https://www.europeandataportal.eu/sk>

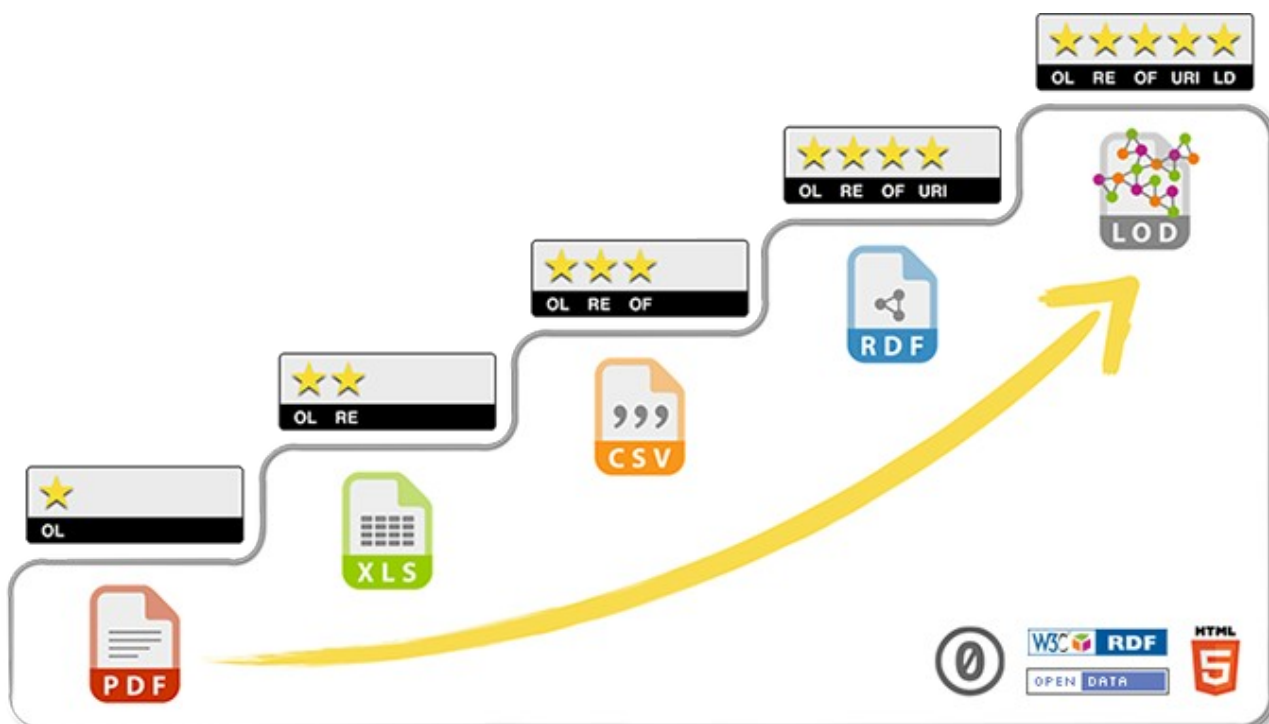
Princípy zverejňovania otvorených dát:

- Čo nie je tajné, je verejné. Verejné je zároveň platené z verejných peňazí.
- Čo je verejné, je zverejnené. Zverejnené má verejnú licenciu.

✓ **Otvorené dáta** sú štruktúrované strojovo spracovateľné údaje, zverejnené s verejnou licenciou, obyčajne v štandardizovanom formáte.

Tim Berners-Lee, vynálezca webu a iniciátor prepojených dát, navrhol hodnotenie otvorených dát hviezdikami (<https://5stardata.info/cs/>):

- * zverejnené dáta s otvorenou (verejnou) licenciou (vo webovom prostredí)
- ** + štruktúrované strojovo spracovateľné dáta (čitateľný formát)
- *** + otvorený štandardizovaný formát (CSV, JSON, XML, HTML/XHTML)
- **** + permanentná URI adresa (permanentný referencovateľný identifikátor)
- ***** + prepojené dáta (prepojené cez referencovateľný identifikátor na iné dáta)



Obrázok 11: Hodnotenie otvorených dát hviezdikami

Na štátnych portáloch by mali byť datasety s minimálne 3 hviezdikami.

Spracovanie TXT/CSV súborov v príkazovom riadku (bash)

Stručný prehľad príkazov pre príkazový riadok operačného systému GNU/Linux (bash)¹¹:

<code>man <command></code>	<i>manual</i> – zobrazenie pomoci k príkazu
<code>echo <string></code>	vypíš textový reťazec na obrazovku
<code>echo <string> > <file></code>	vypíš textový reťazec do súboru (prepíš celý súbor)
<code>echo <string> >> <file></code>	vypíš textový reťazec do súboru (pridaj na koniec)
<code>echo \${<variable>}</code>	vypíš hodnotu premennej
<code><command1> <command2></code>	presmeruj výstup z príkazu 1 do príkazu 2 (<i>pipe</i>)
<code>ls</code>	<i>list</i> – vypíš obsah aktuálneho priečinka
<code>cp <file1> <file2></code>	<i>copy file</i> – kopíruj súbor 1 na súbor 2
<code>mv <file1> <file2></code>	<i>move file</i> – presuň súbor 1 na súbor 2
<code>cat <file></code>	<i>concatenate files</i> – vypíš textový súbor na obrazovku celý, pospájaj viaceré súbory za sebou
<code>join -j <key> <file1> <file2></code>	<i>join files</i> – vypíš dva súbory spojené kľúčom
<code>grep <pattern> <file></code>	hľadaj textový vzor v súbore, vypíš nájdený riadok na obrazovku (textový filter)
<code>grep -E <pattern> <file></code>	<i>global regular expression print</i> – hľadaj (rozšírený) regulárny výraz v textovom súbore
<code>egrep <pattern> <file></code>	<i>extended global regular expression print</i>
<code>sort <file></code>	<i>sort</i> – usporiadaj riadky v súbore
<code>uniq</code>	<i>unique</i> – vypíš len jedinečné riadky (idúce po sebe)
<code>wc <file></code>	<i>word count</i> – spočítaj riadky, slová, bajty v súbore
<code>cut</code>	<i>cut</i> – vystrihni časť textu v riadku
<code>tr</code>	<i>translate or delete characters</i> – nahraď/zmaž znaky
<code>expr <expression></code>	<i>expression</i> – vyhodnoť matematický výraz
<code>let <variable>=<expression></code>	vyhodnoť matematický výraz, prirad do premennej
<code>test <expression></code>	vyhodnoť výraz, testuj znakový reťazec alebo súbor
<code>[<expression>]</code>	vyhodnoť výraz, testuj znakový reťazec alebo súbor
<code>[[<expression>]]</code>	vyhodnoť výraz s textovým reťazcom
<code>Ctrl+C (^C)</code>	<i>interrupt, cancel</i> – preruš vykonávaný príkaz

Stručný prehľad použitia špeciálnych znakov:

<code>'text'</code>	znakový reťazec
<code>"text"</code>	znakový reťazec, ale platia špeciálne znaky \ \$ `
<code>\char</code>	<i>printable character</i> – tlačiteľný znak
<code>`command`</code>	vykonanie príkazu, zobrazí sa návratová hodnota
<code>\$(command)</code>	vykonanie príkazu (<i>bash</i>)
<code>command; command</code>	oddelenie viacerých príkazov v riadku
<code>\$((<expression>))</code>	vyhodnotenie matematického výrazu (<i>bash</i>)

¹¹ Operačné systémy (GNU/Linux), <https://www.shenk.sk/skola/informatika/operacne-systemy-gnu-linux.pdf>

Nasledujúce úlohy robte v operačnom systéme GNU/Linux:

- Vytvorte TXT súbor s kódovaním UTF-8. Dá sa do tohto súboru zapísať text so štýlom?
- Čím sa líši TXT súbor vytvorený v MS Windows od súboru vytvoreného v GNU/Linux?
- Vytvorte CSV súbor na ľubovoľnú tému. Nech je to prepojený dataset na iné datasety, ktorý bude obsahovať aspoň 3 stĺpce (identifikátor, URI, názov) a aspoň 3 riadky (hocijaké zverejnené datasety otvorených dát vo formáte CSV).

Séria úloh na jednom datasete:

- Vyhľadajte aktuálny CSV dataset otvorených dát: *Register nehnuteľných národných kultúrnych pamiatok* pre svoj kraj¹².
- Pomocou príkazu **wget** alebo **curl** stiahnite tento dataset.
- Premenujte stiahnutý súbor na **register.csv** v príkazovom riadku.
- Zistite, koľko má KiB.
- Príkazom vypíšte zo súboru všetky záznamy o cintorínoch.
- Príkazom zistite počet všetkých záznamov o hradoch v súbore.
- Príkazom vypíšte zo súboru len záznamy pre svoj okres.
- Príkazom vypíšte abecedne usporiadaný zoznam okresov v kraji.
- Príkazom vypíšte abecedne usporiadaný zoznam obcí v kraji. Ku každej obci priradte okres.
- Príkazom zistite, koľko obcí v kraji má rovnaký názov.
- Príkazom vypíšte zoznam okresných miest v kraji, ku každému mestu okres.
- Príkazom zistite počet kultúrnych pamiatok pre každý okres v kraji.
- Príkazom zistite priemerný počet kultúrnych pamiatok pre jednu obec v kraji.
- Príkazom zistite priemerný počet kultúrnych pamiatok pre jeden okres v kraji.
- Príkazom rozdeľte súbor **register.csv** do osobitných súborov pre okresy v kraji.
- Príkazom (alebo skriptom) vypíšte písmená abecedy, každé na samostatnom riadku. Ku každému písmenu vypíšte početnosť výskytu písmena v celom zozname obcí kraja. Pre lepšiu názornosť pridajte histogram – graf početností, s mierkou od 0 do 10 bodiek, každá bodka za celých 10 % z maximálnej hodnoty vo výpise.

12 napr. <https://data.gov.sk/dataset/a9b82093-4d0c-4b94-bc9a-7eb8ba7af181/resource/643fa0b7-5415-4137-88be-c8283102ef65/download/nkpaktualizacia20190620zoznamnhnkpokraj.csv>

3.8 Analýza dát v tabuľkovom procesore

Úlohy:

- Stiahnite si CSV dataset *Počet prístupov na ÚPVS za deň* (<https://data.gov.sk/dataset/upvs-web-pristupy-za-den>). Každý použije jeden súbor, s dátami za iný mesiac. II.SA má 30 žiakov. Podľa abecedného poradia 1. použije súbor za január 2017 a tak ďalej, až 30. použije súbor za jún 2019.
- Použite tabuľkový procesor (LibreOffice Calc, OpenOffice.org Calc, MS Excel, iný). Importujte stiahnutý CSV súbor so správne zvoleným kódovaním, ukončením riadka, oddeľovačom buniek.
Upozornenie: MS Excel niekedy nesprávne rozpozna CSV súbor alebo ho nechce otvoriť. CSV formát vytvorený v MS Excel nespĺňa CSV štandard (RFC odporúčanie).
- Doplňte do tabuľky vzorce/funkcie pre výpočet štatistických hodnôt: počet záznamov, suma, minimum, maximum, priemer, medián, štandardná odchýlka.
- Ktoré dni v týždni sú najmenej vyťažené?
- Vytvorte graf počtu prístupov podľa dňa, aj zaznačený priemer. Zvoľte vhodný typ grafu.
- Vytvorte PDF súbor s 1 stranou A4, kde bude graf a podstatné informácie o dátach.

Vysvetlenie

ÚPVS (Ústredný portál verejnej správy, <https://www.slovensko.sk>) poskytuje elektronickú komunikáciu so štátnou/verejnou správou, najmä cez HTML formuláre. A to pomocou elektronického občianskeho preukazu s čítačkou kariet, ktorý slúži na prihlásenie, podpísanie, šifrovanie.

Pri zverejňovaní súborov pre štátnu/verejnú správu platí predpis o štandardoch¹³, kde sú popísané štandardné formáty, štandardné písma, prístupnosť webu atď. Štandardný formát pre zverejňovanie je napríklad:

- finálny textový dokument TXT, HTML/XHTML, PDF (môže obsahovať text, tabuľku, graf, obrázok, hypertextový odkaz, ale nie skenovaný obsah ako obrázok, obsah musí byť strojovo čitateľný)
- editovateľný textový dokument TXT, HTML/XHTML (ak taký formát vyhovuje účelu)
- editovateľný textový dokument ODT a DOCX, a to vždy dvojmo v oboch formátoch, aby bola zaistená kompatibilita medzi rôznymi softvérmi
- editovateľný tabuľkový dokument CSV, XML, JSON
- editovateľný tabuľkový dokument ODS a XLSX, a to vždy dvojmo v oboch formátoch, aby bola zaistená kompatibilita medzi rôznymi softvérmi
- obrázok, video, audio: JPG, PNG, GIF, OGG, MP2, MP4, WAV
- komprimovaný súbor ZIP
- iný formát, na ktorom sa obe strany komunikácie dohodnú

¹³ Vyhláška o štandardoch pre informačné technológie verejnej správy, <https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2020/78/20200501>

Formát CSV umožňuje uložiť len číselné a textové dáta. Nezachovávajú sa: vzorce, funkcie, grafické formátovanie, graf. Preto sme použili formát ODS/XLSX. V bunke mali byť zapísané požadované vzorce alebo funkcie, nie samotné čísla. Najlepšie je použiť rozsah vstupných buniek. Príklad:

=MAX(C2:C32)

Štatistické funkcie COUNT alebo ROWS, MIN, MAX, SUM, AVERAGE sú bežné.

Matematické/štatistické stredné hodnoty (mean) sú rôzne:

- aritmetický priemer (AVERAGE)
- geometrický priemer
- harmonický priemer
- medián (MEDIAN) delí usporiadanú množinu prvkov na dve polovice s rovnakým počtom, s hodnotou menšou alebo väčšou ako medián
- modulus je najčastejšie vyskytujúca sa hodnota v množine

Štandardná odchýlka (STDEV) alebo ešte rozptyl (variance) ukazujú, ako na široko sú hodnoty od priemeru. Existuje viacero verzií týchto funkcií, v danej úlohe na tom nezáleží.

Najmenej vyťažený deň je sobota. Druhý najmenej vyťažený deň je nedeľa, ľudia často dokončujú, čo nestihli. Málo vyťažené dni sú počas sviatkov a veľmi vyťažené dni sú pondelky a tesne pred termínom nejakej povinnosti (napr. podanie daňového priznania). To spravidla platí pre hocikaké informačné systémy.

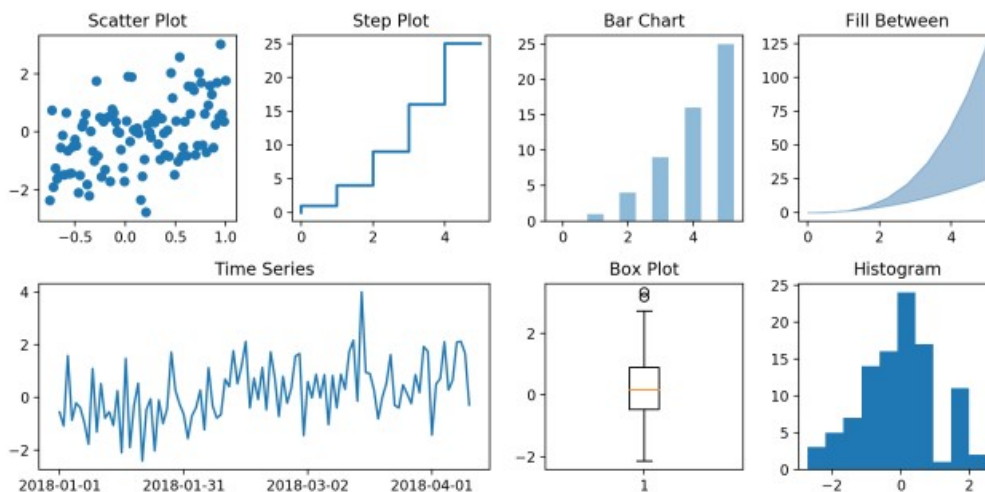
Vhodné typy grafu pre zobrazenie počtu podľa dátumu: stĺpcový, čiarový, xy-bodový. Správne ste určili dátum ako os x (nezávislá premenná) a počet ako os y (závislá premenná). Kvôli čitateľnosti sa môže dátum umiestniť k osi šikmo. V grafe mal byť odlišne vyznačený priemer, najlepšie vodorovnou čiarou, alebo stĺpcom inej farby.

ODS/XLSX súbor je pracovný, vzhľad tabuľky môže byť hocikaký, ale ľahko čitateľný, takže ku vzorcom/funkciám treba pridať popis do vedľajšej bunky.

PDF súbor je finálny, určený pre čitateľa. Graf by mal mať názov a legendu. Podstatná informácia je zdroj dát (URI datasetu), k tomu niektoré štatistické hodnoty. PDF súbor určený pre iného čitateľa by mal mať meno autora, dátum vytvorenia.

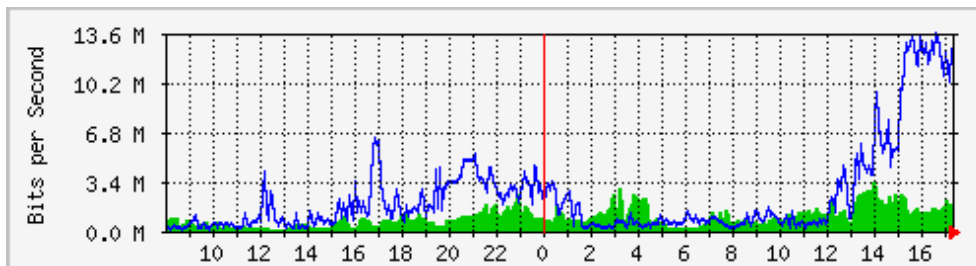
3.9 Vizualizácia číselných dát

- ✓ **Tabuľkový kalkulačtor** (LibreOffice, OpenOffice, MS Office alebo iný) vie vytvoriť pekné grafy niekoľkých typov. Ale keď chceme ešte iné typy, alebo automatizovaný graf, animovaný graf, interaktívny graf, treba použiť iný nástroj.
- ✓ **Webová aplikácia Show My Data** <https://www.showmydata.org>
- ✓ **Python** umožňuje vytvárať grafy z číselných dát, podobne aj iné programovacie jazyky:



Obrázok 12: Python: Matplotlib

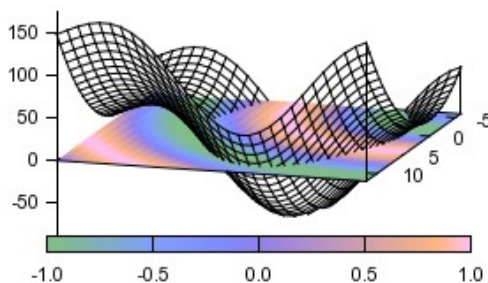
- ✓ **MRTG** (*multi router traffic grapher*) je automat, ktorý číta systémové záznamy (logy) zo sieťových adaptérov a pravidelne generuje graf vo forme PNG obrázku:



Obrázok 13: MRTG

Klikateľná mapa SANET (<http://samon.cvt.stuba.sk>)

- ✓ **Gnuplot** je bezplatný čiastočne otvorený softvér na vizualizáciu 2D a 3D číselných dát:



Obrázok 14: Gnuplot

<https://en.wikipedia.org/wiki/Gnuplot>, <http://www.gnuplot.info>

<https://sourceforge.net/projects/gnuplot/files/gnuplot/5.2.8/>

- ✓ **Aplikácie pre dátovú analytiku**

Grafana (<https://grafana.com>, <https://en.wikipedia.org/wiki/Grafana>)

KNIME (<https://www.knime.com>, <https://en.wikipedia.org/wiki/KNIME>)

Úlohy:

- Nainštalujte si Gnuplot (<https://sourceforge.net/projects/gnuplot/files/gnuplot/5.2.8/>). Funguje v MS Windows, GNU/Linux, MacOS. V niektorých distribúciách GNU/Linux je dostupný balík gnuplot priamo v správcovi softvéru, netreba ho ručne sťahovať.
- Pozrite si video návod:
gnuplot Installation and graph plotting tutorial on (Linux/Ubuntu), 4:28
<https://www.youtube.com/watch?v=wtuo3xEcReM>
- Spustíte gnuplot. Otvorí sa príkazový riadok.
- Vyskúšajte zobraziť niekoľko 2D grafov zadaním funkcie:

```
plot x  
plot x+10  
plot 10*x  
plot x**2  
plot sin(x)
```

- Gnuplot sa končí príkazom quit alebo exit.
- V príkazovom riadku prejdite do priečinka, kde máte súbor CSV z minulého týždňa (<https://data.gov.sk/dataset/upvs-web-pristupy-za-den>). Spustíte gnuplot.
- Zobrazte CSV súbor:

```
plot "subor.csv"
```

Gnuplot zrejme zahlási chybu. Prvý riadok CSV súboru je hlavička, ktorá by mala začínať znakom #, aby sa chápala ako komentár. Opravte riadok, alebo ho vymažte. Ďalej posledný riadok môže byť zlý, lebo nie je tam deň, ale súčet za mesiac. Vymažte ho. Ďalej gnuplot predpokladá, že oddeľovací znak medzi stĺpcami je medzera. Môžete súbor opraviť, alebo nastaviť správny oddeľovač:

```
set datafile separator ","
```

Ak je v súbore viac stĺpcov, treba určiť, ktoré stĺpce sa majú zobraziť v bodovom grafe:

```
plot "subor.csv" using 1:3
```

Čiarový graf, stĺpcový graf:

```
plot "subor.csv" using 1:3 with lines  
plot "subor.csv" using 1:3 with boxes
```

- Gnuplot umožňuje zmeniť typ grafu, mierku, popisy osí, názov, farby a plno iných vlastností. Príkazy sa dajú napísať do skriptu (súboru). Pozrite si návod s ukážkami grafov:
<http://gnuplot.sourceforge.net/demo/>
- Pozrite si tvorbu animovaného GIF z gnuplot:
<http://www.gnuplotting.org/tag/animation/>
- Vytvorte nový graf z CSV súboru z minulého týždňa tak, aby bol farebný a názornejší. V okne zobrazeného grafu v menu hore vľavo sa dá graf uložiť do súboru ako obrázok.

3.10 Veľké dáta

Doteraz sme sa zaoberali databázami a datasetmi s malým množstvom dát, ktoré je možné spracovať na bežnom počítači. Veľké dáta vyžadujú veľké úložisko a veľký výpočtový výkon. Urobme si rekapituláciu pojmov a obsahu predmetu DSY, aby predmet dával hlbší zmysel:

- ✓ **Dáta (údaje)** – postupnosť znakov (symbolov) určená na spracovanie počítačom.
- ✓ **Databáza** – množina štruktúrovaných údajov uložená v počítačovom systéme, ktorý umožňuje automatizované spracovanie týchto údajov.
- ✓ **OLTP** – *Online transaction processing* – spracovanie databázových požiadaviek v nedeliteľných skupinách (transakciách), aby bolo zaručené, že sa vykoná celá séria príkazov alebo žiadny z nich. Príklad: banková transakcia je odpísanie sumy z jedného účtu a pripísanie sumy na druhý účet.
- ✓ **Dataset** – kolekcia štruktúrovaných strojovo spracovateľných údajov, obvyčajne v štandardizovanom formáte. Štandardizovaný formát: napr. CSV, JSON, XML.
- ✓ **Dátový kontajner** – abstraktný dátový typ, dátová štruktúra alebo súbor, ktorého obsahom je kolekcia štruktúrovaných dát. Dátový kontajner: CSV, JSON, XML, komprimovaný XML (ZIP).
- ✓ **Metadáta** – štruktúrované údaje o primárnych údajoch. To je opis obsahu datasetu.
- ✓ **Otvorené dáta** – štruktúrované strojovo spracovateľné údaje, zverejnené s verejnou licenciou, obvyčajne v štandardizovanom formáte.
- ✓ **Prepojené dáta** – distribuované štruktúrované strojovo spracovateľné údaje, obvyčajne v štandardizovanom formáte, zviazané vhodnou metódou alebo aplikačným protokolom.
- ✓ **Veľké dáta** – kolekcia údajov, ktorá je príliš rozsiahla alebo komplexná na bežné spracovanie.
- ✓ **NoSQL** – databázový systém pre dáta, ktoré nie sú štruktúrované do relačnej (tabuľkovej) formy. Obvyčajne sa používa pre veľké dáta alebo dáta s prevažne binárnymi dátovými typmi.
- ✓ **Dolovanie dát (dátová analytika)** – hĺbková analýza údajov, najmä hľadanie trendov a vzorov. Textové dáta sa filtrujú, usporadúvajú, spočítavajú sa riadky. Číselné dáta sa vizualizujú grafmi, s vyznačením štatistických hodnôt (min, max, average...). Trend je napr. rast alebo pokles podľa zistenej matematickej funkcie (lineárnej, kvadratickej, exponenciálnej...). Vzor je opakovanie sa priebehu hodnôt v čase alebo podľa matematickej funkcie (sin). Ďalej sa zisťujú súvislosti a závislosti medzi stĺpcami datasetu. Výpočtová náročnosť dolovania dát sa zvyšuje s množstvom dát a viac-rozmernosťou datasetu.
- ✓ **OLAP** – *Online analytical processing* – automatizovaná dátová analytika, ktorá na pozadí vypočítava často žiadané funkcie, aby človek-používateľ dostal odpoveď rýchlo, v reálnom čase.
- ✓ **Dátová veda** – interdisciplinárny odbor zaoberajúci sa získavaním znalostí z údajov, kombinujúci informatiku, analýzu, štatistiku, strojové učenie.
- ✓ **Informatika** – veda o získavaní, ukladaní a automatizovanom (počítačovom) spracovaní digitálnych dát.
- ✓ **Informácia** – číselné vyjadrenie množstva dát. Informácia sa udáva v bitoch (b). Digitálne dáta tvoria postupnosť číslíc. Ak sa použije číselný základ 2, tak ide o číslice 0 a 1. Násobky pre číselný základ 2 sú definované v norme IEC 60 027-2, napr. Ki = 1024.

Zdroje:

<https://terminologickyportal.sk/wiki/Kategória:It>

<https://sk.wikipedia.org>, <https://en.wikipedia.org>

3.11 Java a dáta (TXT, CSV, JSON, XML)

Java + TXT

Aplikácia programovaná v jazyku Java môže čítať alebo zapisovať súbory v rôznych formátoch. Je dobré využiť knižničné funkcie pre konkrétny formát. Najjednoduchší formát súboru je TXT.

Čítať a zapisovať textový súbor sa dá viacerými spôsobmi, napr.:

```
import java.io.*;
public class Subor {
    public static void main(String[] args) throws IOException {
        FileReader in=null;
        FileWriter out=null;
        try {
            in = new FileReader("in.txt");
            out = new FileWriter("out.txt");
            int c;
            while((c=in.read())!=-1) {
                out.write(c);
            }
        } finally {
            if(in!=null) in.close();
            if(out!=null) out.close();
        }
    }
}
```

```
import java.io.*;
public class Subor {
    public static void main(String[] args) {
        try {
            String text="Ahoj";
            FileWriter out = new FileWriter("out.txt");
            BufferedWriter buf = new BufferedWriter(out);
            buf.write(text);
            buf.close();
        } catch(IOException e) {
            e.printStackTrace();
        }
    }
}
```

Návody:

<https://www.tutorialspoint.com/java/index.htm>

<https://www.javacodegeeks.com>

<https://www.oracle.com/java/technologies/>

Úloha:

- Vytvorte textový súbor **in.txt**, v ktorom bude niekoľko riadkov. Vytvorte program **Reverse.java**, ktorý načíta vstupný súbor **in.txt**, otočí poradie riadkov a zapíše ich do výstupného súboru **out.txt**.

Java + CSV

Čítanie CSV súboru je v princípe rovnaké ako čítanie TXT súboru, ale dáta treba rozložiť do riadkov a riadky do stĺpcov podľa daného znaku (.). Postup je podobný ako v PHP.

Návody:

<https://howtodoinjava.com/java/io/parse-csv-files-in-java/>

<https://mkyong.com/java/how-to-read-and-parse-csv-file-in-java/>

<https://www.baeldung.com/java-csv-file-array>

Úloha:

- Vytvorte Java aplikáciu, ktorá zobrazí CSV súbor ako tabuľku. Použite znova CSV dataset prístupov na portál slovensko.sk z predošlých týždňov.

Java + JSON

JSON je zložitejší formát textového súboru. Je vhodné využiť knižnicu.

Návody:

<https://www.geeksforgeeks.org/parse-json-java/>

<https://stackoverflow.com/questions/2591098/how-to-parse-json-in-java>

Java + XML

XML je ešte zložitejší formát textového súboru.

Návody:

https://www.tutorialspoint.com/java_xml/java_dom_parse_document.htm

<https://mkyong.com/java/how-to-read-xml-file-in-java-dom-parser/>

<https://howtodoinjava.com/xml/read-xml-dom-parser-example/>

<https://www.baeldung.com/java-xml>

<https://www.viralpatel.net/java-xml-xpath-tutorial-parse-xml/>

<https://www.journaldev.com/1194/java-xpath-example-tutorial>

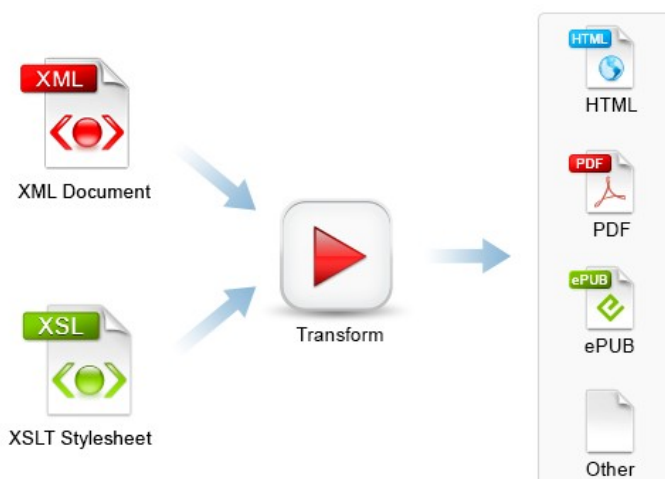
XML transformácia

Tak ako HTML dokument má štýl CSS, XML dokument má štýl CSS alebo XSL. Je možné robiť transformáciu XML formátu pomocou štýlu XSLT do iných formátov. Návody:

<https://docs.oracle.com/javase/tutorial/jaxp/xslt/transformingXML.html>

<https://gist.github.com/serge1/9307868>

<https://stackoverflow.com/questions/4604497/xslt-processing-with-java>



Obrázok 15: XML XSLT

3.12 Java a dáta (SQL, JDBC)

Aplikácia programovaná v jazyku Java môže používať databázu s jazykom SQL pomocou JDBC alebo iných knižníc a ovládačov.

✓ **JDBC** (*Java database connectivity*) – Java DB API.

Ukážka zdrojového kódu:

```
import java.sql.*;
public class DB {
    public static void main(String[] args) {
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception ex) {
        }
        Connection c = null;
        try {
            c = DriverManager.getConnection(
                "jdbc:mysql://localhost/db?user=root&password=");
            Statement s = c.createStatement();
            ResultSet r = s.executeQuery("SELECT * FROM table");
            while(r.next()) {
                System.out.print(r.getString("id"));
                System.out.print("\t");
                System.out.println(r.getString("name"));
            }
            r.close();
            c.close();
        } catch (SQLException sx) {
            System.out.println("SQLException: " + sx.getMessage());
            System.out.println("SQLState: " + sx.getSQLState());
            System.out.println("VendorError: " + sx.getErrorCode());
        }
    }
}
```

DB.java

Návody:

<https://docs.oracle.com/javase/tutorial/jdbc/basics/processingsqlstatements.html>

<https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-usagenotes-connect-drivermanager.html>

<https://mkyong.com/tutorials/jdbc-tutorials/>

<http://www.tutorialspoint.com/jdbc/jdbc-sample-code.htm>

<http://www.java2novice.com/jdbc/>

https://www.ntu.edu.sg/home/ehchua/programming/java/JDBC_Basic.html

<https://www.javacodegeeks.com>

<https://www.oracle.com/java/technologies/>

✓ **ODBC** (*open database connectivity, open and universal DB API*) – univerzálny štandard pripojenia na databázu z hocikákeho programovacieho jazyka, najmä z C++.

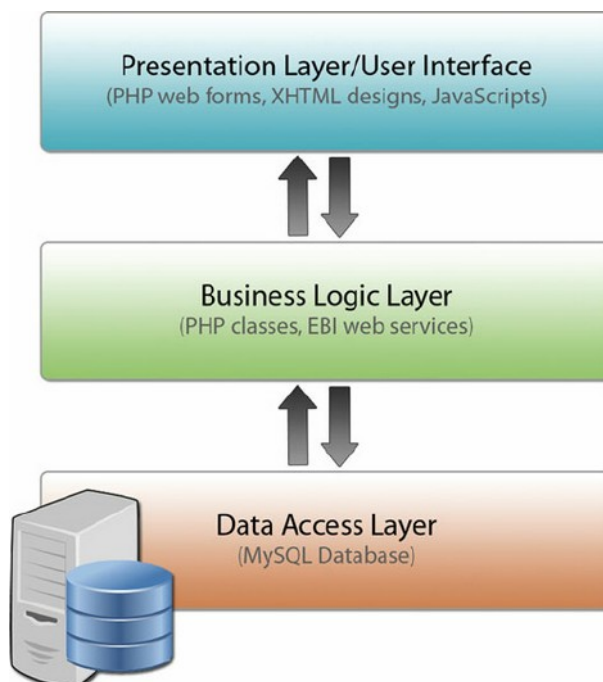
ORB (*object request broker*)

CORBA (*common object request broker architecture*)

DCOM (*distributed component object model*)

4 Objektový návrh aplikácie

Nasledujúca kapitola sa venuje objektovému návrhu aplikácie s dátami. Najprv navrhne aplikáciu pomocou diagramov. Potom naprogramuje databázu, základ aplikácie a používateľské prostredie. Toto rozdelenie je bežné označované v praxi ako 3-vrstvová architektúra aplikácie:



Obrázok 16: 3-vrstvová architektúra aplikácie

Prezentačná vrstva (UI) môže byť prispôbená pre textovú konzolu, grafickú desktopovú aplikáciu, mobilnú aplikáciu, webový prehliadač. Podľa toho sa zvolia programovacie jazyky, knižnice, operačný systém. Dátová vrstva môže byť implementovaná pre relačnú SQL databázu alebo rôzne dátové formáty uložené ako súbory (datasets).

Projektová úloha:

- ✓ Vytvorte dvojicu. Napíšte názov aplikácie a stručný slovný popis aplikácie.

Má to byť nová aplikácia, nie pokračovanie predošlých zadaní. Môže byť hra. Aplikácia má mať konzolové alebo GUI alebo webové používateľské prostredie, dáta bude ukladať do SQL databázy, nastavenie aplikácie bude v súbore CSV alebo JSON alebo XML.

Príklad pre inšpiráciu: Osobný register hudobných albumov pre viac používateľov – Java aplikácia, ktorá má dáta o hudbe uložené v lokálnej SQL databáze, mená a heslá používateľov sú uložené v lokálnom JSON súbore. Iný príklad pre inšpiráciu: Webová hra (piškvorky rozmerov $n \times m$) pre dvoch hráčov – PHP webová stránka, ktorá má dáta o stave hry, mená a heslá hráčov uložené v SQL databáze, konštanty pre hraciu plochu a dekorácie hracej plochy má uložené v CSV súbore cez PHP.

4.1 UML

Grafický jazyk UML umožňuje vytvárať diagramy viacerých typov, aby sa algoritmy dali popísať presne a úplne. Vývojový diagram je len jeden typ diagramu a nepostačuje na presný a úplný popis algoritmov. UML pomáha pri automatizácii softvérového vývoja automatickým generovaním diagramov zo zdrojového kódu, alebo automatickým generovaním zdrojového kódu z diagramov. Existuje veľa nástrojov pre UML jazyk.¹⁴ Napríklad Dia (+ dia2code, cpp2dia).

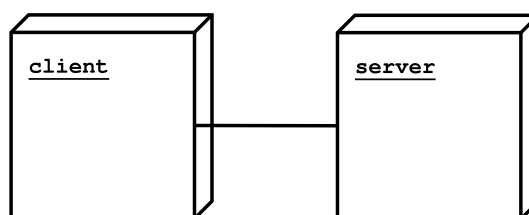
- ✓ **UML (Unified Modeling Language)**¹⁵, v aktuálnej verzii 2.5.1 z roku 2017, sa dnes používa na grafické modelovanie algoritmov. UML je štandard podľa ISO/IEC 19501:2005, s neskoršími aktualizáciami podľa novších verzií UML.
- ✓ **Vývojový diagram** sa kreslí podľa technickej normy STN 36 9030 (1989) alebo ISO 5807 (1985, 1996) pre symboly vývojových diagramov. Norma je zastaraná a prekonaná novším štandardom UML, ale nebola zrušená.

Typy UML diagramov:

- **Statický pohľad – štruktúra algoritmu:**
 - blokové diagramy (rozmiestnenie, balíčky, komponenty, zložená štruktúra)
 - diagramy objektového návrhu (entity, triedy, objekty)
- **Dynamický pohľad – správanie sa algoritmu:**
 - diagram prípadov použitia
 - diagram aktivity (riadenie, tok dát)
 - stavový diagram
 - diagram interakcie (časovanie, sekvencia, komunikácia)

Čím je algoritmus zložitejší, tým viac diagramov by sa malo použiť. Databázové a objektové programovanie potrebuje: blokový diagram, diagram tried, diagram objektov, diagram prípadov použitia, diagram aktivity. Viaceré diagramy musia súvisieť, ak majú popisovať jeden algoritmus. Súvis diagramov je daný jednotným pomenovaním funkcií, tried, stavov, činností...

Rozdelíme algoritmus do blokov:

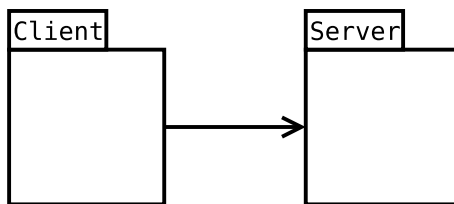


Obrázok 17: Blokový diagram:
diagram rozmiestnenia
(deployment diagram)

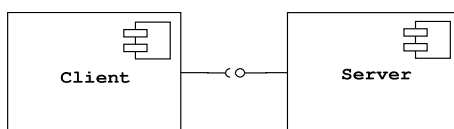
¹⁴ UML nástroje, https://en.wikipedia.org/wiki/List_of_Unified_Modeling_Language_tools

¹⁵ https://sk.wikipedia.org/wiki/Unified_Modeling_Language
https://en.wikipedia.org/wiki/Unified_Modeling_Language
<https://www.uml.org>

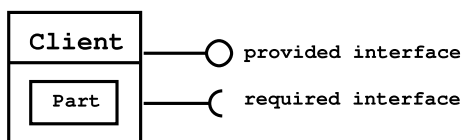
Ak blokový diagram nie je dostatočne jednoznačný, potom bloky možno chápať tiež ako balíčky, komponenty, zloženú štruktúru:



Obrázok 18: Diagram balíčkov
(package diagram)

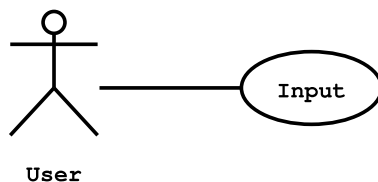


Obrázok 19: Diagram komponentov
(component diagram)



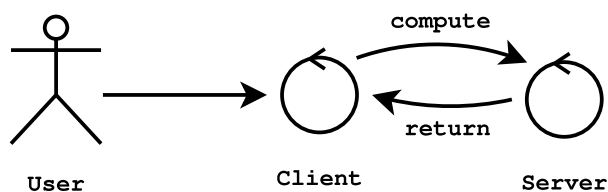
Obrázok 20: Diagram zloženej štruktúry
(composite structure diagram)

ypíšeme prípady použitia algoritmu – činnosti používateľa:



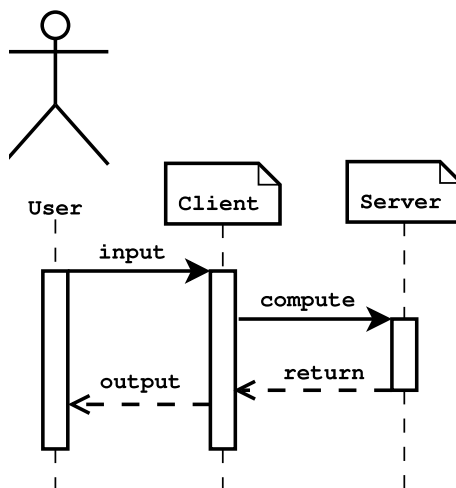
Obrázok 21: Diagram prípadov
použitia (use-case diagram)

Môžeme pridať diagram komunikácie:



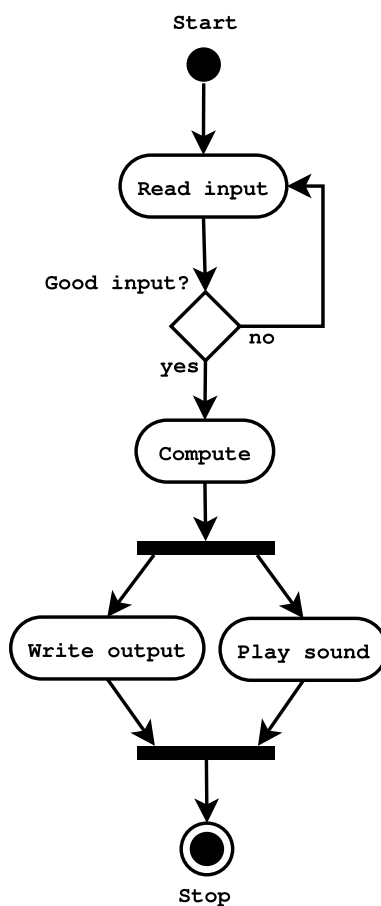
Obrázok 22: Diagram komunikácie
(communications diagram)

Popíšeme sekvenciu operácií:



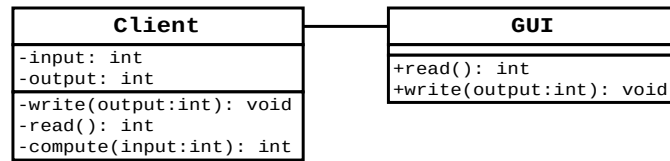
Obrázok 23: Sekvenčný diagram
(sequence diagram)

Popíšeme aktivitu algoritmu pre každý blok zvlášť. Diagram aktivity má rovnaký účel ako starý vývojový diagram. Niektoré značky sa líšia:



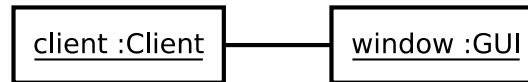
Obrázok 24: Diagram aktivity
(activity diagram)

Objektový návrh aplikácie sa má robiť tak, že aplikáciu rozdelíme do tried podľa dát, ktoré spracúvame a podľa zvolenej architektúry. Jedna trieda nestačí. Potrebujeme aspoň jednu triedu pre strednú vrstvu (*business logic layer*) a aspoň jednu triedu pre prezentačnú vrstvu (UI). Objektový návrh popíšeme diagramom tried:



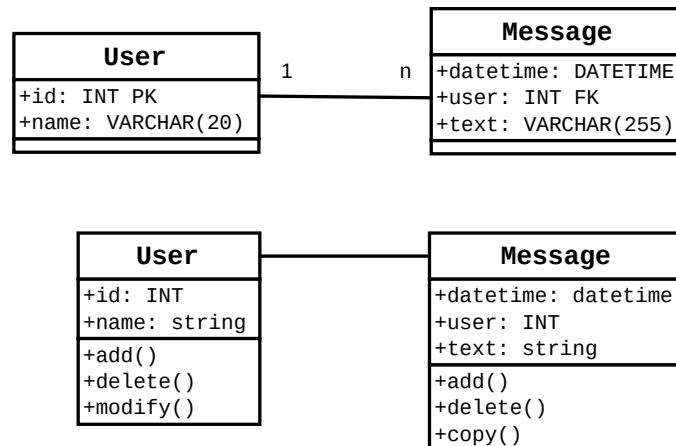
Obrázok 25: Diagram tried
(class diagram)

Z diagramu tried sa odvodí diagram objektov:



Obrázok 26: Diagram objektov
(object diagram)

Databáza sa modeluje pomocou E-R diagramu, čo je typ diagramu zhodný s diagramom tried. Odporúčanie: každá databázová tabuľka v E-R diagrame má príslušnú triedu v diagrame tried, v ktorej sú funkcie/metódy pre prácu s dátami v tej tabuľke. Databázová aplikácia má preto aspoň toľko tried, koľko je v databáze tabuliek a k tomu aspoň jednu triedu pre prezentačnú vrstvu (UI).



Obrázok 27: E-R diagram a diagram tried
(E-R diagram & class diagram)

4.2 Návrh aplikácie (UML, SQL)

Úloha:

➤ Navrhnite aplikáciu pomocou UML diagramov. Vytvorte PDF s obsahom: autor, názov aplikácie, krátky textový popis, diagramy.

Potrebuje statický pohľad (rozdelenie do blokov):

- Jeden blokový diagram, ktorý zobrazí architektúru klient-server, keďže používame klientsku aplikáciu a SQL server, webový prehliadač a webový server).
- Úplný popis SQL databázy pomocou E-R diagramu s kľúčmi a početnosťami spojení.
- Objektový návrh aplikácie pomocou diagramov tried a diagramov objektov.

Ďalej potrebujeme dynamický pohľad na aplikáciu (fungovanie v čase):

- Diagram prípadov použitia, kde sú zobrazené činnosti aplikácie a používateľa.
- Diagram aktivity (to je obvyklý vývojový diagram).

Pre kreslenie diagramov použite napr. grafický editor Dia alebo iný, zvolte katalóg UML značiek.

4.3 Implementácia aplikácie (SQL)

Úloha:

➤ Podľa E-R diagramu vytvorte SQL databázu. Príkazy uložte do SQL skriptu.

4.4 Implementácia aplikácie (PHP alebo Java)

Úloha:

➤ Podľa UML diagramov vytvorte aplikáciu v jazyku PHP alebo v jazyku Java, v ktorej sa použijú dáta z SQL databázy.

5 Dátová bezpečnosť

5.1 Zálohovanie dát

- ✓ **Zálohovanie** je prevencia pred stratou. **Obnova** je vrátenie stavu zo zálohy. Strata dát môže byť spôsobená chybou používateľa, chybou softvéru, chybou hardvéru, nedostupnosťou hardvéru, nedostupnosťou serverovej služby. Zálohovať sa dá rôzne, napr. aj obyčajným kopírovaním súborov, priečinkov, disku na iné médium. Výhodou špecializovaných nástrojov je automatizácia, načasovanie zálohovania a komprimácia zálohovaných dát.
- ✓ **Archivácia** je odloženie dát na určitý čas, zachováva históriu stavu, obyčajne v pravidelných cykloch (denná, týždenná, mesačná, ročná archivácia). Archivácia môže byť v každom cykle kompletná, alebo optimalizovaná na úložný priestor – rozdielová (prvá je kompletná a každá ďalšia je rozdiel oproti prvej) alebo inkrementálna (prvá je kompletná a každá ďalšia je rozdiel oproti predošlej). Ak archivácia duplikuje pôvodné dáta, je zároveň aj zálohou.

5.2 Zálohovanie databázy

SQL databáza počas behu má pracovné dáta v hlavnej pamäti počítača (RAM) a otvorené súbory na disku. Zálohovanie databázy z operačného systému sa robí korektne tak, že najprv sa ukončí aplikácia, vypne sa databázový systém a potom sa v operačnom systéme zálohujú súbory databázy. Niektoré databázové systémy umožňujú zálohovať bežiacu databázu tak, že sa na krátky čas databáza pozastaví a okopíruje, prerušenie aplikácie je potom kratšie. Zálohovanie obsahu databázy z konzoly administrátora databázového servera alebo z SQL aplikácie sa dá urobiť zapísaním celého obsahu databázy aj so štruktúrou vo forme SQL skriptu do súboru (*SQL dump*).

5.3 Bezpečnosť a spoľahlivosť dát databázového servera

V praxi býva zabezpečenie dát viacnásobné:

- Zálohovanie dát používateľom aplikácie (každý používateľ zálohuje svoje dáta).
- Zálohovanie databázy administrátorom databázového systému (*SQL dump*).
- Zálohovanie súborov systémovým administrátorom (súbory v operačnom systéme).
- Ochrana dát súborovým systémom (*file system: journal, snapshot, time shift, ZFS*).
- Ochrana dát diskovým polom (RAID 1, RAID 5, RAID 6).
- Ochrana dát duplikovaním hardvéru (klaster).

5.4 Oprávnenia k dátam

Každé dáta majú svojho vlastníka. Na úrovni operačného systému každý súbor má svojho vlastníka a svoju skupinu. Oprávnenia sa nastavujú pre vlastníka, skupinu a ostatných. Pre zálohované alebo archivované dáta platia rovnaké pravidlá ako pre originálne dáta.

5.5 Šifrovanie dát a podpisovanie dát

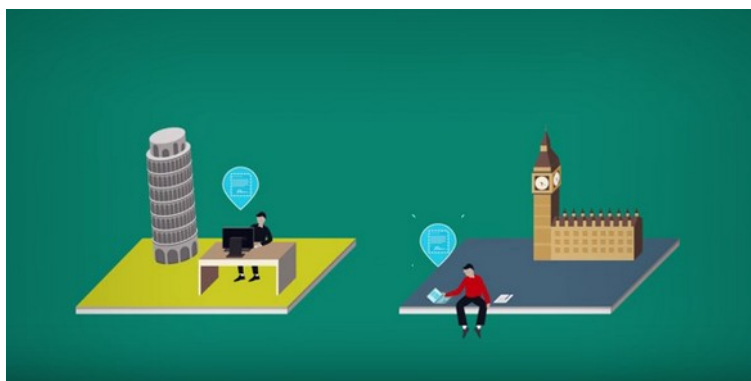
Utajenie dát pred neoprávnenými osobami sa môže dosiahnuť šifrovaním. Pravosť dát sa zabezpečuje digitálnym podpisom. Vlastník dát je držiteľom kľúča pre šifrovanie a podpisovanie.

Otázky:

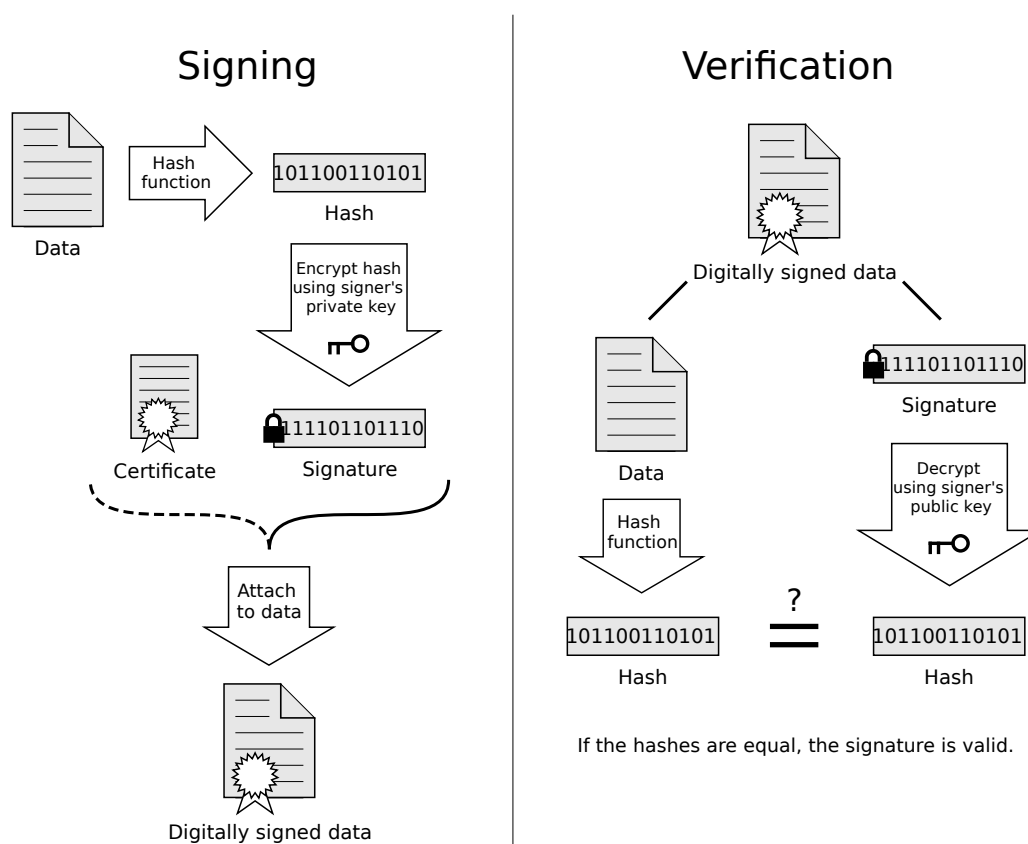
- Akú bezpečnostnú hrozbu minimalizujeme silnými heslami?
- Dostávate občas spam, ktorý na prvý pohľad vyzerá dôveryhodne?
- Ako overíte, či elektronická faktúra za pripojenie do internetu alebo mobil je pravá?

Úlohy:

- What is a digital signature? 2:01 <https://www.youtube.com/watch?v=VlcBpRpiBoc>



- Pokúste sa vysvetliť rodičom nasledujúci obrázok pre **digitálny podpis** (*digital signature*) pomocou asymetrickej kryptografie (*asymmetric cryptography*):



Obrázok 28: Digitálny podpis

- Nainštalujte slobodný softvér pre MS Windows, GNU/Linux, macOS: poštový klient Mozilla Thunderbird¹⁶, aplikáciu GPG/GnuPG/OpenPGP¹⁷ pre šifrovanie a digitálny podpis. Podľa návodu¹⁸ vytvorte poštový účet, vygenerujte šifrovacie kľúče, priradte kľúč ku poštovej adrese, napíšte správu, priložte svoj verejný kľúč, podpíšte správu a odošlite ju niekomu. Overte podpis z prijatej správy. Potom vytvorte podpísanú a šifrovanú správu a odošlite ju nazad. V aplikácii GPG je možné podpisovať a šifrovať samostatné súbory.

¹⁶ <https://www.mozilla.sk/thunderbird/>

¹⁷ <https://www.gnupg.org>

<https://www.openpgp.org>

¹⁸ <https://www.youtube.com/watch?v=vjs5RyA2DfE>

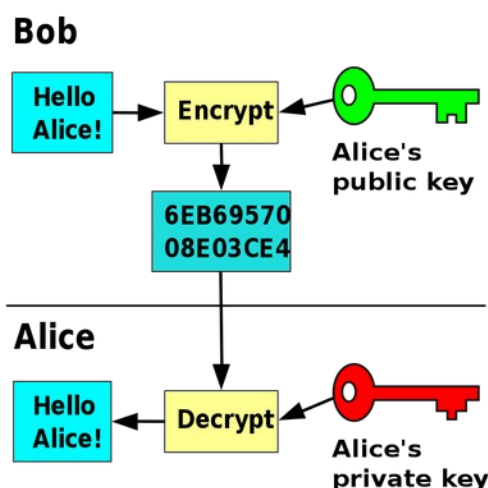
<https://www.youtube.com/watch?v=6AwrJSfV8tg>

Vysvetlenie

Gmail a iný *webmail* obyčajne neumožňuje klientske podpisovanie a šifrovanie správ, pretože *webmail* je poštová služba na vzdialenom webovom serveri. *Webmail* môže používať šifrovanie komunikácie medzi servermi. Klientske podpisovanie a šifrovanie správ si v princípe vyžaduje lokálne inštalovaného poštového klienta a lokálne uloženie kryptografických kľúčov. Potom sa takej komunikácii hovorí: *end-to-end signed & encrypted communication*. Lokálny poštový klient (Mozilla Thunderbird) sa dá pripojiť na Gmail poštovými protokolmi SMTP/IMAP.

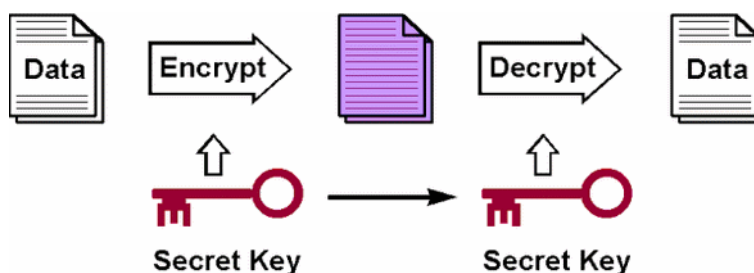
Elektronický podpis a digitálny podpis sa často používajú ako synonymá, ale je medzi nimi rozdiel.

- ✓ **Elektronický podpis**¹⁹ je informácia pripojená k elektronickému dokumentu, ktorá identifikuje podpisovateľa (fyzickú osobu), umožňuje overiť pravosť a integritu podpísaného dokumentu a platí, že iba podpisovateľ ho môže efektívne vyhotoviť. Takto funguje aj elektronické podpisovanie s občianskym preukazom (eID) pomocou čítačky čipovej karty na stránke www.slovensko.sk.
- ✓ **Digitálny podpis** je technická implementácia elektronického podpisu, ktorá využíva asymetrickú kryptografiu. Je to znakový reťazec definovanej dĺžky.
- ✓ **Asymetrická kryptografia** používa dva šifrovacie kľúče (privátny a verejný), ktorými sa vykonávajú jednosmerné operácie (zašifrovať a dešifrovať, alebo podpísať a overiť podpis). Asymetrická kryptografia závisí na veľkej matematickej/výpočtovej zložitosti uhádnutia privátneho kľúča. Asymetrická kryptografia je náročná na procesor počítača, ale je v princípe bezpečnejšia ako symetrická kryptografia.



Obrázok 29: Asymetrická kryptografia

- ✓ **Symetrická kryptografia** má na oboch stranách rovnaký kľúč a je menej náročná na procesor, používa sa najmä na šifrovaný prenos cez dlhodobý prenosový kanál (SSH), pričom najprv sa symetrický kryptografický kľúč preniesie pri vytváraní kanála pomocou asymetrickej kryptografie.



Obrázok 30: Symetrická kryptografia

¹⁹ https://terminologickyportal.sk/wiki/Termín:elektronický_podpis

Úlohy:

- Obsahuje digitálny podpis identifikáciu podpisujúceho?
- Využíva digitálny podpis asymetrickú kryptografiu?
- Používa sa verejný kľúč na digitálne podpísanie správy?
- Ktoré z uvedených skratiek sú kryptografické algoritmy symetrické a ktoré asymetrické?
DES, 3DES, AES, RSA, SHA, Blowfish, MD5, TKIP

Záver

Venovali sme sa týmto kapitolám: základné pojmy z databáz, relačná databáza a jazyk SQL, dáta v aplikáciách a dátové formáty (TXT, CSV, JSON, XML), PHP a dáta, Java a dáta.

Úlohy:

- Urobte si niektoré kvízy z SQL, XML a programovania:

https://www.w3schools.com/sql/sql_quiz.asp

https://sqlzoo.net/wiki/Tutorial_Quizzes

<http://www.sqlquiz.com>

<https://www.afterhoursprogramming.com/tests/sql/>

<https://www.javatpoint.com/sql-quiz>

<https://www.geeksforgeeks.org/dbms-gq/sql-gq/>

<https://career.guru99.com/sql-quiz/>

<http://www.bullraider.com/quiz/sql-quiz>

https://www.w3schools.com/xml/xml_quiz.asp

<http://www.academictutorials.com/quiz.asp?id=5>

<https://career.guru99.com/xml-interview-questions/>

- Pozrite si video

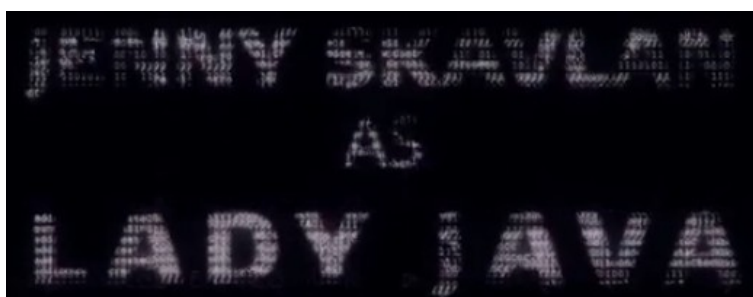
The Python programming song, 06:38, <https://www.youtube.com/watch?v=3UsKYsLSGpU>



- Zapište všetky programátorské výrazy a skratky do textového súboru **in.txt**, po jednom na riadok. Použite GNU/Linux. V príkazovom riadku usporiadajte súbor **in.txt** podľa abecedy, očísľujte riadky a zapíšte do súboru **out.txt**.

- Pozrite si video

Lady Java JavaZone, 2:17, <https://www.youtube.com/watch?v=7bRzvPpHcYM>



Zdroje obrázkov

Obrázok 1: Digitálne dáta, https://www.analyticsinsight.net/wp-content/uploads/2020/07/Data-Literacy-1.jpg	4
Obrázok 2: Zdroje dát, https://upload.wikimedia.org/wikipedia/commons/thumb/6/6d/Data_types_-_en.svg/656px-Data_types_-_en.svg.png	4
Obrázok 4: LibreOffice Calc: usporiadať a filtrovať, https://elearn.ellak.gr/pluginfile.php/4760/mod_page/content/9/sort-filter-toolbar.png	8
Obrázok 5: LibreOffice Calc: použité usporiadanie a použitý filter, https://static.thegeekstuff.com/wp-content/uploads/2011/12/calc-data-filter.png	8
Obrázok 6: SQL joins, https://3.bp.blogspot.com/-oV16uecqHDc/Va85hUqsgI/AAAAAAAAALc/W186ToftNRw/s1600/sql_joins.jpg	10
Obrázok 7: XAMPP control panel: Start: Apache, MySQL, https://devtuts.butlerccwebdev.net/testserver/xampp-control-panel.png	13
Obrázok 8: XAMPP: localhost, https://s3.amazonaws.com/webucator-how-tos/2251.png	13
Obrázok 9: phpMyAdmin: login, https://i.stack.imgur.com/prBIF.png	14
Obrázok 10: ASCII table, http://www.theasciicode.com.ar	21
Obrázok 11: Hodnotenie otvorených dát hviezdikami, https://5stardata.info/cs/	28
Obrázok 12: Python: Matplotlib, https://www.machinelearningplus.com/wp-content/uploads/2019/01/20_Histogram_Boxplot_TimeSeries_Matplotlib-min-640x320.png	33
Obrázok 13: MRTG, https://oss.oetiker.ch/mrtg/192.33.92.249_fa4_1-day.png	33
Obrázok 14: Gnuplot, http://www.gnuplot.info/figs/front2.png	33
Obrázok 15: XML XSLT, https://www.oxygenxml.com/img/ls_xslt_transform.png	37
Obrázok 16: 3-vrstvová architektúra aplikácie, https://www.researchgate.net/profile/Bikram_Parida/publication/249011985/figure/fig1/AS:298334855155712@1448139980114/Diagrammatic-representation-of-the-three-tier-architecture-implemented-in-BBProF.png	39
Obrázok 27: Digitálny podpis, https://upload.wikimedia.org/wikipedia/commons/2/2b/Digital_Signature_diagram.svg	46
Obrázok 28: Asymetrická kryptografia, https://en.wikipedia.org/wiki/File:Public_key_encryption.svg	47
Obrázok 29: Symetrická kryptografia, https://gaonkarwebsite.com/blog/wp-content/uploads/2016/02/secret-key.gif	47

Register

Algoritmus	5
ALTER TABLE	11
Archivácia	45
B	5, 6
B/s	6
Bajt	5, 6
Bash	29
Bit	5, 6
Bps	6
CONSTRAINT	15
CREATE DATABASE	13
CREATE TABLE	11
CREATE USER	13
CSV	21, 37
Dáta	5
Databáza	5
Databázový riadiaci systém	5
Databázový systém	5
DB	5
DBMS	5
DBS	5
DELETE	12
Digitálny podpis	47
DROP DATABASE	14
DROP TABLE	11
DROP USER	14
E-R diagram	5
Elektronický podpis	47
GDPR	16
Gi	6
Gnuplot	33
GRANT	13
GROUP BY	9
HAVING	9
ICT	5
IKT	5
Informácia	5
INSERT	11
IT	5
Java	36
JDBC	38
JOIN	10

JSON	23, 37
Ki	6
MAMP	13, 16
Mi	6
Obnova	45
ODBC	38
ORDER BY	9
Osobný údaj	16
Otvorené dáta	28
PHP	16
PhpMyAdmin	14, 16
Počítač	5
Podpisovanie dát	45
Program	5
Python	33
Relačná databáza	5, 9
REST	26
SELECT	9
SELECT IN SELECT	9
SQL	5, 9
Šifrovanie dát	45
Tabuľka	7
Tabuľkový editor	7
Tabuľkový kalkulátor	33
Ti	6
TRANSACTION	14
TRIGGER	14
TXT	20, 21, 36
Údaje	5
UML	40
UNION	11
UPDATE	12
UTF-8	22
Veľké dáta	35
WAMP	13, 16
WHERE	9
XAMPP	13, 16
XHTML 5	27
XML	25, 37
Zálohovanie	45
Znalosť	5