

# **Tvorba vlastní databáze s více tabulkami**

# NAVRHUJEME VLASTNÍ DATABÁZI

**Datové typy**

**<https://www.sqlite.org/datatype3.html>**

## STANDARD SQL vs SQLite

- **BOOLEAN**
  - (true/false) jako TINYINT (0, 1)
- **DATETIME**
  - jako Text (YYYY-mm-dd HH:MM:SS)
  - Unix time (od 1.1.1970) jako INTEGER
  - pro práci s datem u SQLite využíváme  
[https://www.sqlite.org/lang\\_datefunc.html](https://www.sqlite.org/lang_datefunc.html)
- **SQLITE** tedy využívá pouze (TEXT, NUMERIC, INTEGER, REAL, BLOB)
- **STANDARD** je definován zde
  - <http://www.tutorialspoint.com/sql/sql-data-types.htm>
- **POROVNÁNÍ AFFINITY TYPES**
  - [http://www.tutorialspoint.com/sqlite/sqlite\\_data\\_types.htm](http://www.tutorialspoint.com/sqlite/sqlite_data_types.htm)

## **VELIKOST DATOVÉHO TYPU**

**snažíme se volit ne zbytečně velký datový typ**

- CHAR vs VARCHAR vs TEXT
- INT vs SMALLINT vs TINYINT VS ... BIGINT
- DATETIME (jaké máme formáty ?)
- BOOLEAN vs TINYINT
- BLOB

# Tabulka

- PRIMARY KEY - unikátní klíč tabulky (počet záznamů VS primární klíč, co je větší ?)
- AUTO INCREMENT - poslední záznam + 1
- NULL ? - může, či nemůže být atribut prázdný  
IS (NOT) NULL
- UNIQUE - automatická kontrola unikátnosti atributu
- DEFAULT VALUE - typicky aktuální čas
- INDEX -

<http://www.zive.cz/clanky/sql-po-seste---pouzivejte-indexy/sc-3-a-4886/default.aspx>

# **DATOVÉ MODELOVÁNÍ & RELACE**

**pojdme si vytvořit databázi knih a mnohem lépe**

## JE TAKOVÉ HLEDÁNÍ POKAŽDÉ EFEKTIVNÍ ?

<https://aleph.muni.cz/>

- hledání podle vydavatele
- hledání podle jazyka
- hledání podle země

**ŘEŠENÍ? *TÉMĚŘ* KAŽDÝ DUPLICITNÍ ZÁZNAM MÁ  
VLASTNÍ TABULKU**

**NORMALIZACE DATABÁZE**

**[http://cs.wikipedia.org/wiki/Normalizace\\_databáze](http://cs.wikipedia.org/wiki/Normalizace_databáze)**



- vytvoříme tabulku publishers, countries, languages
- výhoda?

- v databázi z 2. přednášky můžeme využít pro language\_original i language\_traslation jednu tabulku languages

## TYPY RELACÍ

- 1:1 ...
- 1:N ...
- N:M ...

## TYPY RELACÍ

- 1:1 ... ne moc časté .. uživatel:heslo
- 1:N ... vydavatel:knihy / příspěvěk na FB:komentáře
- N:M ...autor:knihy

Databáze knih ve studijních materiálech - 7. přednáška  
books\_2

<http://tinyurl.com/qzhjpop>

Otázka: Počet knih VS primární klíč. Souhlasí ?

# INNER JOIN

Vypiš všechny knihy s vydavatelem.

Vypiš všechny knihy s jazykem i vydavatelem.

Vypiš vydatele a počet jím vydaných knih. Seřad' od nejčastějšího vydavatele.

Vypiš jazyk a počet knih v něm napsaných. Seřad' od nejčastějšího jazyka.

**Co když kniha nemá v databázi jazyk nebo vydavatele?**

Co když má kniha více jazyků?



# **INNER JOIN vs LEFT / RIGHT JOIN**

**NULL a NOT NULL při návrhu**

## NOT NULL - CONSTRAINTS

<https://www.sqlite.org/foreignkeys.html>

- při vytváření tabulek pojmenujeme vzájemné závislosti
  - mazání záznamů z tabulky pak musí mít logickou návaznost
  - první mažeme závislá data a až pak ta nadřazená:  
*(První smažeme knížky a až pak jazyky, protože knížky obsahují informaci o jazyce)*

Vypiš všechny knihy. Vypiš jejich vydavatele a jazyk v němž jsou napsány. (název knihy, jméno jazyka, jméno vydavatele)

*Porovnej variantu s INNER JOIN a LEFT JOIN*

**Rozdíl mezi těmito 3 dotazy:**

**Použij INNER JOIN**

Vypiš jazyk knihy a počet knih napsaných v tomto jazyce.

**Použij LEFT JOIN**

Vypiš jazyk knihy a počet knih napsaných v tomto jazyce.

**Použij LEFT JOIN a obrat' pořadí tabulek. (RIGHT JOIN)**