

# Databázové praktikum

ZS 2019/2020

# Organizácia kurzu

- Ján Mazák - [mazak@dcs.fmph.uniba.sk](mailto:mazak@dcs.fmph.uniba.sk)
  - Štvrtok, 12:20, M-217
- Michal Rjaško – [rjasko@dcs.fmph.uniba.sk](mailto:rjasko@dcs.fmph.uniba.sk)
  - Streda, 11:30, M-217
- **Hodnotenie:**
  - **3x domáca úloha za 30, 30 a 40 bodov**
    - 1. domáca úloha po 3. týždni
  - A: 90 a viac bodov
  - B: 80 až 89 bodov
  - C: 70 až 79 bodov
  - D: 60 až 69 bodov
  - E: 50 až 59 bodov

# Plán kurzu

- Datalog
- SQL, DDL, DML
- Práca s databázov v JAVE
- SQLite
- Explain - analýza / optimalizácia dotazov
- ...

# Datalog

- Podrobnejšie teoretické základy na prednáškach predmetu Úvod do databáz
- Program v datalogu je množina pravidiel – implikácii tvaru:
  - $\text{zlozene\_cislo}(Z) \leftarrow \text{krat}(X, Y, Z), \text{not } X = 1, \text{not } Y = 1.$
- Syntax:
  - `<atom>`
  - `<hlava> :- <telo>`
    - `<hlava>: <atom>`
    - `<telo>: <atom> | \+ <atom> | <telo>, <atom>`
- `\+` - negácia
- `:-` “implikácia”
- Riadky začínajúce znakom % sú komentáre

# Datalog

- Príklad datalogovského pravidla:
  - $\text{res}(N,J) \text{ :- emp}(\_,N,J,\_,\_,S,\_,\_), S \geq 2000.$
- Na ľavej strane len jeden pozitívny atóm
- Premenné začínajú veľkým písmenom
- Konštanty malými písmenami
- Každá premenná musí byť v tele pravidla uvedená aspoň v jednom „pozitívnom EDB kontexte“
- `_` znamená anonymnú premennú
- Na vyhodnocovanie aritmetických výrazov slúži operátor `is`:
  - napr.  $X \text{ is } 2+3,$
  - nie  $X = 2+3$
  - (v tom druhom prípade `symbol` = bude interpretovaný ako unifikácia termov a nedôjde k žiadnej aritmetickej operácii).

# Práca s datalogom

- Budeme robiť na serveri cvika.dcs.fmph.uniba.sk.
- Pripojiť sa cez ssh na počítač cvika (cvika.dcs.fmph.uniba.sk)
  - Prihlasovacie meno / heslo by malo byť ako v AISe
- Na počítači cvika, skopírovať súbory pre dané cvičenie, napr. z ~rjasko/db1 do svojho home-directory na počítači cvika:  
**cp -r ~rjasko1/db1 ~; cd ~/db1**
- Je rozumné otvoriť si cca 3 ssh okna na cvika.dcs.fmph.uniba.sk
  - V jednom okne editujete súbor, v ktorom píšete dotazy, napr.  
**vim queries\_emp.pl**
  - V druhom okne máte spustené prostredie prologu:  
**swipl -s queries\_emp.pl**

# Práca s datalogom

- Po vpísaní dotazu do súboru v OKNE1 treba súbor queries\_emp.pl uložiť na disk (v editore vim sa tak urobí postupným stlačením "ESC" a ":w").
- Následne v OKNE2 skompilujete novu verziu súboru príkazom **make**.
  - Je dobre pozrieť sa, či kompilátor hlási nejaké chyby a prípadne ich opraviť
- Potom sa v OKNE2 dajú písať dotazy ako napríklad **?- q(job(J)).**
  - Predikát "q(\_)" slúži na pekné formátovanie výstupu a elimináciu "duplikátov" (ktoré v skutočnosti nie sú duplikátmi, len tými istými viacnásobne nájdenými N-ticami)

# Databáza EMP

```
%emp(Empno, Ename, Job, Mgr, Hiredate, Sal, Comm, Deptno)
emp(7839, king, president, null, 19811117, 5000, null, 10).
emp(7698, blake, manager, 7839, 19810501, 2850, null, 30).
emp(7782, clark, manager, 7839, 19810609, 1500, null, 10).
emp(7566, jones, manager, 7839, 19810402, 2975, null, 20).
emp(7654, martin, salesman, 7698, 19810928, 1250, 1400, 30).
emp(7499, allen, salesman, 7698, 19810220, 1600, 300, 30).
emp(7844, turner, salesman, 7698, 19810908, 1500, 0, 30).
emp(7900, james, clerk, 7698, 19811203, 950, null, 30).
emp(7521, ward, salesman, 7698, 19810222, 1250, 500, 30).
emp(7902, ford, analyst, 7566, 19811203, 3000, null, 20).
emp(7369, smith, clerk, 7902, 19801217, 800, null, 20).
emp(7788, scott, analyst, 7566, 19821209, 3000, null, 20).
emp(7876, adams, clerk, 7788, 19830112, 1100, null, 20).
emp(7934, miller, clerk, 7782, 19820123, 1300, null, 10).
```

```
%salgrade(Grade, Losal, Hisal)
salgrade(1, 700, 1200).
salgrade(2, 1201, 1400).
salgrade(3, 1401, 2000).
salgrade(4, 2001, 3000).
salgrade(5, 3001, 9999).
```

```
%dept(Deptno, Dname, Loc)
dept(10, accounting, newyork).
dept(20, research, dallas).
dept(30, sales, chicago).
dept(40, operations, boston).
```



# Datalog a negácia

- Dvojice J, D, že pozícia J nie je v departmente D

jobDept(J, D) :- emp(\_,\_,J,\_,\_,\_,\_,D).

nemaJob(D) :- emp(\_,\_,J,\_,\_,\_,\_,\_),emp(\_,\_,\_,\_,\_,\_,\_,D),\+ jobDept(J,D)

Prečo nestačí napísať?

nemaJob(D,J) :- emp(\_,\_,J,\_,\_,\_,\_,\_),emp(\_,\_,\_,\_,\_,\_,\_,D),\+emp(\_,\_,J,\_,\_,\_,\_,D)

# Datalog a všeobecný kvatifikátor

- „Departments containing all job positions“
  - `allJobs(D) :- dept(D,_,_,_), \+ nemaJob(D)`