

# ŘEDITEL

## 1 Úloha ŘEDITEL – zadání

---

Jste ředitelem prestižního vzdělávacího ústavu, který ve svých mnoha třídách (asi čtyřech) různých ročníků (asi třech) vyučuje nepřeberné davy studentů (dohromady asi deset). Pro jejich závratný počet můžete vytvářet i několik tříd stejného ročníku a stanovujete pro každou třídu maximální počet studentů a jejich třídního učitele. Protože ve vašem městě není tak velká budova, která by pojala všechny vaše třídy, jsou třídy umístěny na různých místech v různých městských čtvrtích.

Studenti nejen studují, ale také vykonávají zkoušky z mnoha náročných předmětů (asi třech) s různým výsledkem. Zkoušející jsou odborníci ve zkoušených předmětech, ale vůbec to nemusí být třídní učitelé.

Vytvořte databázi, která vám odpoví na závažné otázky, jako např.:

1. Kolik ještě v každém ročníku zbývá volných míst?
2. Kolik ještě v každé třídě zbývá volných míst?
3. Který třídní nezkouší?
4. Ve které městské čtvrti mají studenti nejlepší průměr z matematiky?
5. Ve kterém ročníku je víc žen než mužů?

## 2 Úloha ŘEDITEL – analýza

---

### 2.1 Data ke sledování

#### 2.1.1 Data z popisu situace:

**Ročník; Třída; Student; MaxStudentů; Třídní; Místo; Předmět;  
Známka; Zkoušející**

#### 2.1.2 Další data z otázek:

**Pohlaví**

#### 2.1.3 Nutno tedy sledovat alespoň:

<b>Ročník</b>	<b>Třída</b>	<b>MaxStudentů</b>	<b>Třídní</b>	<b>Místo</b>	<b>Student</b>	<b>Pohlaví</b>	<b>Předmět</b>	<b>Zkoušející</b>	<b>Známka</b>
---------------	--------------	--------------------	---------------	--------------	----------------	----------------	----------------	-------------------	---------------

Označme tuto tabulku **T0**.

### 2.2 Dekompozice dat

#### 2.2.1 Datové typy

Zkusme cvičně vyplnit uvedenou datovou strukturu a přitom uvažujme o datovém typu každého pole. Je to jen na nás, popis situace a kladené otázky nás jen mírně vedou.

- **Ročník:** Zvolme to, co je obvyklé na školách: arabskými číslicemi, zde celé číslo od 1. Šlo by použít i římské číslice (tedy text), to by však mohlo být problémem při řazení.
- **Třída:** Opět se přidržme školských zvyklostí, tedy písmena A atd. Datový typ Text.
- **MaxStudentů:** Evidentně celé číslo.
- **Třídní:** Viz poznámka ke sledovaným údajům níže, např. Černý - tedy text.
- **Místo:** Pokud nepoužijeme pro daný účel trochu nepraktického označení místa např. pomocí PSC nebo GPS, zbývá slovní označení lokality, např. Vršovice - tedy text.
- **Student:** Pro jeho označení určitě nepoužijeme např. rodné číslo. Každý má ale příjmení, tedy text. Jen na nás je, jestli se spokojíme pro účely procvičování s malým objemem dat jen s příjmením, nebo do tohoto pole zahrneme i jméno, nebo zvolíme dvě textová pole - příjmení a jméno.
- **Pohlaví:** Možností je několik, dokonce většiny datových typů. Pomineme častý jednoznakový text M a Ž. Protože jsou jen dvě pohlaví a existuje datový typ mající jen dvě různě hodnoty (často označovaný jako Logický nebo Ano/Ne), zvolme pro indikaci pohlaví právě tento typ. Pak však bude logičtější, když se toto datové pole nebude jmenovat Pohlaví, ale buď Žena nebo Muž. Dáme přednost dámám a přidělíme tomuto datovému poli jméno ŽENA - a hodnota bude buď Ano nebo Ne.
- **Předmět:** Zvolme vypovídající hodnoty typu Mat, Fyz apod. Půjde o datový typ text.
- **Zkoušející:** Stejně jako u třídního - viz poznámka ke sledovaným údajům níže, např. Novák - tedy text.
- **Známka:** Ze zadání (má se pracovat s průměrnou známkou) vyplývá, že půjde o číslo. Jaký způsob známkování zvolíme, je na našem gustu: středoškolské celé číslo od 1 do 5, vysokoškolské od 1 do 4, bodové hodnocení třeba i necelým číslem od 10.0 do 0.0 apod.

Nyní tedy vyplnění nějakými cvičnými daty:

Ročník	Třída	Třídní	Místo	MaxStudentů	Student	Žena	Předmět	Zkoušející	Známka
2	A	Kos	Slovany	20	Aristoteles Sakrapulos Onasis	Ne	Fyz	Rác	4

## 2.2.2 Úvaha 1

Elementem umístění studentů je určitě nějaká třída jistého ročníku (určitě nebude student X studovat ve 2A a 2C současně). Kdyby nějaký student studoval souběžně v 1. ročníku a 3. ročníku (např. jiné specializace), administrativně půjde o jiného studenta – už z důvodu pobírání státní dotace MŠ podle počtu studentů. Z toho důvodu vyplývá struktura dat jedné třídy jednoho ročníku např. takto:

Tabulka **TŘÍDY**

Ročník	Třída	Třídní	Místo	MaxStudentů
2	A	Kos	Slovany	20

Primárním klíčem je evidentně spojený klíč [Ročník;Třída].

Tabulka T0 se tedy redukuje na tabulku **T1**:

Ročník	Třída	Student	Žena	Předmět	Zkoušející	Známka
2	A	Aristoteles Sakrapulos Onasis	Ne	Fyz	Rác	4

## 2.2.3 Úvaha 2

Každý student má své jméno (které podle národních zvyklostí může být značně složité), a v daném okamžiku dává k dispozici další údaje o své osobě – např. jde-li o ženu nebo muže (event. další, zde nesledované). Dále v daném okamžiku studuje v konkrétním ročníku konkrétní třídě. Logicky by tedy studenti měli mít svou tabulku, pro účely vazby na další tabulky opatřené nějakým jednoduchým jedinečným kódem (např. oblíbeným osobním číslem):

Tabulka **STUDENTI**

OsČís	Student	Žena	Ročník	Třída
ASO	Aristoteles Sakrapulos Onasis	Ne	2	A

Tabulka T1 se dále redukuje na tabulku **T2** při současné náhradě označení studenta (místo plným jménem) jeho osobním číslem:

OsČís	Předmět	Zkoušející	Známka
ASO	Fyz	Rác	4

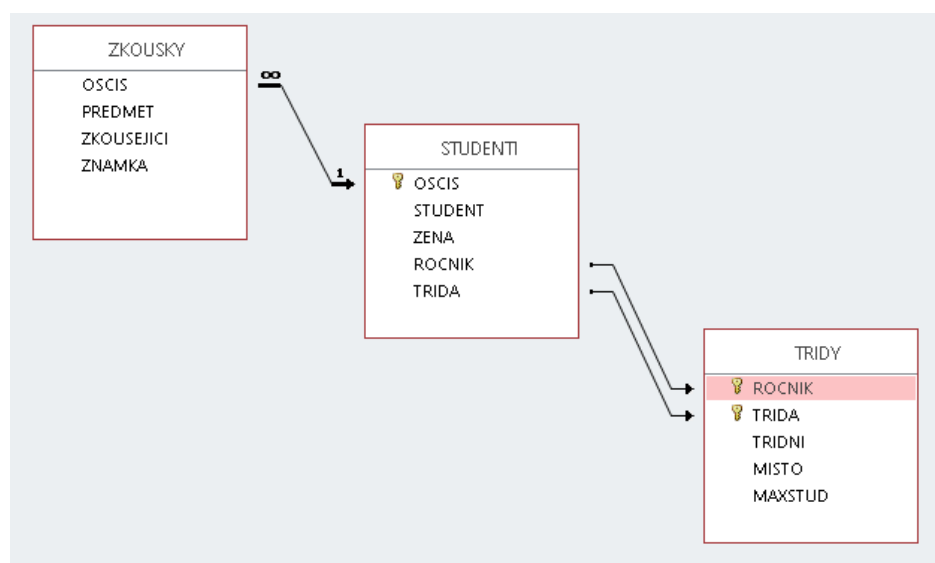
## 2.2.4 Úvaha 3

Každý řádek tabulky T2 je však nutnou a postačující informací o vykonání jedné zkoušky jednoho studenta z jednoho předmětu. Jde tedy o seznam (=tabulku) vykonaných zkoušek. Označme proto tabulku T2 např. Zkoušky:

Tabulka **ZKOUŠKY**

OsČís	Předmět	Zkoušející	Známka
ASO	Fyz	Rác	4

## 2.2.5 Vazby mezi tabulkami



Relace v úloze **ŘEDITEL**

## 2.2.6 Poznámky ke sledovaným údajům

Nikde (ani v popisu situace ani v kladených otázkách) není nic, co by nutilo sledovat u pedagogů něco jiného než jeho jednoznačnou identifikaci. Jestli např. angličtináře pošle ředitel zkoušet matiku, je to čistě věc mezi nimi dvěma. Protože celá tato úloha má za cíl hlavně procvičit práci v prostředí databázových programů a zejména sestavování příkazu Select kategorie SQL, netřeba vymýšlet další tabulku. Pro identifikaci kantorek a kantorů zde postačí např. jednoduché jedinečné příjmení. Ovšem individuálnímu mentálnímu rozletu se samozřejmě meze nekladou 😊. Analogicky např. městské čtvrti, předměty ...

Často mají studující databází tendenci jednoznačně identifikovat třídu v ročníku jediným údajem (= datovým polem). To samozřejmě lze, ale je to samoučelné a mnohdy matoucí. Zde by byl nejhorší možný způsob volit tzv. Automatické číslo. Na první pohled by nad hodnotou např. 2 mnozí kroutili hlavou, proč je třída 2 třídou 4. ročníku. Proto se zde procvičuje spojený indexový klíč. V kontextu relací nemusí sice všechny databázové systémy podporovat všechny jejich funkce (např. aktualizaci v kaskádě), pro tvorbu a vykonávání dotazů však žádné omezení není.

## 3 Úloha ŘEDITEL - Dotazy

---

### 3.1 Úloha 1

```
select
  S.ROCNIK,
  (
    select sum(T.MAXSTUD)
    from TRIDY T
    where S.ROCNIK=T.ROCNIK
  ) - Count(S.ROCNIK) as VOLNYCH_MIST
from STUDENTI S
group by S.ROCNIK
```

### 3.2 Úloha 2

```
select
  T.ROCNIK,
  T.TRIDA,
  T.MAXSTUD -
  (select count(S.ROCNIK)
   from STUDENTI S
   where
     T.ROCNIK=S.ROCNIK and T.TRIDA=S.TRIDA
  ) AS VOLNYCH_MIST
from TRIDY T
group by T.ROCNIK, T.TRIDA, T.MAXSTUD
```

## 3.3 Úloha 3

```
select TRIDNI as NEZKOUSI
from TRIDY
where TRIDNI not in (select ZKOUSEJICI from ZKOUSKY)
```

## 3.4 Úloha 4

### 3.4.1 S pomocným dotazem:

Pomocný dotaz **Uloha4a\_PrumeryMat**:

```
select
  T.MISTO, Avg(Z.ZNAMKA) AS PRUM_MAT
from
  (ZKOUSKY Z left join STUDENTI S on Z.OSCIS=S.OSCIS)
  left join TRIDY T on (S.ROCNIK = T.ROCNIK) and (S.TRIDA = T.TRIDA)
where Z.PREDMET="Mat"
group by T.MISTO
```

**Finální Uloha4:**

```
select MISTO
FROM Uloha4a_PrumeryMat
where PRUM_MAT=(select Min(PRUM_MAT) from Uloha4a_PrumeryMat)
```

### 3.4.2 Jediným dotazem:

**Poznámka:** Tohle vypadá hrozně, ale vytvoření je velmi jednoduché. Vyjde se z předchozího řešení s pomocným dotazem **Uloha4a\_PrumeryMat**, jehož text se zkopíruje do schránky. Výsledný dotaz vznikne z předchozího třířádkového finálního řešení nahrazením dvou výskytů totožných zdrojů obsahem schránky. To koneckonců přesně kopíruje logiku uvažování při analýze úlohy 4.

```
select MISTO
from
(
  select
    T.MISTO, Avg(Z.ZNAMKA) AS PRUM_MAT
  from
    (ZKOUSKY Z left join STUDENTI S on Z.OSCIS=S.OSCIS)
    left join TRIDY T
      on (S.TRIDA = T.TRIDA) and (S.ROCNIK = T.ROCNIK)
  where Z.PREDMET="Mat"
  group by T.MISTO
)
```

```

where PRUM_MAT=
(
  select Min(PRUM_MAT) from
  (
    select
      T.MISTO, Avg(Z.ZNAMKA) AS PRUM_MAT
    from
      (ZKOUSKY Z left join STUDENTI S on Z.OSCIS=S.OSCIS)
      left join TRIDY T
        on (S.TRIDA = T.TRIDA) and (S.ROCNIK = T.ROCNIK)
    where Z.PREDMET="Mat"
    group by T.MISTO
  )
)

```

## 3.5 Úloha 5

```

select
  ROCNIK,
  (select count(ROCNIK) as POC_ZEN
   from STUDENTI SZ
   where ZENA and S.ROCNIK=SZ.ROCNIK) AS ZEN,
  (select count(ROCNIK) as POC_MUZU
   from STUDENTI SM
   where not ZENA and S.ROCNIK=SM.ROCNIK) AS MUZU
from STUDENTI S
group by ROCNIK
having
  (select count(ROCNIK) as POC_ZEN
   from STUDENTI SZ
   where ZENA and S.ROCNIK=SZ.ROCNIK) >
  (select count(ROCNIK) as POC_MUZU
   from STUDENTI SM
   where not ZENA and S.ROCNIK=SM.ROCNIK)
... having ZEN > MUZU

```

**Poznámka:** Uvedený text příkazu Select by měl pracovat ve všech databázových systémech. Některé méně zdařilé systémy **nepovolují** naprosto logický tvar klauzule Having:

```
... having ZEN > MUZU
```

Rev.: 09/2024