

DATABÁZOVÉ SYSTÉMY I

Sbírka úloh ke cvičení

Petr Lukáš, Peter Chovanec, Radim Bača

20. října 2020

Obsah

1	Základy SQL, příkaz SELECT	9
2	Spojování tabulek	11
3	Agregační funkce a shlukování	13
4	Množinové operace a kvantifikátory	15
5	Poddotazy	17
6	Příkazy pro modifikaci a definici dat	19

Úvod

Tento studijní materiál slouží pro výuku dotazování v jazyce SQL. Materiál je rozdělen celkem do pěti okruhů, které odpovídají látce probírané na jednotlivých cvičeních předmětu Databázové systémy 1 (DS1). Každý okruh obsahuje přibližně 30 příkladů. První okruh se věnuje základnímu použití příkazu SELECT, druhý se zabývá především spojováním tabulek, třetí používáním agregačních funkcí, čtvrtý množinovými operacemi a poslední, pátý, poddotazy a složitějšími dotazy. Studentům jsou k dispozici dvě verze tohoto materiálu: verze bez řešení a verze s řešením včetně slovního vysvětlení. Na cvičení studenti pracují výhradně s verzí bez řešení. Verze s řešením je pak vhodná pro samostatnou přípravu na test.

Prosíme studenty, aby případné nedostatky (nejednoznačná zadání, chyby v řešení, nejasnosti v popisu řešení nebo jiné návrhy na zlepšení) reportovali na jednu z e-mailových adres: `petr.lukas@vsb.cz`, `peter.chovanec@vsb.cz` nebo `radim.baca@vsb.cz`. Přispějete tím ke zkvalitnění výuky pro další studenty. Zároveň děkujeme všem studentům, kteří již odhalili předchozí nedostatky.

*Jmenovité poděkování za odhalení chyb patří následujícím studentům:
Matěj Šimko, Alfons Václavík, Borek Tuleja*

Databáze Sakila

V následujících cvičeních předmětu DS1 budeme pracovat s databází fiktivní filmové půjčovny pojmenovanou jako Sakila. Tato databáze byla původně navržena pro demonstraci příkladů nad databázovým systémem MySQL¹. Postupem času však vznikly porty i pro jiné systémy² jako třeba Microsoft SQL Server, který my budeme používat. Skripty pro vytvoření struktury databáze a naplnění daty naleznete na stránkách předmětu na dbedu.cs.vsb.cz. Pro potřeby předmětu byl obsah databáze mírně modifikován, aby bylo možné lépe demonstrovat některé rysy jazyka SQL, tzn. aby dotazy např. nevracely prázdný výsledek nebo aby vracely zjevně nesprávný výsledek v případě použití nesprávné konstrukce.

Relační datový model

Strukturu relační databáze obvykle znázorňujeme tzv. E-R (Entity-Relationship) diagramem. E-R diagram databáze Sakila vidíme na Obrázku 1. Doporučujeme si obrázek vytisknout, jelikož do něj budeme, alespoň ze začátku, nahlížet velmi často.

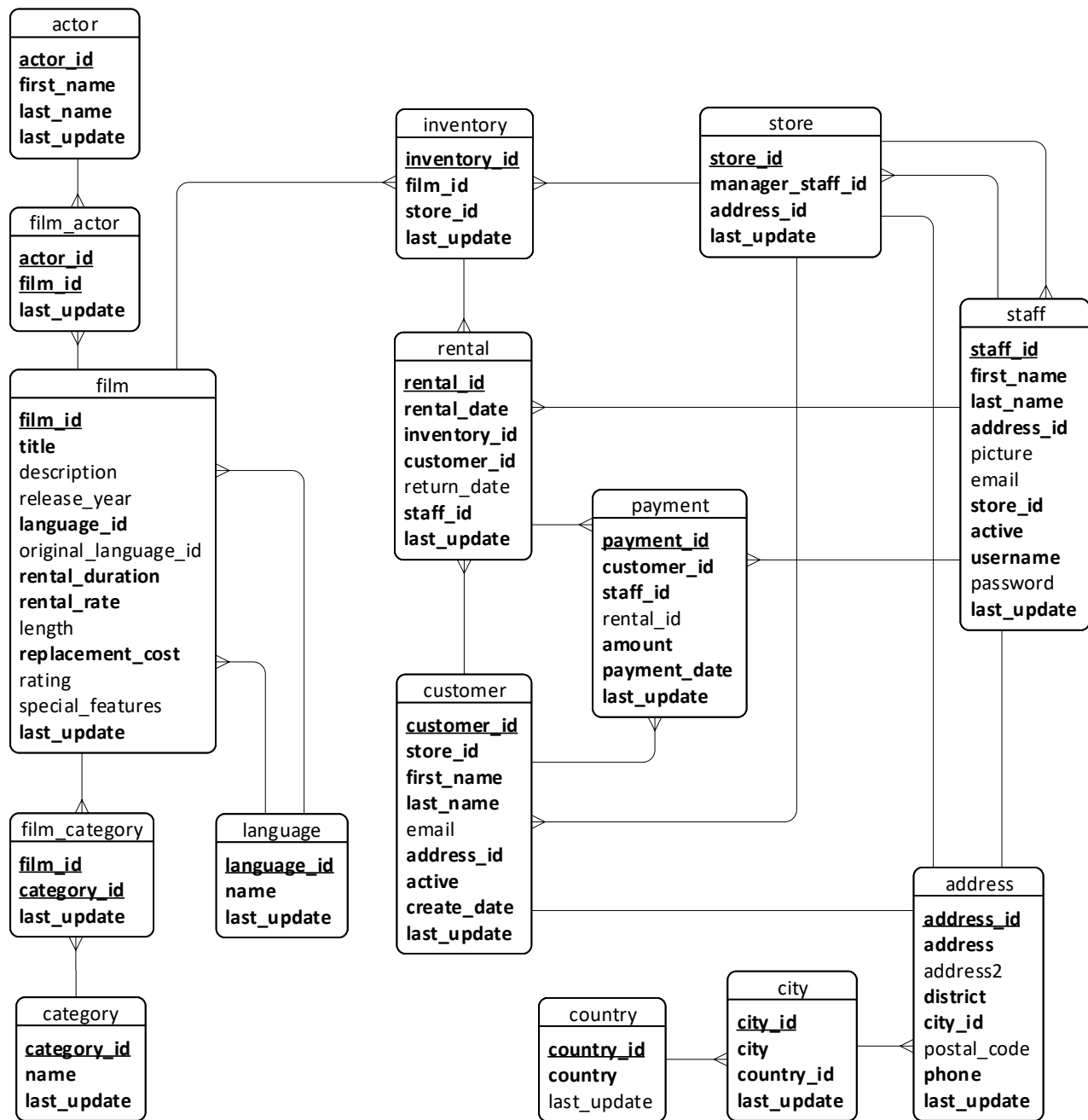
Na obrázku vlevo uprostřed se nachází tabulka `film`, tedy tabulka filmů, která je pomocí vazební tabulky `film_actor` propojena s tabulkou `actor` obsahující herce. Mezi filmy a herci je tedy vazba M:N, což znamená, že v jednom filmu může hrát více herců a jeden herec může hrát ve více filmech. Podobně je tomu s filmovými kategoriemi v tabulce `category`, které jsou s filmy propojeny vazební tabulkou `film_category`. Jeden film tedy může být zařazen do více kategorií (horror, komedie apod.) a naopak. Film je dále ve vazbě N:1 propojen s tabulkou `language` představující jazyky. Vazbu mezi filmy a jazyky vidíme hned dvakrát. První výskyt představuje vazbu na skutečný jazyk filmu, druhý vazbu na originální jazyk v případě, že byl film dabován.

Pokračujeme dále na tabulku `inventory` představující jednotlivé kopie filmů. Půjčovna totiž může jeden film vlastnit ve více kopiích. Mezi filmem a kopií je přirozeně vazba 1:N. Následuje tabulka `rental` představující jednotlivé výpůjčky. Výpůjčka se vždy týká nějaké kopie filmu, provedl ji nějaký zákazník v tabulce `customer` a vyřídil ji nějaký zaměstnanec v tabulce `staff`. Mezi tabulkou `rental` a tabulkami `inventory`, `customer` a `staff` jsou tedy vazby N:1. Dále si všimněme tabulky `payment` představující platby. Platbu vždy provedl nějaký zákazník v tabulce `customer` a zpracoval ji nějaký zaměstnanec v tabulce `staff`. Platba se pak může ale také nemusí vztahovat k výpůjčce. Platby, které se nevztahují k výpůjčce mohou představovat např. předplatné.

V databázi dále nalezneme tabulky `country`, `city` a `address`, které na sebe postupně navazují vazbami 1:N, tzn. v jednom státu se nachází více měst a v jednom městě nalezneme více adres. Adresa je pak vazbou 1:1 navázaná na zákazníka, sklad nebo zaměstnance. Každá adresa se váže výlučně na jeden záznam z těchto tří tabulek, tzn. buď na zákazníka, nebo na sklad, nebo na zaměstnance.

¹<https://dev.mysql.com/doc/sakila/en/>

²<https://github.com/jOOQ/jOOQ/tree/master/jOOQ-examples/Sakila>



primární klíč
 povinný atribut
 nepovinný atribut

Obrázek 1: E-R diagram databáze Sakila

Datový slovník

Přestože jsou názvy tabulek i atributů v databázi Sakila většinou samopopisné, uvádíme zde pro pořádek kompletní datový slovník. Datový slovník není nutné podrobně studovat, můžete však do něj nahlédnout, kdykoli si nebudete jisti významem některého z atributů.

NULL informace, zda je atribut nepovinný, tzn. zda může nabývat hodnoty NULL

PK informace, zda jde o primární klíč

FK informace, zda jde o cizí klíč

RENTAL

tabulka výpůjček

sloupec	datový typ	NULL	PK	FK	popis
rental_id	celé číslo	ne	ano	ne	automaticky generovaný PK
rental_date	datum a čas	ne	ne	ne	datum zahájení výpůjčky
inventory_id	celé číslo	ne	ne	ano	ID vypůjčené kopie filmu
customer_id	celé číslo	ne	ne	ano	ID zákazníka, který výpůjčku provedl
return_date	datum a čas	ano	ne	ne	čas vrácení, pokud není vyplněno, jde o běžící výpůjčku
staff_id	celé číslo	ne	ne	ano	ID zaměstnance
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

ACTOR

tabulka herců

sloupec	datový typ	NULL	PK	FK	popis
actor_id	celé číslo	ne	ano	ne	automaticky generovaný PK
first_name	řetězec, max. 45 znaků	ne	ne	ne	křestní jméno
last_name	řetězec, max. 45 znaků	ne	ne	ne	příjmení
last_update	datum a čas	ne	ne	ne	datum a čas poslední aktualizace

COUNTRY

tabulka států

sloupec	datový typ	NULL	PK	FK	popis
country_id	celé číslo	ne	ano	ne	automaticky generovaný PK
country	řetězec, max. 50 znaků	ne	ne	ne	název státu
last_update	datum a čas	ano	ne	ne	čas poslední aktualizace záznamu

CITY

tabulka měst

sloupec	datový typ	NULL	PK	FK	popis
city_id	celé číslo	ne	ano	ne	automaticky generovaný PK
city	řetězec, max. 50 znaků	ne	ne	ne	název města
country_id	celé číslo	ne	ne	ano	ID státu
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

ADDRESS

tabulka adres

sloupec	datový typ	NULL	PK	FK	popis
address_id	celé číslo	ne	ano	ne	automaticky generovaný PK
address	řetězec, max. 50 znaků	ne	ne	ne	první řádek adresy
address2	řetězec, max. 50 znaků	ano	ne	ne	nepovinný druhý řádek adresy
district	řetězec, max. 20 znaků	ne	ne	ne	region
city_id	celé číslo	ne	ne	ano	ID města
postal_code	řetězec, max. 10 znaků	ano	ne	ne	poštovní směrovací číslo
phone	řetězec, max. 20 znaků	ne	ne	ne	telefon
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

LANGUAGE

tabulka jazyků

soupec	datový typ	NULL	PK	FK	popis
language_id	celé číslo	ne	ano	ne	ID jazyka
name	řetězec, max. 20 znaků	ne	ne	ne	název jazyka
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

CATEGORY

tabulka filmových kategorií

soupec	datový typ	NULL	PK	FK	popis
category_id	celé číslo	ne	ano	ne	automaticky generovaný PK
name	řetězec, max. 25 znaků	ne	ne	ne	název kategorie
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

CUSTOMER

tabulka zákazníků

soupec	datový typ	NULL	PK	FK	popis
customer_id	celé číslo	ne	ano	ne	automaticky generovaný PK
store_id	celé číslo	ne	ne	ano	ID skladu
first_name	řetězec, max. 45 znaků	ne	ne	ne	křestní jméno
last_name	řetězec, max. 45 znaků	ne	ne	ne	příjmení
email	řetězec, max. 50 znaků	ano	ne	ne	e-mail
address_id	celé číslo	ne	ne	ano	ID adresy
active	řetězec, max. 1 znaků	ne	ne	ne	příznak, zda je zákazník aktivní
create_date	datum a čas	ne	ne	ne	čas vytvoření záznamu
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

FILM

tabulka filmů

soupec	datový typ	NULL	PK	FK	popis
film_id	celé číslo	ne	ano	ne	automaticky generovaný PK
title	řetězec, max. 255 znaků	ne	ne	ne	název filmu
description	text	ano	ne	ne	popis nebo stručný obsah filmu
release_year	řetězec, max. 4 znaků	ano	ne	ne	rok vydání
language_id	celé číslo	ne	ne	ano	ID jazyka
original_language_id	celé číslo	ano	ne	ano	ID originálního jazyka v případě dabovaného filmu
rental_duration	celé číslo	ne	ne	ne	standardní délka výpůjčky ve dnech
rental_rate	des. číslo	ne	ne	ne	částka za výpůjčku trvající standardní délkou
length	celé číslo	ano	ne	ne	délka filmu v minutách
replacement_cost	des. číslo	ne	ne	ne	částka za náhradu v případě ztráty nebo poškození média
rating	řetězec, max. 10 znaků	ano	ne	ne	klasifikace filmu MPAA
special_features	řetězec, max. 255 znaků	ano	ne	ne	speciální vlastnosti
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

FILM_ACTOR

vazební tabulka mezi herci a filmy

soupec	datový typ	NULL	PK	FK	popis
actor_id	celé číslo	ne	ano	ano	ID herce
film_id	celé číslo	ne	ano	ano	ID filmu
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

FILM_CATEGORY

vazební tabulka mezi kategoriemi a filmy

sloupec	datový typ	NULL	PK	FK	popis
film_id	celé číslo	ne	ano	ano	ID filmu
category_id	celé číslo	ne	ano	ano	ID kategorie
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace

INVENTORY

tabulka kopií filmů, jeden film může půjčovna vlastnit ve více kopiích

sloupec	datový typ	NULL	PK	FK	popis
inventory_id	celé číslo	ne	ano	ne	automaticky generovaný PK
film_id	celé číslo	ne	ne	ano	ID filmu
store_id	celé číslo	ne	ne	ano	ID skladu
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

STAFF

tabulka zaměstnanců půjčovny

sloupec	datový typ	NULL	PK	FK	popis
staff_id	celé číslo	ne	ano	ne	automaticky generovaný PK
first_name	řetězec, max. 45 znaků	ne	ne	ne	křestní jméno
last_name	řetězec, max. 45 znaků	ne	ne	ne	příjmení
address_id	celé číslo	ne	ne	ano	ID adresy
picture	image	ano	ne	ne	fotografie
email	řetězec, max. 50 znaků	ano	ne	ne	e-mail
store_id	celé číslo	ne	ne	ano	ID skladu
active	bit	ne	ne	ne	příznak, zda jde o aktivního zaměstnance
username	řetězec, max. 16 znaků	ne	ne	ne	uživatelské jméno
password	řetězec, max. 40 znaků	ano	ne	ne	heslo zašifrované pomocí SHA1
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

STORE

tabulka skladů

sloupec	datový typ	NULL	PK	FK	popis
store_id	celé číslo	ne	ano	ne	automaticky generovaný PK
manager_staff_id	celé číslo	ne	ne	ano	ID zaměstnance – správce skladu
address_id	celé číslo	ne	ne	ano	ID adresy
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace

PAYMENT

tabulka plateb, platba se nemusí vždy vztahovat jen k výpůjčce

sloupec	datový typ	NULL	PK	FK	popis
payment_id	celé číslo	ne	ano	ne	automaticky generovaný PK
customer_id	celé číslo	ne	ne	ano	ID zákazníka – plátce
staff_id	celé číslo	ne	ne	ano	ID zaměstnance, který platbu zpracoval
rental_id	celé číslo	ano	ne	ano	ID výpůjčky
amount	des. číslo	ne	ne	ne	částka
payment_date	datum a čas	ne	ne	ne	čas platby
last_update	datum a čas	ne	ne	ne	čas poslední aktualizace záznamu

1 Základy SQL, příkaz SELECT

Na tomto cvičení se zaměříme na základní syntaxi příkazu SELECT. Ve všech dotazech budeme pracovat vždy jen s jednou tabulkou, přičemž si vyzkoušíme jednoduchou selekci, projekci, skládání logických podmínek, základní funkce pro práci s datem a textem a tzv. agregační funkce.

1. Vypište e-mailové adresy všech neaktivních zákazníků.
2. Vypište názvy a popisy všech filmů s klasifikací (atribut `rating`) G. Výstup bude setříděn sestupně podle názvu filmu.
3. Vypište všechny údaje o platbách, které proběhly v roce 2006 nebo později a částka byla menší než 2.
4. Vypište popisy všech filmů klasifikovaných jako G nebo PG.
5. Vypište popisy všech filmů klasifikovaných jako G, PG nebo PG-13.
6. Vypište popisy všech filmů, které nejsou klasifikovány jako G, PG nebo PG-13.
7. Vypište všechny údaje filmů, jejichž délka přesahuje 50 minut a doba výpůjčky je 3 nebo 5 dní.
8. Vypište názvy filmů, které obsahují „RAINBOW“ nebo začínají na „TEXAS“ a jejich délka přesahuje 70 minut. Zamyslete se nad nejednoznačností formulace této úlohy v přirozeném jazyce.
9. Vypište názvy všech filmů, v jejichž popisu se vyskytuje „And“, jejich délka spadá do intervalu 80 až 90 minut a standardní doba výpůjčky (atribut `rental_duration`) je liché číslo.
10. Vypište vlastnosti (atribut `special_features`) všech filmů, kde částka za náhradu škody (atribut `replacement_cost`) je v intervalu 14 až 16. Zajistěte, aby se vlastnosti ve výstupu neopakovaly. Setříd'te vybrané vlastnosti abecedně. Zamyslete se, proč je výsledek i bez explicitního požadavku na setřídění již abecedně setříděný.
11. Vypište všechny údaje filmů, jejichž standardní doba výpůjčky je menší než 4 dny, nebo jsou klasifikovány jako PG. Nesmí však splňovat obě podmínky zároveň.
12. Vypište všechny údaje o adresách, které mají vyplněno PSČ.
13. Vypište ID všech zákazníků, kteří aktuálně mají vypůjčený nějaký film. Dokázali byste spočítat, kolik takových zákazníků je?
14. Pro každé ID platby vypište v samostatných sloupcích rok, měsíc a den, kdy platba proběhla. Sloupce vhodně pojmenujte.
15. Vypište filmy, jejichž délka názvu není 20 znaků.
16. Pro každou ukončenou výpůjčku (její ID) vypište dobu jejího trvání v minutách.

17. Pro každého aktivního zákazníka vypište jeho celé jméno v jednom sloupci. Výstup tedy bude obsahovat dva sloupce – `customer_id` a `full_name`.
18. Pro každou adresu (atribut `address`) vypište PSČ. Jestliže PSČ nebude vyplněno, bude se místo něj zobrazovat text „(prázdné)“.
19. Pro všechny uzavřené výpůjčky vypište v jednom sloupci interval od – do (tj. obě data oddělená pomlčkou), kdy výpůjčka probíhala.
20. Pro všechny výpůjčky vypište v jednom sloupci interval od – do (tj. obě data oddělená pomlčkou), kdy výpůjčka probíhala. Pokud výpůjčka dosud nebyla vrácena, vypište pouze datum od.
21. Vypišete počet všech filmů v databázi.
22. Vypišete počet různých klasifikací filmů (atribut `rating`).
23. Vypišete jedním dotazem počet adres, počet adres s vyplněným PSČ a počet různých PSČ.
24. Vypišete nejmenší, největší a průměrnou délku všech filmů. Ověřte si zjištěnou průměrnou délku pomocí podílu součtu a počtu.
25. Vypišete počet a součet všech plateb, které byly provedeny v roce 2005.
26. Vypišete celkový počet znaků v názvech všech filmů.

2 Spojování tabulek

Předchozí cvičení bylo zaměřeno na práci vždy s jednou tabulkou. Za klauzulí `FROM` se však obvykle nachází tabulek více. Na tomto cvičení si ukážeme, jak tabulky spojovat, přičemž se zaměříme na tzv. vnitřní spojení a levé vnější spojení. Při řešení nebudeme potřebovat žádné agregační funkce, poddotazy nebo konstrukce jako `IN` nebo `EXISTS`. Všechny úkoly bude možné řešit pouze vhodným spojením několika tabulek a případně omezením duplicitních výsledků pomocí klíčového slova `DISTINCT`.

1. Vypište všechny informace o městech včetně odpovídajících informací o státech, kde se tato města nachází.
2. Vypište názvy všech filmů včetně jejich jazyka.
3. Vypište ID všech výpůjček zákazníka s příjmením `SIMPSON`.
4. Vypište adresu (atribut `address` v tabulce `address`) zákazníka s příjmením `SIMPSON`. Porovnejte tento příklad s předchozím co do počtu řádků ve výsledku.
5. Pro každého zákazníka (jeho jméno a příjmení) vypište adresu bydliště včetně názvu města.
6. Pro každého zákazníka (jeho jméno a příjmení) vypište název města, kde bydlí.
7. Vypište ID všech výpůjček včetně jména zaměstnance, jména zákazníka a názvu filmu.
8. Pro každý film (jeho název) vypište jména a příjmení všech herců, kteří ve filmu hrají. Kolik řádků bude ve výsledku tohoto dotazu?
9. Pro každého herce (jeho jméno a příjmení) vypište jména všech filmů, kde herec hrál. Čím se bude tento dotaz lišit od předchozího? Co můžeme říct o operaci vnitřního spojení tabulek?
10. Vypište názvy všech filmů v kategorii „Horror“.
11. Pro každý sklad (jeho ID) vypište jméno a příjmení jeho správce. Dále vypište adresu skladu a adresu správce (u obou adres stačí atribut `address` v tabulce `address`). Řešení dále rozšiřte o výpis adresy včetně názvu města a státu.
12. Pro každý film (ID a název) vypište ID všech herců a ID všech kategorií, do kterých film spadá. Tzn. napište dotaz, jehož výsledkem bude tabulka s atributy `film_id`, `actor_id` a `category_id`, seřazeno dle `film_id`. Z výsledku pohledem zjistíte, kolik herců hraje ve filmu s `film_id = 1`, kolik tomuto filmu odpovídá kategorií a kolik je pro tento film celkem řádků ve výsledku.
13. Vypište všechny kombinace atributů ID herce a ID kategorie, kde daný herec hrál ve filmu v dané kategorii. Výsledek seřadíte dle ID herce. Dotaz dále rozšiřte o výpis jména a příjmení herce a názvu kategorie.
14. Vypište jména filmů, které půjčovna vlastní alespoň v jedné kopii.
15. Zjistěte jména herců, kteří hrají v nějaké komedii (kategorie „Comedy“).

16. Vypište jména všech zákazníků, kteří pochází z Itálie a někdy měli nebo mají půjčený film s názvem MOTIONS DETAILS.
17. Zjistěte jména a příjmení všech zákazníků, kteří mají aktuálně vypůjčený nějaký film, kde hraje herec SEAN GUINNESS.
18. Vypište ID a částku všech plateb a u každé platby uveďte datum výpůjčky, tj. hodnotu atributu `rental_date` v tabulce `rental`. U plateb, které se nevztahují k žádné výpůjčce bude datum výpůjčky `NULL`.
19. Pro každý jazyk vypište názvy všech filmů v daném jazyce. Zajistěte, aby byl jazyk ve výsledku obsažen, i když k němu nebude existovat odpovídající film.
20. Pro každý film (ID a název) vypište jeho jazyk a jeho původní jazyk.
21. Vypište názvy filmů, které si někdy půjčil zákazník TIM CARY, nebo je jejich délka 48 minut.
22. Vypište názvy filmů, které půjčovna nevlastní ani v jedné kopii (tj. nejsou obsaženy v inventáři).
23. Vypište jména a příjmení všech zákazníků, kteří mají nějakou nezaplacenou výpůjčku.
24. U každého názvu filmu vypište jazyk filmu, pokud jazyk začíná písmenem „I“, v opačném případě bude jazyk `NULL`.
25. Pro každého zákazníka vypište ID všech plateb s částkou větší než 9. U zákazníků, kteří takovéto platby nemají, bude `payment_id` rovno `NULL`.
26. Pro každou výpůjčku (její ID) vypište název filmu, pokud obsahuje písmeno „U“, a město a stát zákazníka, jehož adresa obsahuje písmeno „A“. Podobně jako v předchozích úlohách – jestliže údaj nesplňuje danou podmínku, bude v příslušném poli uvedeno `NULL`.
27. Vypište všechny dvojice název filmu a příjmení zákazníka, kde si zákazník vypůjčil daný film. Pokud výpůjčka proběhla po datu 1.1.2006, bude příjmení zákazníka nevyplněné (tj. `NULL`). Z výsledku odstraňte duplicitní řádky a seřadíte je podle názvu filmu.

3 Agregáčn  funkce a shlukov n 

S agrega n mi funkcemi jsme se j   setkali na prvn m cvi en , kde jsme je v  ak pou  ivali tak,  e v  sledkem byl v  dy jeden souhrnn   r  dek obsahuj  c  jednu nebo v  ce vypo  ten  ch hodnot. Na tomto cvi en  si ale uk   eme,  e agregace lze po   tat i samostatn   pro ur  it  skupiny z  znam  . V  sledkem agregac  tedy nemus   b  t jedin   r  dek, ale v  ce r  dk  , kter   seskupuj   z  znamy dle ur  it  ch podm  nek. Za  neme jednodu    mi p   klady nad jednou tabulku, pot   v  ak nav    eme na znalosti z p  edchoz  ho cvi en  a agregace budeme aplikovat na v  ce spojen  ch tabulek.

1. Vyp   te po  ty film   pro jednotliv   klasifikace (atribut `rating`).
2. Pro ka  d   ID z  kazn  ka vyp   te po  et jeho p   jmen  . Je ve v  sledku n  co p  ekvapiv  ho?
3. Vyp   te ID z  kazn  k   set   zen   podle sou  tu v  ech jejich plateb. Z  kazn  ky, kter   neprovedli    dnou platbu neuva  ujte.
4. Pro ka  d   jm  no a p   jmen   herce vyp   te po  et herc   s takov  m jm  nem a p   jmen  m. V  sledek set    te dle po  tu sestupn  .
5. Vyp   te sou  ty v  ech plateb za jednotliv   roky a m  s  ce. V  sledek uspo    ejte podle rok   a m  s  c  .
6. Vyp   te ID sklad   s v  ce n   2 300 kopiemi film  .
7. Vyp   te ID jazyk  , pro kter   je nejkrat     film del     n   46 minut.
8. Vyp   te roky a m  s  ce plateb, kdy byl sou  et plateb v  t     n   20 000.
9. Vyp   te klasifikace film   (atribut `rating`), jejich   delka je men     n   50 minut a celkov   delka v dan   klasifikaci je v  t     n   250 minut. V  sledek set    te sestupn   podle abecedy.
10. Vyp   te pro jednotliv   ID jazyk   po  et film  . Vyne  ejte jazyky, kter   nemaj      dn   film.
11. Vyp   te n  zvy jazyk   a k nim po  ty film  . Vyne  ejte jazyky, kter   nemaj      dn   film.
12. Vyp   te n  zvy v  ech jazyk   a k nim po  ty film  . Ve v  sledku budou zahrnuty i ty jazyky, kter   nemaj      dn   film.
13. Vyp   te pro jednotliv   z  kazn  ky (jejich ID, jm  no a p   jmen  ) po  ty jejich v  p    ek  .
14. Vyp   te pro jednotliv   z  kazn  ky (jejich ID, jm  no a p   jmen  ) po  ty r  zn  ch film  , kter   si v  p    ili.
15. Vyp   te jm  na a p   jmen   herc  , kter   hraj   ve v  ce n   20-ti filmech.
16. Pro ka  d  ho z  kazn  ka vyp   te, kolik celkem utratil za v  p    ky film   a jak   byla jeho nejmen    , nejv  t     a p   m  rn       tka platby.
17. Vyp   te pro ka  dou kategorii p   m  rnou delku filmu.
18. Pro ka  d   film vyp   te, jak   byl celkov   p   jem z v  p    ek  . Vyp   te jen filmy, kde byl celkov   p   jem v  t     n   100.

19. Pro každého herce vypište, v kolika různých kategoriích filmů hraje.
20. Vypište adresy zákazníků (atribut `address.address`) včetně názvu města a státu, kde ve filmech, které si zákazníci půjčili, hrálo dohromady alespoň 40 různých herců.
21. Pro všechny filmy (ID a název) spadající do kategorie „Horror“ uveďte, v kolika různých městech bydlí zákazníci, kteří si daný film někdy půjčili.
22. Pro všechny zákazníky z Polska vypište, do kolika různých kategorií spadají filmy, které si tito zákazníci vypůjčili.
23. Vypište názvy všech jazyků k nim počty filmů v daném jazyce, které jsou delší než 350 minut.
24. Vypište, kolik jednotliví zákazníci utratili za výpůjčky, které začaly v měsíci červnu.
25. Vypište seznam kategorií setříděný podle počtu filmů, jejichž jazyk začíná písmenem „E“.
26. Vypište názvy filmů s délkou menší než 50 minut, které si zákazníci s příjmením BELL půjčili přesně 1x.

4 Množinové operace a kvantifikátory

Až do teď jsme mohli každou úlohu vyřešit bez používání tzv. poddotazů. To znamená, že se klauzule `SELECT` v dotazu objevila vždy přesně jednou. Na tomto cvičení se zaměříme na používání konstrukcí `IN`, `EXISTS`, `ANY` a `ALL`, které používání poddotazů naopak vyžadují. Přestože některé z následujících úloh lze řešit i pomocí agregačních funkcí, pokusme se procvičit si především výše zmíněné konstrukce. Pokud nebude v zadání uvedeno jinak, je možné všechny následující úlohy vyřešit bez použití agregačních funkcí a shlukování. V praxi (nebo na testu) je pak ale jen na Vás, jaký způsob pro vyřešení úlohy použijete.

1. Vypište ID a názvy filmů, ve kterých hrál herec s ID = 1. Dotaz vyřešte bez použití `JOIN`.
2. Vypište ID filmů, ve kterých hrál herec s ID = 1.
3. Vypište ID a názvy filmů, ve kterých hrál herec s ID = 1 zároveň s hercem s ID = 10.
4. Vypište ID a názvy filmů, ve kterých hrál herec s ID = 1 nebo herec s ID = 10.
5. Vypište ID filmů, ve kterých nehrál herec s ID = 1.
6. Vypište ID a názvy filmů, ve kterých hrál herec s ID = 1 nebo herec s ID = 10, ale ne oba dohromady.
7. Vypište ID a názvy filmů, ve kterých hrál herec PENELOPE GUINNESS zároveň s hercem CHRISTIAN GABLE.
8. Vypište ID a názvy filmů, ve kterých nehraje herec PENELOPE GUINNESS.
9. Vypište jména zákazníků, kteří si půjčili všechny filmy `ENEMY ODDS`, `POLLOCK DELIVERANCE` a `FALCON VOLUME`.
10. Vypište jména a příjmení zákazníků, kteří si půjčili film `GRIT CLOCKWORK` v květnu i v červnu (libovolného roku).
11. Vypište jména a příjmení zákazníků, kteří se příjmením jmenují stejně jako nějaký herec.
12. Vypište jména filmů, které jsou stejně dlouhé, jako nějaké jiné filmy.
13. Vypište názvy filmů, které jsou kratší než nějaký film, ve kterém hraje BURT POSEY.
14. Vypište jména herců, kteří hráli v nějakém filmu kratším než 50 minut.
15. Nalezněte filmy, které byly půjčeny alespoň dvakrát.
16. Nalezněte filmy, které si půjčili alespoň 2 různí zákazníci.
17. Vypište zákazníky, kteří měli v určitou chvíli ve výpůjčce zároveň více různých filmů.
18. Vypište jména a příjmení zákazníků, kteří si půjčili film `GRIT CLOCKWORK` v květnu i červnu téhož roku.
19. Vypište názvy filmů, které jsou kratší než všechny filmy, ve kterých hraje BURT POSEY.

20. Vypište jména herců, kteří hráli jen ve filmech kratších než 180.
21. Vypište zákazníky, kteří v žádném měsíci neměli více než 3 výpůjčky. Pro zjištění počtu výpůjček v jednotlivých měsících použijte agregační funkci a shlukování.
22. Vypište zákazníky, kteří si půjčovali filmy pouze v letních měsících (tj. červen až srpen včetně).
23. Vypište zákazníky, kteří vždy vrátili film do 8-mi dnů. Výpůjčky, které zákazník dosud nevrátil, ignorujte.
24. Vypište zákazníky, jejichž všechny výpůjčky byly delší než 1 den a půjčili si film, kde hraje DEBBIE AKROYD.
25. Vypište jména a příjmení zákazníků, kteří provedli přesně jednu výpůjčku.
26. Vypište názvy filmů, kde hraje jediný herec.
27. Vypište zákazníky, kteří si půjčovali vždy jeden a ten samý film.
28. Vypište názvy filmů, které si někdy půjčili zákazníci, kteří si nikdy nepůjčili jiný film.
29. Vypište všechny zákazníky (jména a příjmení) a jazyky, pokud si zákazník půjčoval pouze filmy v daném jazyce.
30. Vypište názvy filmů, které si vždy půjčovali jen zákazníci, kteří si nikdy jiný film nepůjčili.
31. Vypište jména a příjmení zákazníků, kteří si vždy půjčovali pouze filmy, kde hrál herec CHRISTIAN GABLE.
32. Vypište herce, kteří hráli vždy jen ve filmu, který půjčovna vlastní alespoň ve třech kopiích. Pro zjištění počtu kopií v inventáři použijte agregační funkci.
33. Nalezte filmy, jejichž všechny kopie byly půjčeny alespoň 4x. Pro zjištění počtu kopií v inventáři použijte agregační funkci.
34. Vypište herce, jejichž všechny filmy, kde hráli, jsou delší než filmy, ve kterých hrál herec CHRISTIAN GABLE.
35. Vypište herce, jejichž filmy delší než 180 si půjčovali vždy zákazníci ze stejné země.

5 Poddotazy

Poslední cvičení na dotazování v jazyce SQL je zaměřeno na používání poddotazů a to nejen v souvislosti s konstrukcemi, které jsme si ukázali na cvičení předchozím. Poddotazy nám nakonec umožní řešit poměrně komplikované úkoly.

1. Pro každý film vypište, kolik v něm hraje herců a v kolika se nachází kategoriích.
2. Pro každého zákazníka vypište počet výpůjček trvajících méně než 5 dní a počet výpůjček trvajících méně než 7 dní.
3. Pro každý sklad (jeho ID) vypište počet kopií filmů (tj. položek v inventáři) pro filmy v anglickém jazyce a pro filmy ve francouzském jazyce.
4. Pro každý film vypište následující údaje:
 - (a) počet herců hrajících ve filmu,
 - (b) počet různých zákazníků, kteří si film půjčili v srpnu,
 - (c) průměrnou částku platby za výpůjčku filmu.
5. Vypište zákazníky, kteří v měsíci červnu provedli více než 5 plateb a nejdelší film, který si půjčili, má alespoň 185 minut.
6. Vypište zákazníky, jejichž většina plateb je o částce větší než 4.
7. Vypište herce, kteří hrají více než 2x častěji v komediích než v hororech.
8. Vypište herce, kteří hrají nejčastěji ve filmech delších než 150 minut, tzn. hrají častěji ve filmech delších než 150 minut než v ostatních filmech.
9. Vypište zákazníky, jejichž celkové uhrazené platby jsou menší než kolik by měli zaplatit dle atributů `film.rental_duration`, `film.rental_rate` a rozdílu `rental_date` a `return_date`. Při řešení můžete ignorovat nevrácené výpůjčky.
10. Vypište zákazníky, kteří si častěji půjčovali film, kde hraje TOM MCKELLEN, než film, kde hraje GROUCHO SINATRA.
11. Vypište zákazníky, kteří si půjčovali pouze filmy v anglickém jazyce, a u každého napište, kolik mají výpůjček.
12. Vypište všechny zákazníky, kteří si půjčili film, ve kterém hraje alespoň 15 herců, a u každého z nich vypište celkovou sumu z plateb, které provedli.
13. Vypište název nejdelšího filmu.
14. Vypište název nejdelšího filmu pro každou klasifikaci (atribut `film.rating`).
15. Pro každého zákazníka nalezněte film, který si půjčil naposledy. Seříd'te výsledek abecedně dle příjmení a jmen zákazníků.
16. Pro každého herce (jméno a příjmení) vypište název nejdelšího filmu, kde hrál.

17. Pro každý film vypište zákazníka, který měl tento film půjčený nejdelší dobu (v rámci jedné výpůjčky). Délku výpůjčky počítejte v sekundách pomocí funkce `DATEDIFF(second, rental_date, return_date)`.
18. Pro každého zákazníka vypište poslední vypůjčený film, ve kterém hraje herec PENELOPE GUINNESS. Pokud si zákazník film s hercem PENELOPE GUINNESS nikdy nepůjčil, nebude zákazník vypsán. Výsledek seřadíte dle ID zákazníků.
19. Vypište zákazníky, kteří si půjčili nejkratší i nejdelší film.
20. Vypište herce, kteří hráli alespoň 2x v nejdelším filmu.
21. Vypište filmy, které si alespoň dva zákazníci půjčili naposledy.
22. Pro každého herce vypište průměrný počet výpůjček za filmy, ve kterých hraje.
23. Pro každou klasifikaci filmu (atribut `film.rating`) vypište největší počet herců hrajících ve filmu v dané klasifikaci.
24. Vypište nejčastěji obsazované herce, tj. takové herce, kteří hrají v největším počtu filmů. Počet filmů, ve kterých herec hraje, bude součástí výpisu.
25. Vypište zákazníky s největším počtem výpůjček.
26. Vypište názvy filmů, které byly vypůjčeny nejvícekrát. Počet výpůjček bude součástí výpisu.
27. Vypište zákazníky, kteří provedli největší počet plateb. Největší počet plateb vypište také.
28. Vypište názvy filmů s nadprůměrným počtem výpůjček. Tj., spočítejte průměrný počet výpůjček na filmy a vypište filmy, kde počet výpůjček je nad tímto průměrem.
29. Vypište herce, kteří hrají nejčastěji ve filmech delších než 150, tzn. v rámci filmů s délkou přes 150 jde o nejčastěji obsazované herce.
30. Vypište zákazníky s největším rozdílem minimální a maximální platby za výpůjčku filmu započatou v červnu. Rozdíl bude součástí výpisu.
31. Vypište filmy, které byly vypůjčeny jedním zákazníkem nejvícekrát.
32. Vypište zákazníky, kteří si nejvícekrát půjčili stejný film.
33. Pro každé město vypište zákazníka s největším počtem výpůjček.
34. U každého zákazníka vypište, kolikrát si půjčil nějaký film nejčastěji a který film to byl. Ignorujte zákazníky bez výpůjčky.
35. Pro každou kategorii vypište filmy s nejmenším počtem výpůjček.
36. Pro každou kategorii filmu vypište nejčastěji obsazované herce hrající ve filmu v dané kategorii.
37. Pro každého zákazníka vypište jeho nejoblíbenějšího herce, tj. herce, který hrál v nejvíce různých filmech, které si zákazník půjčil. Ignorujte zákazníky bez výpůjčky.

6 Příkazy pro modifikaci a definici dat

Až do této chvíle jsme v naší databázi neprováděli žádné úpravy. Používali jsme pouze jediný příkaz SELECT, jehož možnosti jsou sice obrovské, ale samotná data a struktura databáze zůstávají nedotčeny. Dnes si naopak ukážeme příkazy spadající do kategorie DML (Data Manipulation Language) pro úpravu obsahu tabulek a příkazy spadající do kategorie DDL (Data Definition Language) pro úpravu struktury tabulek.

Toto cvičení se bude svou strukturou mírně odlišovat od těch předchozích z několika důvodů. Některé úlohy se budou skládat z více bodů, které je nutné řešit v přesně daném pořadí. Dále, pokud nebude přímo uvedeno, můžete úlohy řešit i více SQL příkazy, které budete spouštět postupně.

Než se pustíme do řešení úloh, poznamenejme, že zatímco syntaxe příkazu SELECT bývá napříč různými SRBD téměř stejná (je dodržován standard ANSI SQL), u příkazů v kategoriích DML a DDL se často objevují mírné rozdíly. V této sbírce budeme ukazovat syntaxi pro Microsoft SQL Server. Principiálně se ale řešení úloh pro jiné relační SRBD nebude lišit.

- (a) Vložte do databáze nového herce se jménem Arnold Schwarzenegger. Pro časové razítko poslední aktualizace záznamu (`last_update`) ponechte výchozí hodnotu (tzn. tuto hodnotu nenastavujte).
 - (b) Vložte do databáze film Terminátor. Popis a délku filmu zjistěte např. v databázi CSFD³. Jazyk filmu nastavte na Angličtinu, standardní dobu výpůjčky na 3 dny a cenu na 1,99. Ostatní atributy budou nevyplněné nebo budou nastaveny na výchozí hodnotu.
 - (c) Upravte obsah databáze tak, aby herec Arnold Schwarzenegger hrál ve filmu Terminátor. ID herce a filmu si předem zjistěte vhodnými dotazy.
 - (d) Zařaďte film Terminátor do kategorií „Action“ a „Sci-Fi“. ID příslušných záznamů zjistěte předem vhodnými dotazy.
 - (e) Zařaďte film Terminátor do kategorie „Comedy“. Úlohu vyřešte jedním příkazem za použití poddotazů tak, abyste se vyhnuli ručnímu zápisu konstant pro ID filmu a kategorie. Potřebná ID dohledejte podle názvů (`film.title` a `category.name`).
 - (f) Upravte cenu výpůjčky filmu Terminátor na hodnotu 2,99. Upravte současně také atribut `last_update` podle aktuálního časového razítka.
- (a) Vytvořte zaměstnance s Vaším jménem a Vaší adresou (údaje o adrese samozřejmě mohou být smyšlené). Uživatelské jméno bude Váš login a budete zařazeni do skladu s ID = 2. Potřebné konstanty pro cizí klíče zjistěte předem vhodnými dotazy.
 - (b) Vytvořte v databázi adresu naší univerzity.
 - (c) Vytvořte nový sklad na adrese naší univerzity. V novém skladu budete Vy manažerem.
 - (d) Pro každý film, který půjčovna vlastní alespoň v jedné kopii, přesuňte do nového skladu (viz předchozí bod) jeho kopii s nejvyšším ID.
- Navyšte cenu výpůjčky všech filmů, ve kterých hraje herec ZERO CAGE, o 10 %. Úlohu vyřešte jedním příkazem bez zápisu konstanty pro ID herce (herec bude identifikován svým jménem).

³<https://www.csfd.cz/film/1249-terminator/video/#play-video-157711308>

4. Všem filmům, jejichž původní jazyk (`original_language`) je Mandarínština (Mandarin), nastavte původní jazyk na NULL. Vyhněte se zápisu ID pro daný jazyk.
5. Pro každý film, ve kterém hraje herec GROUCHO SINATRA, vložte do tabulky `inventory` jednu novou kopii. Všechny tyto nové kopie budou umístěny ve skladu s ID = 2. Datum poslední aktualizace záznamu ponechte na výchozí hodnotě. Úlohu opět vyřešte jedním příkazem bez zápisu konstanty pro ID herce (herec bude identifikován svým jménem).
6. Odstraňte z databáze jazyk Mandarínština (Mandarin). Úlohu řešte až po vyřešení příkladu 4.
7. Odstraňte z databáze film Terminátor (úlohu řešte až po vyřešení příkladu 1). Je možné tuto úlohu řešit pouze odebráním příslušného záznamu z tabulky `film`?
8. Odstraňte z databáze všechny neaktivní zákazníci.
9. Přidejte do tabulky `film` nepovinný celočíselný atribut `inventory_count`. Nastavte tento atribut pro všechny filmy tak, aby odpovídal počtu kopií daného filmu (počtu odpovídajících záznamů v tabulce `inventory`).
10. Upravte atribut `name` v tabulce `category` tak, aby bylo možné zadat až 50 znaků.
11. Přidejte do tabulky `customer` povinný textový atribut `phone` o maximální délce 20 znaků. Telefon nastavte podle atributu `phone`, který je součástí adresy zákazníka.
12. Přidejte do tabulky `rental` povinný atribut `create_date`, jehož výchozí hodnota bude aktuální časové razítko.
13. Atribut `rental.create_date` vytvořený v předchozí úloze smažte.
14. Přidejte do tabulky `film` nepovinný atribut `creator_staff_id`, který bude cizím klíčem do tabulky `staff`. Cizí klíč pojmenujte jako `fk_film_staff`.
15. Nastavte kontrolu atributu `staff.email` tak, aby hodnota e-mailu vždy obsahovala znak „@“ následovaný znakem „.“.
16. Kontrolu nastavenou v předchozí úloze zrušte.
17. Nastavte kontrolu výpůjček tak, aby datum vrácení byl vždy větší než je datum výpůjčky.
18. Vytvořte novou tabulku `reservation`, tj. tabulku rezervací, s celočíselným automaticky generovaným primárním klíčem `reservation_id`. Tabulka bude dále obsahovat následující atributy: povinné datum rezervace `reservation_date` s výchozí hodnotou na aktuální datum, povinné datum konce rezervace `end_date`, povinné ID zákazníka `customer_id` jako cizí klíč do tabulky `customer`, povinné ID filmu `film_id` jako cizí klíč do tabulky `film` a nepovinné ID zaměstnance `staff_id` jako cizí klíč do tabulky `staff`.
19. Vložte do tabulky vytvořené v předchozí úloze dva libovolné záznamy. Poté druhý ze záznamů smažte. Jaké ID bude přiděleno dalšímu vkládanému záznamu?

20. (a) Vytvořte tabulku `review` s atributy `film_id` a `customer_id`, které budou představovat cizí klíče do tabulek `film`, resp. `customer`. Oba tyto atributy budou dohromady tvořit složený primární klíč. Součástí tabulky bude dále povinný atribut `stars`, který bude nabývat celých čísel v intervalu $\langle 1, 5 \rangle$, a nepovinný atribut `actor_id`, který bude cizím klíčem do tabulky `actor`. Zajistěte, aby se při smazání zákazníka nebo filmu automaticky smazaly také odpovídající záznamy v tabulce `review`. Dále zajistěte, aby při smazání herce došlo u odpovídajících záznamů v `review` k nastavení `actor_id` na `NULL`.
- (b) Vložte do tabulky `review` dva záznamy:
- Hodnocení filmu `ARMY FLINTSTONES` zákazníka `BRIAN WYMAN` – 4 hvězdy, bez uvedení herce.
 - Hodnocení filmu `ARSENIC INDEPENDENCE` zákazníka `CHERYL MURPHY` – 5 hvězd s uvedením herce `EMILY DEE`.
- (c) Odstraňte z databáze zákazníka `BRIAN WYMAN` a herce `EMILY DEE`. Následně si prohlédněte obsah tabulky `review`.
21. Zálohujte obsah tabulky `film` do nové tabulky `film_backup`. Nová tabulka bude svou strukturou totožná s tabulkou `film` s tím rozdílem, že nebude obsahovat nastavení primárních ani cizích klíčů. Jinými slovy, atributy jako `film_id`, `language_id` budou běžné celočíselné (neklíčové) atributy.
22. Odstraňte tabulky `review` a `film_backup` vytvořené v předchozích dvou úlohách.
23. Vytvořte tabulku `rating` s atributy `rating_id` – celočíselný automaticky generovaný primární klíč, `name` – povinný řetězec o maximálně 10-ti znacích a `description` – nepovinný řetězec s neomezeným počtem znaků. Vložte do nové tabulky záznamy pro unikátní hodnoty atributu `rating` v tabulce `film`. Vytvořte v tabulce `film` povinný atribut `rating_id`, který bude cizím klíčem do nově vytvořené tabulky `rating`. Hodnoty tohoto atributu budou nastaveny tak, aby odpovídaly skutečnosti dle atributu `rating`. Původní atribut `rating` nakonec smažte.
24. Odstraňte z databáze všechny tabulky.