# Vysoké učení technické v Brně

Fakulta informačních technologií

# Počítačové komunikace a sítě

2019/2020

Dokumentace projektu

**Varianta Zeta – Sniffer paketů**

## 

## Daniel Pátek (xpatek08) Brno, 1. května 2020

**Obsah**

[1. Úvod 3](#_Toc38987551)

[2. Zadání: 3](#_Toc38987552)

[3. Datový model (ERD) 4](#_Toc38987553)

[4. Model případů užití 4](#_Toc38987554)

[5. Vytvoření základních objektů schématu databáze 5](#_Toc38987555)

[6. Implementace několika dotazů SELECT 6](#_Toc38987556)

[1) Dva dotazy využívající spojení dvou tabulek 6](#_Toc38987557)

[2) Jeden dotaz využívající spojení tří tabulek 7](#_Toc38987558)

[3) Dva dotazy s klauzulí GROUP BY a agregační funkcí 7](#_Toc38987559)

[4) Jeden dotaz obsahující predikát EXISTS 8](#_Toc38987560)

[5) Jeden dotaz s predikátem IN s vnořeným selectem 8](#_Toc38987561)

[7. Vytvoření netriviálních databázových triggerů 9](#_Toc38987562)

[8. Vytvoření netriviálních uložených procedur 10](#_Toc38987563)

[9. Použití EXPLAIN PLAN a INDEX 10](#_Toc38987564)

[10. Přístupová práva 11](#_Toc38987565)

[11. Materializovaný pohled 11](#_Toc38987566)

[12. Závěr 11](#_Toc38987567)

# Úvod

Cílem řešeného projektu byl návrh a implementace síťového analyzátoru. Tento analyzátor bude schopný na určitém síťovém rozhraní zachytávat a filtrovat pakety. Také bylo nutné vytvořit k projektu dokumentaci.

# Vytvoření základních objektů schématu databáze

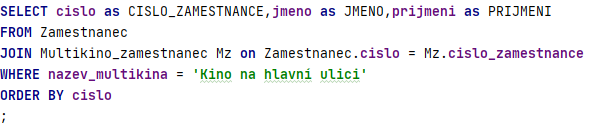
# Implementace několika dotazů SELECT

Tyto dotazy nám umožnují zobrazení vybraných dat pro konkrétní účely. Pro lehčí získaní potřebných informací. Celkově jsme vytvořili 10 těchto dotazů. A to abychom splnili upřesňující požadavky zadání, které by se dali rozdělit do pěti kategorii.

## Dva dotazy využívající spojení dvou tabulek

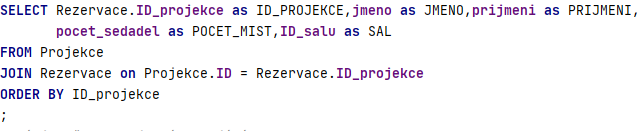
Pomocí JOIN jsme umožnili spojení tabulek za pomocí například stejných identifikátorů.

* Dotaz pro výpis zaměstnanců pracujících pro určité multikino(Kino na hlavní ulici).

Díky WHERE jsme schopni konkretizovat informace, které chceme zobrazit.  


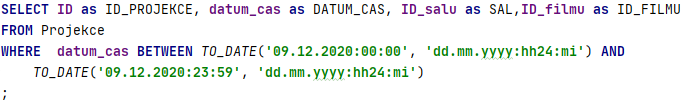
* Dotaz pro výpis rezervací na danou projekci .

Díky ORDER BY jsme schopni seřadit informace, které chceme zobrazit.



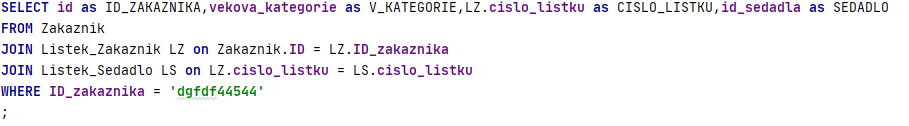
* Dotaz pro výpis všech projekcí konajících se určitý den (09.12.2020).

Zde jsme využili BETWEEN a AND pro konkretizaci data zobrazených projekcí.



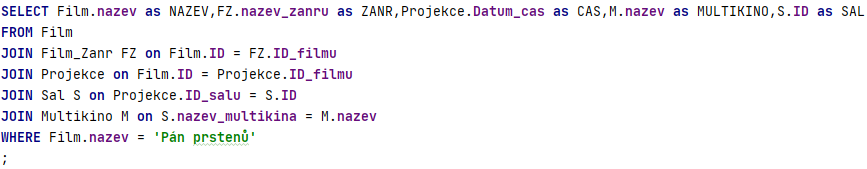
## Jeden dotaz využívající spojení tří tabulek

* Dotaz pro zjištění veškerých informací o lístku zákazníka a zákazníkovi samotném(například při kontrole lístku) id zákazníka = 'dgfdf44544' .



Spojení pěti tabulek

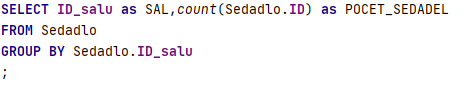
* Dotaz pro zobrazení všech promítaní filmu (Pán prstenů).



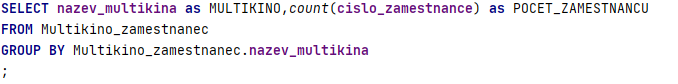
## Dva dotazy s klauzulí GROUP BY a agregační funkcí

Pomocí funkce count() jsme schopni získat počet námi zvolených dat.

* Dotaz pro zobrazení počtu poskytnutelných sedadel v určitém sálu.



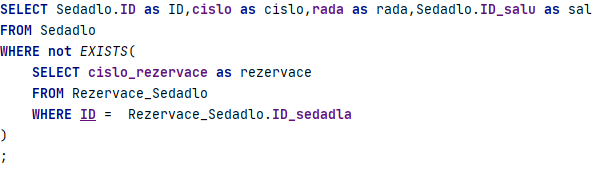
* Dotaz pro zobrazení počtu zaměstnanců v daných multikinech

****

## Jeden dotaz obsahující predikát EXISTS

Zde využíváme vnořeného dotazu SELECT

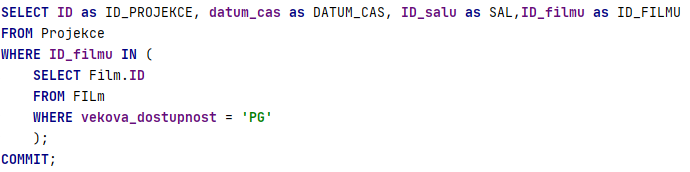
* Dotaz pro výpis sedadel na, která nejsou vypsány rezervace.



## Jeden dotaz s predikátem IN s vnořeným selectem

Zde využíváme vnořeného dotazu SELECT

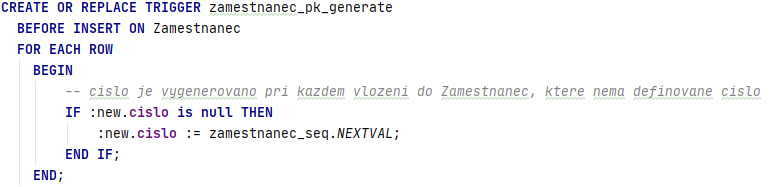
* Dotaz pro výpis všech projekcí s věkovou dostupností PG



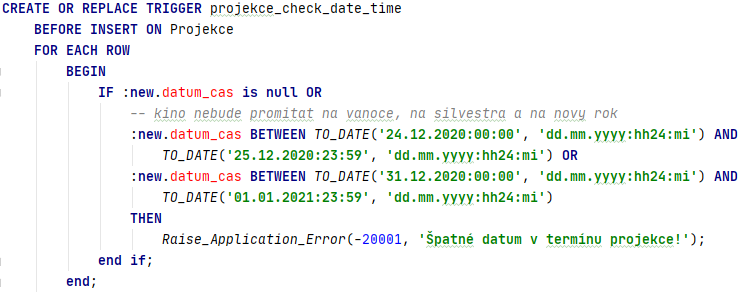
# Vytvoření netriviálních databázových triggerů

Implementovali jsme 2 typy triggerů.

* První trigger generuje primární klice ze sekvence (generovaní čísla zaměstnance) postará se o to, aby číslo nemohlo byt nedefinované.



* Druhý trigger, který se spustí vždy přidáni projekce, kontroluje zadávaní času a data projekce.



# Vytvoření netriviálních uložených procedur

Implementovali jsme dva typy procedur.

První procedura vypočítá průměrnou cenu za jeden lístek, argument slouží pro specifikaci projekce (ID). Pro získání této ceny první inkrementujeme počet zakoupených lístků a následně sčítáme i jejich ceny. Po sečtení všech lístků, vypočítáme průměrnou cenu za jeden lístek.

Druhá procedura k vypočtu procentuálního zastoupeni publika u filmu (všech projekci filmu). Při této proceduře zjišťujeme věkovou kategorii všech zákazníků, kteří navštívili konkrétní film. Zjistíme jejich počet pomocí inkrementace konkrétních čítačů a následně vypočítáme jejich procentuální zastoupení.

# Použití EXPLAIN PLAN a INDEX

Využili jsme možnosti kombinace EXPLAIN PLAN a INDEX, která byla naznačená v zadání.

EXPLAIN PLAN ukazuje detail určitého příkazu. U nás je tímto SELECT přes 2 tabulky, který ukáže seznam multikin a k nim průměrnou kapacitu sálů, kterými disponují. Po vytvoření INDEX by se mělo zrychlit vyhledávání, neboť systém nebude muset procházet všechny sály sekvenčně.

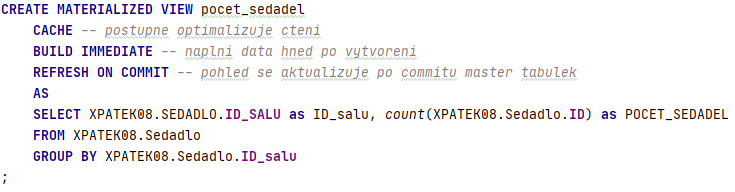
# Přístupová práva

V skriptu jsme také implementovali přidělení práv druhému (xcecha06) členovi týmu.

# Materializovaný pohled

Slouží na uložení často využívaného pohledu lokálně na disk, za účelem rychlého přístupu při opakovaném žádaní o tento pohled.

V naší implementaci se nachází materializovaný pohled pro přehled sedadel sálech.



# Závěr

Výsledkem našeho snažení je funkční SQL skript obsahující několik objektů, dotazů SELECT, TRIGGERŮ, PROCEDUR a EXPLAIN PLAN. Díky projektu jsme se seznámili se základy jazyku SQL a ovládání databází. Seznámili jsme se s programem DataGrip a školním serverem Oracle. Také jsme si vyzkoušeli spolupráci na projektu a rozdělení práce ve dvojici.