



Dokumentace ke projektu do IDS - 4.část  
Zadání číslo 28 - Nemocnice

Maxim Plička(xplick04)  
Jan Zdeněk(xzdene01)  
2. května 2022

# 1 Triggery

## 1.1 DeleteHistory

Tento trigger slouží k zálohování hodnot z tabulky `Pacients`. Trigger se spouští před smazáním kteréhokoli pacienta. Pro tento trigger byla vytvořena tabulka `PacientsHistory` do které se tyto hodnoty uloží. Kromě těchto hodnot se navíc uloží i datum, kdy bylo mazání provedeno.

## 1.2 CheckRoomCapacity

Trigger slouží k dodatečné kontrole před přidáním hospitalizace do databáze. Trigger kontroluje, jestli pokoj, na kterém se hospitalizace nachází, má dostatečný počet lůžek pro přidání pacienta. V případě, že je tento pokoj již plný, spustí příkaz `raise_application_error`.

```
Error starting at line : 336 in command -
INSERT INTO Hospitalizations (StartTime, DoctorID, PatientID, RoomID)
VALUES (TO_DATE('2021-03-31', 'yyyy/mm/dd'), 2, 1, 1)
Error report -
ORA-20000: Room is already full.
ORA-06512: na "XPLICK04.CHECKROOMCAPACITY", line 10
ORA-04088: chyba během provádění triggeru 'XPLICK04.CHECKROOMCAPACITY'
```

Obrázek 1: Chyba při přidání pacienta na pokoj, který je plný.

# 2 Procedury

## 2.1 GetPatientHeader ()

Tato procedura byla navržena pro vypsání "hlavičky" daného pacienta například při odesílání dopisů. Jako parametr si bere `PatientID`. Následně se založí proměnné `FirstNameOUT`, `LastNameOUT`, `StreetOUT`, `CityOUT` a `ZipCodeOUT`. Tyto proměnné jsou stejného typu jako mají odpovídající sloupce v tabulce `Pacients`. Poté jsou tyto proměnné naplněny hodnotami a následně se v odpovídajícím formátu vypíší na výstup. V případě, že neexistuje `PatientID`, které je předáno v argumentu nastane výjimka.

```
Andrew Denim
Oxford 2
England
11900
```

```
Call completed.
```

Obrázek 2: Výpis informací o pacientovi v daném formátu.

```
Cannot find Pacient by his ID
```

```
Call completed.
```

Obrázek 3: Výjimka při přístupu k pacientovi, který není v databázi.

## 2.2 DoctorStats()

Procedura složí k vypsání hospitalizací a prohlídek, které provedli jednotliví doktoři. Pro realizaci bylo využito kurzoru, do kterého se na začátku uloží tabulka `Doctors`. Následně je tato tabulka procházena po jednotlivých záznamech. Během každého průchodu se do proměnné `singleRow` uloží aktuálně procházený řádek a dohledá se počet prohlídek a hospitalizací. Tyto počty jsou uloženy do proměnných `HospiCount` a `ExamiCount`. Nakonec se všechny tyto informace vypíše na výstup. V případě chyby se na výstup vypíše chybová hláška.

```
Doctors statistics:  
Doctor Jan Pokora has done 3 hospitalizations and 2 examinations.  
Doctor Janis Pokora has done 3 hospitalizations and 1 examinations.  
Doctor John Petrohrad has done nothing.
```

```
Call completed.
```

Obrázek 4: Ukázka výpisu statistik doktorů.

## 3 Explain plan

Zobrazuje detaily při zpracování daného příkazu jako například posloupnost vykonávání operací. V našem případě jsme pro optimalizaci využili indexů, díky kterým SQL engine nemusí procházet všechny záznamy sekvenčně, ale pouze si v tabulce indexací dohledá dané pozice a přistoupí k nim. Tento způsob má významný vliv na cenu a čas, bohužel ale v naší tabulce nemáme dostatek hodnot, aby to bylo znatelné.

Id	Operation	Name	Rows	Bytes	Cost	(%CPU)	Time
0	SELECT STATEMENT		1	66	7	(15)	00:00:01
1	HASH GROUP BY		1	66	7	(15)	00:00:01
2	NESTED LOOPS		1	66	6	(0)	00:00:01
3	NESTED LOOPS		3	66	6	(0)	00:00:01
4	TABLE ACCESS FULL	HOSPITALIZATIONS	3	78	3	(0)	00:00:01
* 5	INDEX UNIQUE SCAN	SYS_C001654452	1		0	(0)	00:00:01
PLAN_TABLE_OUTPUT							
* 6	TABLE ACCESS BY INDEX ROWID	PACIENTS	1	40	1	(0)	00:00:01

Obrázek 5: Explain plan před zavedením indexů.

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	66	2 (50)	00:00:01
1	HASH GROUP BY		1	66	2 (50)	00:00:01
2	NESTED LOOPS		1	66	1 (0)	00:00:01
3	INDEX FULL SCAN	HOSPITALIZATIONSIDX	3	78	1 (0)	00:00:01
* 4	INDEX RANGE SCAN	PACIENTSIDX	1	40	0 (0)	00:00:01

Obrázek 6: Explain plan po zavedení indexů.

## 4 Přístupová práva

Práva na všechny tabulky přidělil vedoucí týmu xplick04 pomocí příkazu GRANT ALL ON svému kolegovi xzdene01.

## 5 Materializovaný pohled

Materializovaný pohled obsahuje Jméno, příjmení a počet léků, které pacient užívá. Byl založen členem xzdene01 a používá tabulky vedoucího týmu xplick04. Pohled se v naší demonstraci aktualizuje při vložení nového záznamu do tabulky PatientsUsesMedicine a zadání příkazu COMMIT.

FIRSTNAME	LASTNAME	POCETLEKU
Andrew	Denim	2
Jan	Latka	1
Kalhoty	Latka	2

Obrázek 7: Materializovaný pohled před vložení záznamu.

FIRSTNAME	LASTNAME	POCETLEKU
Andrew	Denim	3
Jan	Latka	1
Kalhoty	Latka	2

Obrázek 8: Materializovaný pohled po vložení záznamu.