

# **ADMINISTRACIJA BAZA PODATAKA**

---

## **INFORMATION SCHEMA**

(Skup sistemskih pogleda za pristup metapodacima  
relacionih baza podataka )

---

# Information Schema - Uvod

---

- INFORMATION\_SCHEMA obezbeđuje mogućnost da se na standardizovan način dođe do metapodataka u različitim SUBP-ovima.
    - Metapodaci?
  - INFORMATION\_SCHEMA se nalazi u većini modernih Relacionih SUBP. Omogućava korisnicima da pristupe metapodacima o strukturi i sadržaju baze podataka.
  - **INFORMATION\_SCHEMA – Nije baza podataka**, već skup sistemskih pogleda. Omogućava korisnicima da pristupe metapodacima o strukturi i sadržaju baze podataka.
-

# Šta je information schema?

---

- INFORMATION\_SCHEMA se koristi za prikazivanje informacija o objektima baze podataka kao što su tabele, pogledi, ključevi, ograničenja, funkcije, indeksi itd.
  - Administrator baze podataka (i svi korisnici kojima je da administratorska prava u okviru instance SUBP-a) mogu koristiti ovaj sistemski skup pogleda za pregled podataka koje koristi sam SUBP .
  - INFORMATION\_SCHEMA se može koristiti za standardizovan pristup rečniku različitih relacionih SUBP, jer je podržava većina modernih relacionih SUBP, kao što su: SQL Server, MySQL, PostgreSQL, itd.
-

# Pristup REČNIKU preko Information Schema-e

---

- Rečnik, odnosno sistemska baza podataka različitih SUBP, sadrži slične ili gotovo iste podatke, međutim njihova organizacija je drugačija od jednog do drugog SUBP-a.
  - Uvođenjem koncepta Information Schema-e obezbeđuje se da, se skrivajući implementacione detalje konkretnog SUBP-a, rečniku pristupa na potpuno isti funkcionalan način korišćenjem pogleda Information Scheme.
  - Pristup rečniku preko pogleda Information Scheme je potpuno isti kod svih SUBP-ova koji je podržavaju - (to je smisao i cilj uvođenja koncepta Information Schema-e)
-

# Kako se koristi Informatio schema?

---

- ☐ U SUBP-ovima koji je podržavaju postoji skup pogleda u okviru Information Schema-e.
  - ☐ Preko tih pogleda Information schema-e se vide metapodaci o samim bazama podataka i njihovim objektima.
  - ☐ Sistemske poglede iz Information schema-e moduće je koristiti za upite i preglede sistemskih (meta) podataka, ali se preko toh pogleda metapodaci ne mogu menjati (ažurirati).
-

# Zašto je uveden?

---

- ☐ Bio je potreban standard za pregled metapodataka u SUBP-ovima
  - ☐ Pre information schema-e su se za pregled metapodataka koristile komande.
  - ☐ Metapodacima u svim komercijalnim SUBP (koji podržavaju koncept INFORMATION\_SCHEMA) pristupa se na isti način bez obzira na različitu organizaciju sistemskih baza podataka svakog SUBP.
-

# **Standard**

---

## **Database languages - SQL Standard**

- ISO/IEC 9075 standard— je standard SQL upitnog jezika

- Ukupno je usvojeno 7 standarda do sada

- Standard se deli na 14 delova

- 11.) *Information and Definition Schemas*  
(SQL/Schema)

---

# **Standard**

---

Neki od SUBP-ova koji podržavaju *Information* :

- MS SQL Server
    - od verzije 7 pa naviše
  - MySQL
    - od verzije 5 pa naviše
  - PostgreSQL
    - od verzije 7.4 pa naviše
  - SQLite
    - od verzije 3.7 pa naviše
  - Oracle?
-



# Pogledi – U okviru Information Schema-e

---

Nemaju svi SUBP-ovi isto tabela (pogleda) u Information\_Schema-i

- ❑ MySQL ima više tabela (pogleda) od ostalih SUBP-ova koji podržavaju Information Schema

Nek od pogleda su :

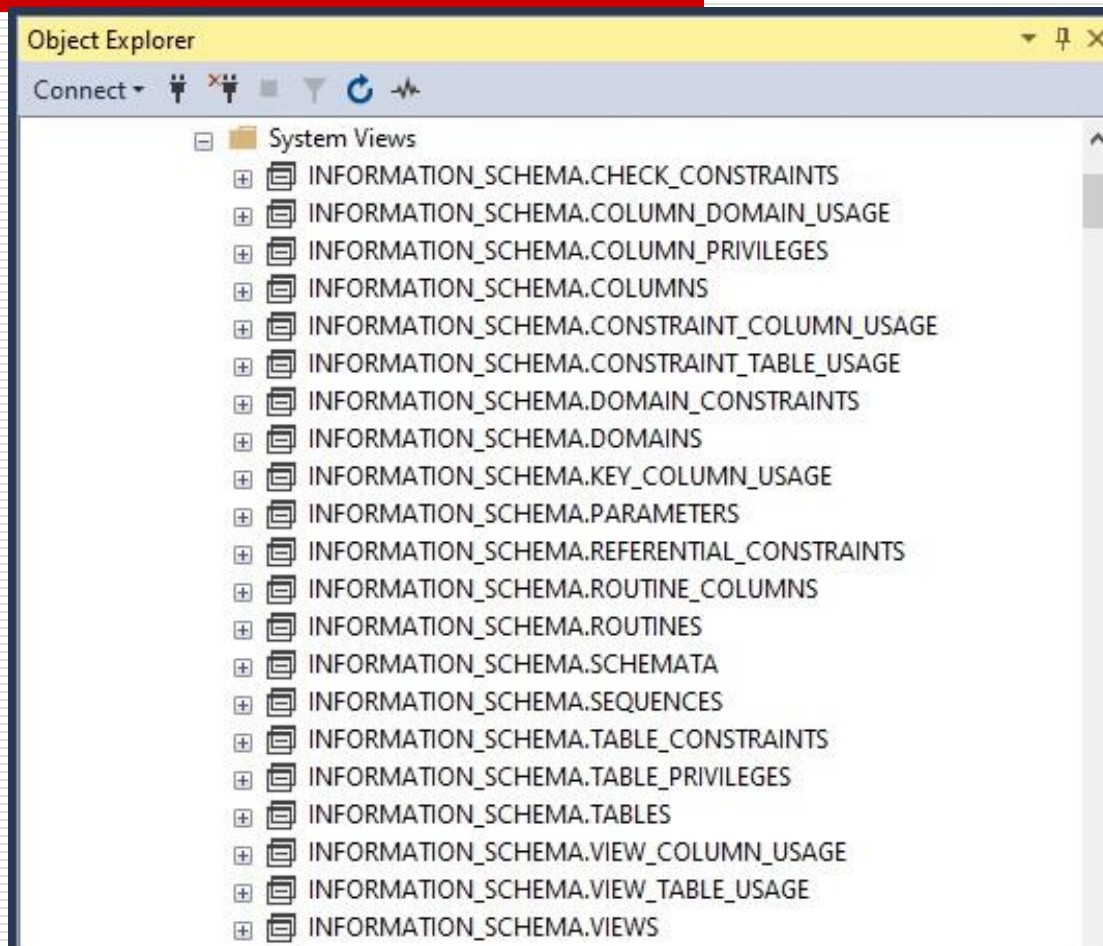
- *Tables , Columns , Key\_Column\_Usage , Views, Referential\_Constraints...*

- ❑ Svi pogledi u okviru Information Schema imaju kolone kojima je opisana svaka baza podataka kojom rukuje konkretna instanca SUBP-a.

- ❑ Na primer, pogled **Tables** ima sledeće kolone:

- TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, TABLE\_TYPE

# INFORMATION SCHEMA – Skup pogleda



# Primer 1 - Korišćenja Information Schema-e

Prikazati sva obeležja svih tabela baze podataka FAKULTET za koja je prilikom kreiranja šeme te baze podataka definisano da ne smeju poprimiti *NULL* vrednost.

	Column_Name	Table_Name	Is_Nullable	Data_Type
1	Br_Ind	Student	NO	char
2	Br_Ind	Položio	NO	char
3	Budzet	Projekat	NO	money
4	Casova	Predmet	NO	int
5	Datum	Položio	NO	datetime
6	GodStud	Student	NO	char
7	Ime	Student	NO	varchar
8	IznosHonorara	Ucestvuje	NO	money
9	Naziv_Kat	Katedra	NO	char
10	Naziv_Pred	Predmet	NO	varchar
11	Naziv_Pro	Projekat	NO	char
12	Ocena	Položio	NO	int

# Primer 1 - Korišćenja Information Schema-e

---

Uradićemo zadatak pomoću tabele ***COLUMNS*** , koja u sebi sadrži sva obeležja kolona svih tabela svih baza podataka. U sledećem primeru koristićemo kolone:

***Column\_name*** – Naziv kolone.

***Table\_name*** – Naziv tabele kojoj kolona pripada.

***Is\_nullable*** – Da li vrednost kolone sme biti null

***Data\_type*** – Tip podataka kolone.

---

# Primer 1 - Korišćenja Information Schema-e

---

SQLI kod:

```
Select Column_Name, Table_Name, Is_Nullable, Data_Type  
  From Information_Schema.Columns  
    Where Is_Nullable = 'NO'  
  Order by Column_Name
```

---

## Primer 2 - Korišćenja Information Schema-e

Prikazati sve relacije (Tabele) u bazi podataka Hotel.

Results		Messages
	TABLE_CATALOG	TABLE_NAME
1	Hotel	Cenovnik
2	Hotel	Gost
3	Hotel	Iznajmljivanje
4	Hotel	Soba
5	Hotel	TipIznajmljivanja
6	Hotel	TipSobe

## Primer 2 - Korišćenja Information Schema-e

---

Udadićemo zadatak pomoću pogleda ***TABLES koja sadrži sve*** sve tabele aktuelne baze podataka. Iskoristićemo sledeće kolone iz pogleda:

### **TABLES:**

***Table\_Catalog*** – Naziv baze podataka.

***Table\_Name*** – Naziv tabele u aktuelnoj bazi podataka.

***Table\_Type*** – Vrsta (tip) tabele (da li je bazna tabela ili izedena tabela, odnosno pogled)

---

# Primer 2 - Korišćenja Information Schema-e

---

SQL kod:

```
Select TABLE_CATALOG, TABLE_NAME, TABLE_TYPE  
      From INFORMATION_SCHEMA.Tables  
Order By TABLE_NAME
```



# Zadaci

---

1. Koristeci INFORMATION\_SCHEMA prikazati sve tabele u bazi podataka Fakultet.
  2. Koristeci INFORMATION\_SCHEMA prikazati sva obelezja tabele Nastavnik iz baze podataka Fakultet.
  3. Koristeci INFORMATION\_SCHEMA prikazati sve poglede u bazi podataka Fakultet.
-