

Izračunavanje procenata u SQL jeziku

Uvod i sintaksa

Izračunavanje procenata u SQL jeziku se često koristi za analizu i izveštavanje u bazama podataka. Korišćenjem funkcija kao što su **SUM()**, **COUNT()** i operatora poput **GROUP BY**, može se efikasno raditi sa procentualnim vrednostima.

Kako funkcioniše

SQL izračunavanje procenata generalno uključuje deljenje dela skupa podataka sa celokupnim setom i množenje rezultata sa 100 da bi se dobila procentualna vrednost.

Primeri upotrebe

Osnovni nivo: Izračunavanje učešća određenog proizvoda u ukupnoj prodaji:

```
SELECT
    Proizvod,
    ROUND((SUM(Kolicina) * 100.0) / (SELECT SUM(Kolicina) FROM
    Prodaja), 2) AS Procenat
FROM
    Prodaja
GROUP BY
    Proizvod;
```

Srednji nivo: Izračunavanje procenta prodaje po regionu u odnosu na ukupnu prodaju:

```
SELECT
    Region,
    ROUND((SUM(IznosProdaje) * 100.0) / (SELECT
    SUM(IznosProdaje) FROM Prodaja), 2) AS Procenat
FROM
    Prodaja
GROUP BY
    Region;
```

Napredni nivo: Poređenje procentualne prodaje između različitih godina:

```
SELECT
    YEAR(DatumProdaje) AS Godina,
    ROUND((SUM(IznosProdaje) * 100.0) / (SELECT
    SUM(IznosProdaje) FROM Prodaja), 2) AS Procenat
```

```
FROM
  Prodaja
GROUP BY
  YEAR(DatumProdaje);
```

Izračunavanje grupisanih procenta u SQL jeziku

Izračunavanje grupisanih procenta u SQL jeziku

Uvod i sintaksa

Grupisani procenti u SQL jeziku omogućavaju analizu odnosa delova podataka prema celini unutar određenih grupa. Ovo je korisno za dubinsku analizu podataka kako bi se razumeli trendovi i odnosi unutar različitih segmenata podataka.

Kako funkcioniše

Da biste izračunali grupisane procenete, koriste se podupiti ili privremene tabele za izračunavanje totala po grupi, a zatim se svaka vrednost u grupi poredi sa tim totalom da bi se dobio procentualni udeo.

Primeri upotrebe

Osnovni nivo: Izračunavanje procenta prodaje svakog proizvoda unutar svake kategorije:

```
SELECT
  Kategorija,
  Proizvod,
  SUM(IznosProdaje) AS UkupnaProdajaPoProizvodu,
  ROUND((SUM(IznosProdaje) * 100.0) / (SELECT
SUM(IznosProdaje) FROM Prodaja WHERE Kategorija =
p.Kategorija), 2) AS Procenat
FROM
  Prodaja p
GROUP BY
  Kategorija, Proizvod;
```

Srednji nivo: Izračunavanje procenta ukupnih troškova po odeljenju unutar firme:

```
SELECT
  Odeljenje,
  ROUND((SUM(Troskovi) * 100.0) / (SELECT SUM(Troskovi) FROM
Finansije), 2) AS Procenat
FROM
```

```

Finansije
GROUP BY
    Odeljenje;

```

Napredni nivo: Upotreba CTE (Common Table Expression) za izračunavanje procenta prodaje po regionu, poredjenje godina:

```

WITH RegionalnaProdaja AS (
    SELECT
        Region,
        YEAR(DatumProdaje) AS Godina,
        SUM(IznosProdaje) AS Ukupno
    FROM
        Prodaja
    GROUP BY
        Region, YEAR(DatumProdaje)
)
SELECT
    a.Region,
    a.Godina,
    a.Ukupno AS GodisnjaProdaja,
    ROUND((a.Ukupno * 100.0) / b.TotalGodina, 2) AS Procentat
FROM
    RegionalnaProdaja a
JOIN
    (SELECT Godina, SUM(Ukupno) AS TotalGodina FROM
    RegionalnaProdaja GROUP BY Godina) b
ON
    a.Godina = b.Godina;

```

Filtriranje podataka na osnovu procenta u SQL jeziku

Uvod i sintaksa

Filtriranje podataka na osnovu procentualnih uslova u SQL jeziku koristi se za izdvajanje delova podataka koji zadovoljavaju određene procentualne kriterijume. Ova tehnika je korisna u analizi efikasnosti, kontrolama kvaliteta, segmentaciji tržišta i drugim situacijama gde je relevantan relativni udeo neke vrednosti.

Kako funkcioniše

Za filtriranje na osnovu procenta, često se koriste podupiti ili CTE (Common Table Expression) za izračunavanje ukupnih vrednosti po grupama, a potom se primenjuju kriterijumi u **WHERE** klauzuli da bi se izdvojili podaci koji ispunjavaju procentualne uslove.

Primeri upotrebe

Osnovni nivo: Filtriranje proizvoda koji čine više od 5% ukupne prodaje:

```
SELECT
    Proizvod,
    SUM(IznosProdaje) AS ProdajaPoProizvodu,
    (SUM(IznosProdaje) * 100.0 / (SELECT SUM(IznosProdaje) FROM
Prodaja)) AS Procenat
FROM
    Prodaja
GROUP BY
    Proizvod
HAVING
    (SUM(IznosProdaje) * 100.0 / (SELECT SUM(IznosProdaje) FROM
Prodaja)) > 5;
```

Srednji nivo: Filtriranje prodajnih podataka za regione koji doprinose manje od 10% ukupnim godišnjim prodajama:

```
WITH RegionalnaProdaja AS (
    SELECT
        Region,
        SUM(IznosProdaje) AS UkupnaProdaja,
        (SUM(IznosProdaje) * 100.0 / (SELECT SUM(IznosProdaje)
FROM Prodaja)) AS Procenat
    FROM
        Prodaja
    GROUP BY
        Region
)
SELECT
    Region,
    UkupnaProdaja,
    Procenat
FROM
    RegionalnaProdaja
WHERE
    Procenat < 10;
```

Napredni nivo: Filtriranje meseci sa prodajom koja čini manje od 3% godišnje prodaje:

```
WITH MesecnaProdaja AS (
    SELECT
        MONTH(DatumProdaje) AS Mesec,
        YEAR(DatumProdaje) AS Godina,
        SUM(IznosProdaje) AS MesečnaProdaja,
        (SUM(IznosProdaje) * 100.0 / (SELECT SUM(IznosProdaje)
FROM Prodaja WHERE YEAR(DatumProdaje) =
YEAR(p.DatumProdaje))) AS Procenat
```

```

FROM
    Prodaja p
GROUP BY
    MONTH (DatumProdaje), YEAR (DatumProdaje)
)
SELECT
    Mesec,
    Godina,
    MesečnaProdaja,
    Procenat
FROM
    MesecnaProdaja
WHERE
    Procenat < 3;

```

Performanse

1. **Indeksiranje:** Indeksiranje na poljima kao što su **DatumProdaje** ili **Region** može poboljšati efikasnost grupisanja i filtriranja.
2. **Korišćenje CTE-ova:** CTE može poboljšati čitljivost i potencijalno performanse upita tako što olakšava razumevanje logike upita i omogućava višestruko korišćenje rezultata.

Još akcija u radu sa procentima

U radu sa procentima u SQL jeziku, osim izračunavanja i filtriranja, moguće je obavljati i druge akcije koje omogućavaju bolju analizu i interpretaciju podataka. Evo nekoliko dodatnih akcija koje se često koriste:

1. Rangiranje na osnovu procenta

Rangiranje podataka prema procentualnim vrednostima može pomoći u identifikaciji koje kategorije, proizvodi ili regioni zauzimaju najviša ili najniža mesta u smislu učinka ili doprinosa.

Primer:

```

WITH ProcentualnaProdaja AS (
    SELECT
        Proizvod,
        SUM(IznosProdaje) AS ProdajaPoProizvodu,
        (SUM(IznosProdaje) * 100.0 / (SELECT SUM(IznosProdaje)
FROM Prodaja)) AS Procenat
FROM
    Prodaja

```

```

GROUP BY
    Proizvod
)
SELECT
    Proizvod,
    ProdajaPoProizvodu,
    Procenat,
    RANK() OVER (ORDER BY Procenat DESC) AS Rang
FROM
    ProcentualnaProdaja;

```

2. Kumulativni procenti

Kumulativni procenti se koriste za analizu kako se akumulirani učinak povećava sa svakom grupom ili kategorijom.

Primer:

```

SELECT
    Proizvod,
    SUM(IznosProdaje) AS ProdajaPoProizvodu,
    SUM(SUM(IznosProdaje)) OVER (ORDER BY SUM(IznosProdaje)
DESC) * 100.0 / (SELECT SUM(IznosProdaje) FROM Prodaja) AS
KumulativniProcenat
FROM
    Prodaja
GROUP BY
    Proizvod
ORDER BY
    ProdajaPoProizvodu DESC;

```

3. Poređenje procentualnih promena

Poređenje procentualnih promena između različitih perioda može dati uvid u trendove i promene u performansama.

Primer:

```

WITH GodisnjaProdaja AS (
    SELECT
        YEAR(DatumProdaje) AS Godina,
        SUM(IznosProdaje) AS UkupnaProdaja
    FROM
        Prodaja
    GROUP BY
        YEAR(DatumProdaje)
)
SELECT
    a.Godina,
    a.UkupnaProdaja,
    (a.UkupnaProdaja - b.UkupnaProdaja) * 100.0 /
b.UkupnaProdaja AS PromenaProcenta

```

```
FROM
  GodisnjaProdaja a
JOIN
  GodisnjaProdaja b ON a.Godina = b.Godina + 1;
```

4. Vizualizacija procentualnih podataka

Kreiranje vizualnih prikaza procentualnih distribucija, kao što su grafikoni u alatima za izveštavanje, može poboljšati interpretaciju podataka.

5. Analiza uslova na osnovu procentualnih praga

Postavljanje određenih procentualnih pragova može pomoći u identifikaciji anomalija, izuzetaka ili važnih obrazaca u podacima.

Primer:

```
SELECT
  Kategorija,
  Proizvod,
  SUM(IznosProdaje) AS ProdajaPoProizvodu,
  (SUM(IznosProdaje) * 100.0 / (SELECT SUM(IznosProdaje) FROM
  Prodaja WHERE Kategorija = p.Kategorija)) AS Procenat
FROM
  Prodaja p
GROUP BY
  Kategorija, Proizvod
HAVING
  Procenat > 10;
```

6. Procentualni udeo prodaje po zaposlenom

Ovaj upit prikazuje kako možete izračunati procentualni udeo svakog zaposlenog u ukupnoj prodaji firme:

```
WITH UkupnaProdaja AS (
  SELECT
    SUM(IznosProdaje) AS Total
  FROM
    Prodaja
)
SELECT
  Zaposleni.Ime,
  SUM(Prodaja.IznosProdaje) AS ProdajaZaposlenog,
  ROUND((SUM(Prodaja.IznosProdaje) * 100.0 /
  UkupnaProdaja.Total), 2) AS ProcenatProdaje
FROM
  Prodaja
JOIN
  Zaposleni ON Prodaja.ZaposleniID = Zaposleni.ID
CROSS JOIN
  UkupnaProdaja
```

```
GROUP BY
  Zaposleni.Ime, UkupnaProdaja.Total;
```

7. Procentualno uvećanje cena proizvoda

Ako želite da analizirate kako su se cene proizvoda menjale tokom vremena, možete koristiti sledeći upit koji izračunava procentualno povećanje cena između dva perioda:

```
WITH CenePoPeriodima AS (
  SELECT
    ProizvodID,
    Period,
    AVG(Cena) AS ProsečnaCena
  FROM
    ProdajnaIstorija
  GROUP BY
    ProizvodID, Period
)
SELECT
  Trenutni.Period AS TrenutniPeriod,
  Prethodni.Period AS PrethodniPeriod,
  Trenutni.ProizvodID,
  Trenutni.ProsečnaCena AS TrenutnaCena,
  Prethodni.ProsečnaCena AS PrethodnaCena,
  ROUND(((Trenutni.ProsečnaCena - Prethodni.ProsečnaCena) /
  Prethodni.ProsečnaCena) * 100, 2) AS PromenaProcenta
FROM
  CenePoPeriodima Trenutni
JOIN
  CenePoPeriodima Prethodni ON Trenutni.ProizvodID =
  Prethodni.ProizvodID
WHERE
  Trenutni.Period = '2023' AND Prethodni.Period = '2022';
```

8. Filtriranje odseka po procentualnom udelu zaposlenih koji su postigli cilj

Ovaj upit pokazuje kako možete filtrirati odseke u firmi na osnovu procenta zaposlenih koji su postigli svoje prodajne ciljeve:

```
WITH Ciljevi AS (
  SELECT
    Odsek,
    COUNT(*) AS UkupnoZaposlenih,
    SUM(CASE WHEN Prodaja >= Cilj THEN 1 ELSE 0 END) AS
  ZaposleniKojiSuPostigliCilj
  FROM
    Zaposleni
  GROUP BY
    Odsek
```



```

)
SELECT
    Odsek,
    ZaposleniKojiSuPostigliCilj,
    UkupnoZaposlenih,
    ROUND((ZaposleniKojiSuPostigliCilj * 100.0 /
UkupnoZaposlenih), 2) AS ProcentatPostignuća
FROM
    Ciljevi
WHERE
    (ZaposleniKojiSuPostigliCilj * 100.0 / UkupnoZaposlenih) >
75;

```

Saveti vezanih za rad sa procentima

Rad sa procentima u SQL-u može biti ključan za razumevanje i analizu podataka. Evo deset saveta koji će vam pomoći da efikasnije radite sa procentima:

- **Koristite precizne tipove podataka:** Za rad sa procentima, koristite decimalni ili numerički tip podataka da biste izbegli greške u preciznosti koje mogu nastati korišćenjem tipa podataka float.
- **Izbegavajte deljenje nulom:** Uvek proverite delilac pre deljenja da biste izbegli grešku deljenja nulom. To možete uraditi koristeći funkciju **NULLIF** koja vraća NULL ako su oba argumenta jednaka.

```
SELECT Iznos / NULLIF(UkupanIznos, 0) FROM tabela;
```

- **Koristite funkciju ROUND za formatiranje procenta:** Da bi procenti bili čitljiviji i lakši za interpretaciju, koristite funkciju **ROUND()** za ograničavanje broja decimala.

```
SELECT ROUND((vrednost / total) * 100, 2) AS Procent FROM
tabela;
```

- **Upotreba CTE za složene izračunavanja:** Ako vaš upit zahteva više koraka ili ponovljeno korišćenje istih rezultata, razmislite o upotrebi Common Table Expressions (CTE) kako biste poboljšali čitljivost i organizovanost upita.

- **Upotreba podupita za totalne vrednosti:** Ako vam trebaju totalne vrednosti za izračunavanje procenta unutar istog upita, koristite podupite unutar SELECT naredbe.

```
SELECT Proizvod, Cena, (Cena * 100.0 / (SELECT SUM(Cena) FROM Prodaja)) AS Procentat FROM Prodaja;
```

- **Analizirajte trendove koristeći procente:** Kada analizirate trendove, koristite procentualne promene kako biste identifikovali značajne promene ili anomalije u podacima.
- **Oprez pri interpretaciji rezultata:** Procenti mogu biti varljivi, posebno ako se baziraju na malim uzorcima. Uvek razmatrajte kontekst podataka pre donošenja zaključaka.
- **Automatizacija izračunavanja procenta:** Razmotrite korišćenje skripti ili funkcija koje automatski izračunavaju i ažuriraju procentualne vrednosti u vašim bazama podataka, posebno kada su podaci često ažurirani.
- **Testiranje upita na manjim skupovima podataka:** Pre implementacije upita na velike skupove podataka, testirajte svoje SQL upite na manjim skupovima kako biste osigurali da su logika i performanse upita zadovoljavajuće.
- **Edukacija o značaju procenta:** Osigurajte da svi korisnici koji interpretiraju izveštaje razumeju kako su procenti izračunati i šta oni znače u kontekstu poslovnih operacija.
- **Koristite prozorske funkcije za procentualne izračunavanja:** Prozorske funkcije poput SUM() OVER() mogu vam pomoći da izračunate kumulativne ili procentualne vrednosti unutar određene grupe bez potrebe za podupitima.

```
SELECT Proizvod, Cena, SUM(Cena) OVER (PARTITION BY Kategorija) AS UkupnaCenaKategorije FROM Prodaja;
```

- **Pazite na grupisanje:** Kada koristite **GROUP BY**, uverite se da su sve potrebne kolone uključene kako biste izbegli greške u logici ili rezultatima.
- **Optimizujte upite koristeći indeksiranje:** Pravilno indeksiranje kolona koje se često koriste u kalkulacijama procenta može znatno poboljšati performanse upita.

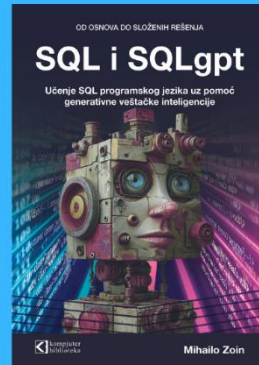
- **Iskoristite privremene tabele za složene izračunavanja:** Ako vaš upit zahteva više složenih koraka, razmotrite upotrebu privremenih tabela da biste izbegli ponavljanje izračunavanja i smanjili opterećenje na server.
- **Koristite logičke operatore za kontrolu uslova:** Pri filtriranju podataka na osnovu procentualnih uslova, koristite logičke operatore kao što su **AND**, **OR** za definisanje složenijih kriterijuma.
- **Verifikujte podatke:** Pre nego što počnete sa procentualnim izračunavanjima, proverite podatke za eventualne greške ili anomalije koje bi mogle uticati na tačnost vaših rezultata.
- **Razmatrajte uticaj ekstremnih vrednosti:** U situacijama gde ekstremne vrednosti mogu narušiti procentualne analize, razmotrite upotrebu statističkih metoda poput mediane umesto proseka.
- **Kombinujte procenite sa apsolutnim vrednostima:** Prikazivanje procentualnih vrednosti zajedno sa apsolutnim vrednostima može pružiti dublji uvid u značaj promena ili trendova.
- **Automatizacija izveštavanja:** Razmotrite automatizaciju procesa generisanja izveštaja koji uključuju procentualne analize kako bi se smanjio ručni rad i povećala konzistentnost.
- **Edukujte tim:** Osigurajte da svi korisnici koji rade s izveštajima razumeju kako su procenti izračunati i kako ih interpretirati u kontekstu većih podatkovnih setova.

Zaključak o izračunavanju procenata

Izračunavanje procenata u SQL-u omogućava organizacijama da efikasno kvantifikuju odnose između podskupova podataka i celokupnih skupova. Ovo može pomoći u donošenju obrazovanih odluka zasnovanih na preciznim i relevantnim informacijama. Procenti mogu ilustrovati udeo, rast, smanjenje, i doprinose specifičnih kategorija, proizvoda ili regiona unutar šireg poslovnog ekosistema.

Ukoliko još niste kupili knjigu

Naučiti sintaksu SQL komandi je važan prvi korak u razumevanju rada sa bazama podataka, ali to nije dovoljno za potpuno razumevanje i efikasno upravljanje bazama podataka. Pored sintakse, važno je razumeti koncepte relacionih baza podataka, načine na koje se podaci strukturiraju i povezuju, kao i različite aspekte performansi, sigurnosti i optimizacije baza podataka. Takođe, veštine u rešavanju problema, analitičko razmišljanje, i razumevanje poslovnih potreba su ključni za efikasnu primenu SQL-a u realnim scenarijima.



Više o knjizi i link za naručivanje:

<https://knjige.kombib.rs/sql-i-sqlgpt-od-osnova-do-slozenih-resenja>

Link za korišćenje SQLgpt-a za dodatna objašnjenja:

<https://chat.openai.com/g/g-b1VoJCEIT-sql-gpt>