

Operatorii *union*, *intersect* și *except*

1. Obiectivul lucrării:

Formarea și dezvoltarea abilităților de interogare a bazelor de date.

2. Breviar teoretic cu exerciții și probleme rezolvate

Operatorul UNION

Operatorul UNION efectuează operația de reuniune a rezultatelor a două interogări într-un singur rezultat.

Sintaxa:

```
fraza select
    UNION [ALL]
fraza select
    UNION [ALL]
fraza select
... .
```

1. UNION – Returnează rezultatele celor două interogări eliminând rândurile duplicat;
2. UNION ALL – Returnează rezultatele celor două interogări incluzând rândurile duplicat;

Există următoarele reguli de folosire a operatorului UNION:

- interogările trebuie să conțină același număr de coloane;
- coloanele corespondente trebuie să aibă același tip de dată;
- în rezultat vor apărea numele coloanelor din prima interogare, nu cele din a doua interogare chiar dacă aceasta folosește alias-uri;
- prima interogare poate conține clauza *into tabel_nou*, pentru crearea unui tabel care să conțină setul rezultat din reuniune;
- clauza ORDER BY poate fi folosită o singură dată și trebuie poziționată la sfârșitul comenzii.

Operatorul INTERSECT

Operatorul INTERSECT returnează intersecția rezultatelor a două interogări. Sunt furnizate valori distincte.

```
Fraza select  
    INTERSECT  
Fraza select  
    INTERSECT  
Fraza select  
    ...
```

Operatorul EXCEPT

Operatorul EXCEPT returnează valorile distincte obținute prin diferența dintre seturile de rezultate furnizate de două interogări.

```
fraza select  
    EXCEPT  
fraza select  
    EXCEPT  
fraza select  
    ...
```

Regulile de folosire a operatorului UNION, rămân valabile și pentru operatorii INTERSECT și EXCEPT

Operatorii INTERSECT și EXCEPT au fost introduși începând cu SQL Server 2008.

Exemple:

Presupunând că angajații firmei noastre sunt încadrați în două filiale și memorați în baze de date diferite(`dbFiliala1`, `dbFiliala2`), să se afișeze lista tuturor angajaților

```
select Marca,Nume,Prenume  
from dbFiliala1.dbo.tAngajati  
    union  
select Marca,Nume,Prenume  
from dbFiliala2.dbo.tAngajati  
order by 2,3
```

Observatie: clauza

```
order by 2,3
```

este echivalentă cu

```
order by Nume,Prenume
```

Să se determine studenții care au promovat examenele la cursurile 'c1' și 'c2'.

```
select tStudenti.* from tStudenti
inner join
(select codStd from tNote
where CodCurs='c1' and Nota>=5
intersect
select codStd from tNote
where CodCurs='c2' and Nota>=5) as A
on tStudenti.CodStd=A.CodStd
```

Să se determine studenții care au promovat examenul la cursul 'c1' dar nu au promovat examenul la cursul 'c2'.

```
select tStudenti.* from tStudenti
inner join
(select codStd from tNote
where CodCurs='c1' and Nota>=5
except
select codStd from tNote
where CodCurs='c2' and Nota>=5) as A
on tStudenti.CodStd=A.CodStd
```