

LABORATORUL 6

UNIRI(JOIN)

Instrucțiunea **SELECT** este utilizată pentru a selecta date din două sau mai multe tabele. Numele acestor tabele sunt incluse în clauza **FROM**. Adăugarea unei clauze **WHERE** dă o condiție de unire între cel puțin o coloană din fiecare tabel. Clauza **WHERE** creează un tabel compus temporar, în care perechile de linii care satisfac condiția de unire sunt legate și formează o singură linie.

Unirea *simplă* combină informațiile din două sau mai multe tabele, folosind câte o coloană din fiecare tabel. O unire *compusă* unește două sau mai multe tabele în care legăturile sunt determinate de două sau mai multe coloane din fiecare tabel.

Ca să realizăm o unire trebuie să specificăm legătura, numită condiție de unire, dată de cel puțin o coloană din fiecare tabel. Deoarece datele din coloanele celor două tabele sunt comparabile ele trebuie să aibă tipuri de date comparabile.

Conceptual, în prima etapă a oricărei uniri se creează un produs cartezian și apoi se verifică liniile de date care nu verifică condiția de unire din instrucțiunea **SELECT**.

O interogare poate obține date din unul sau mai multe tabele. Printr-o unire se pot obține rezultate din două sau mai multe tabele. Expresia utilizată la unirea a două tabele se numește criteriu de unire. Unirile pot fi interne sau externe. Unirile externe produc rezultate chiar și atunci când criteriul de unire nu este îndeplinit.

O **echiunire (EQUIJOIN)** este o unire în care se verifică dacă valorile din coloanele comune sunt egale iar ca rezultat sunt afișate și coloanele redundante. Egalitatea este indicată de operația de comparare din clauzele **ON** sau **WHERE** cu semnul **=**.

INNER JOIN

```
SELECT S.Nume,S.Prenume,C.cod_disciplina,C.Nota,C.Data  
FROM Student S INNER JOIN Catalog C  
ON S.NrLeg=C.NrLeg
```

```
SELECT S.Nume,S.Prenume,C.cod_disciplina,C.Nota,C.Data  
FROM Student S, Catalog C
```

WHERE S.NrLeg=C.NrLeg

Echiunirea (Equijoin)

```
SELECT *  
FROM Student S, Catalog C  
WHERE S.NrLeg=C.NrLeg
```

```
SELECT *  
FROM Student S INNER JOIN Catalog C  
ON S.NrLeg=C.NrLeg
```

O **unire naturală** este o unire în care se verifică dacă valorile din coloanele comune sunt egale iar coloanele redundante sunt afișate o singură dată.

Unirea naturală(Natural join)

```
SELECT S.*,C.cod_disciplina,C.Nota,C.Data  
FROM Student S INNER JOIN Catalog C  
ON S.NrLeg=C.NrLeg
```

```
SELECT S.*,C.cod_disciplina,C.Nota,C.Data  
FROM Student S, Catalog C  
WHERE S.NrLeg=C.NrLeg
```

OUTER JOIN realizează o unire a unui tabel dominant cu un tabel subordonat. Se afișează toate liniile din tabelul dominant.

Unirea externă (Outer join)

Unirea externă la stanga (Left =▷◁)

```
SELECT S.NrLeg, Nume, Prenume, Nota  
FROM Student S LEFT OUTER JOIN Catalog C  
ON S.Nrleg=C.NrLeg
```

```
SELECT S.NrLeg, Nume, Prenume, Nota  
FROM Student S, Catalog C  
WHERE S.Nrleg=C.NrLeg
```

```
SELECT S.NrLeg, Nume, Prenume, Nota  
FROM Student S LEFT OUTER JOIN Catalog CM  
ON S.Nrleg=C.NrLeg AND C.Nrleg='101'
```

```
SELECT S.NrLeg, Nume, Prenume, Nota  
FROM Student S, Catalog C  
WHERE S.Nrleg=C.NrLeg  
AND C.Nrleg='101'
```

Unirea externă la dreapta (Right ▷◁=)

```
SELECT S.NrLeg, Nume, Prenume, Nota  
FROM Student S RIGHT OUTER JOIN Catalog C
```

ON S.Nrleg=C.NrLeg

SELECT S.NrLeg, Nume, Prenume, Nota
FROM Student S, Catalog C
WHERE S.Nrleg=C.NrLeg

SELECT S.NrLeg, Nume, Prenume, Nota
FROM Student S RIGHT OUTER JOIN Catalog C
ON S.Nrleg=C.NrLeg AND C.Nrleg='101'

SELECT S.NrLeg, Nume, Prenume, Nota
FROM Student S, Catalog C
WHERE S.Nrleg=C.NrLeg AND C.Nrleg='101'

SELECT C.cod_disciplina,Nota, D.cod_disciplina,Denumire
FROM Catalog C RIGHT OUTER JOIN Discipline D
ON C.cod_disciplina=D.cod_disciplina

SELECT C.cod_disciplina,Nota, D.cod_disciplina, Denumire
FROM Catalog C, Discipline D
WHERE C.cod_disciplina=D.cod_disciplina

SELECT C.cod_disciplina,Nota, D.cod_disciplina,Denumire
FROM Catalog C RIGHT OUTER JOIN Discipline D
ON C.cod_disciplina=D.cod_disciplina AND C.cod_disciplina='2'

SELECT C.cod_disciplina,Nota, D.cod_disciplina,Denumire
FROM Catalog C, Discipline D
WHERE C.cod_disciplina=D.cod_disciplina AND C.cod_disciplina='2'

Unirea completa Full =><=

SELECT S.NrLeg, Nume, Prenume, Nota
FROM Student S FULL OUTER JOIN Catalog C
ON S.Nrleg=C.NrLeg

SELECT C.cod_disciplina,Nota, D.cod_disciplina,Denumire
FROM Catalog C FULL OUTER JOIN Discipline D
ON C.cod_disciplina=D.cod_disciplina

Autounirea (Self join)

ALTER TABLE Student ADD NrLeg_sot varchar(5)

INSERT INTO Student
VALUES ('117', 'Popescu', 'F', 'Florentina', 'f',
CONVERT(smalldatetime,'15/04/1978',103), 'C', '223','102')

INSERT INTO Student
VALUES ('118', 'Ionescu', 'I', 'AlexANDra', 'f', CONVERT (smalldatetime, '25/06/1976',
103), 'C', '223','106')

SELECT *

FROM Student

UPDATE Student
SET NrLeg_sot='117'
WHERE NrLeg='102'

UPDATE Student
SET NrLeg_sot='118'
WHERE NrLeg='106'

SELECT *
FROM Student S1, Student S2
WHERE S1.NrLeg=S2.NrLeg_sot

SELECT *
FROM Student S1 **INNER JOIN** Student S2
ON S1.NrLeg=S2.NrLeg_sot

SELECT *
FROM Student S1, Student S2
WHERE S1.NrLeg=S2.NrLeg_sot **AND** S1.sex='M'

SELECT *
FROM Student S1 **INNER JOIN** Student S2
ON S1.NrLeg=S2.NrLeg_sot **AND** S1.sex='M'

SELECT *
FROM Student S1, Student S2
WHERE S1.NrLeg=S2.NrLeg_sot
AND S1.Data_nastere>S2.Data_nastere

SELECT *
FROM Student S1, Student S2
WHERE S1.NrLeg=S2.NrLeg_sot
AND Datediff(year,S1.Data_nastere, getdate())>Datediff(year,S2.Data_nastere, getdate())

SELECT *
FROM Student S1, Student S2
WHERE Datediff(year,S1.Data_nastere, getdate())>Datediff(year,S2.Data_nastere, getdate())
AND S1.NrLeg<S2.NrLeg

Semiunirea (Semijoin)

SELECT S.*
FROM Student S, Catalog C
WHERE S.NrLeg=C.NrLeg

SELECT S.*
FROM Student S **INNER JOIN** Catalog C
ON S.NrLeg=C.NrLeg

Non-echiunire (non-Equijoin)

SELECT *
FROM Student S, Catalog C

WHERE S.NrLeg>C.NrLeg

SELECT *

FROM Student S **INNER JOIN** Catalog C

ON S.NrLeg>C.NrLeg