



UNIVERSITETI AAB
FAKULTETI SHKENCA KOMPJUTERIKE
Inxhinieria Softuerike

Querit më të avancuar

- Përdorimi i JOIN për të lidhur tabelat
 - Lidhja e dy tabelave
 - Lidhja e shumë tabelave
 - Lidhja e tabelave me vetvete
- Kuptimi i LEFT dhe RIGHT join
- Përdorimi i nën (sub) querit

Hyrje

- Bashkimi i të dhënave është një nga pikat e forta dhe më të rëndësishme të bazës së të dhënave relacionale.
- Një lidhje në queri kombinon rreshtat nga dy ose më shumë tabela.
- Lidhja lejojnë përdoruesit e bazës së të dhënave për të kombinuar të dhënat nga një tabelë me të dhëna me një ose më shumë tabela të tjera, për aq kohë sa ato janë në relacion.

SQL – kthen të dhëna nga dy ose më shumë tabela

SQL ofron dy mënyra për kthimin e të dhënave prej lidhjeve të tabelave:

- **Join** - Ku dy ose me shumë tabela lidhen nga fushat përkatëse.
- **Subqueries** – Ku në klauzulen Select, From, Where përdorim nenqueri tjerë që lidhen me fushat përkatëse me tabelën kryesore.

Lidhjet (JOIN)

- Lidhja inner join (e quajtur nganjëherë një lidhje e thjeshtë) është një lidhje e dy ose më shumë tabelave që kthehen vetëm ato rreshta që plotësojnë kushtin për bashkim.
- Kthen ato rekorde **vetëm** nëse egjtojnë në të dy tabelat
- Kushtet në lidhje janë përdorur zakonisht për të kufizuar kombinimet e të dhënave në tabela vetëm me rreshtat që përputhen me tabelën tjetër.

Lidhjet (JOIN)

- Lidhja (ang. JOIN) paraqet operacionin i cili sipas kriterit të caktuar lidh atributet nga dy relacione (tabela) dhe për rezultat jep një relacion të ri (relacion virtual) me bashkësi të attributeve nënbashkësi nga unioni i bashkësive të attributeve
- Operacioni JOIN mund të paraqitet si
 - i brendëshëm INNER JOIN dhe
 - si i jashtëm OUTER JOIN.

SQL - Joins

- Inner Join – paraqiten rekorde nga dy tabelat që janë përzgjedhur vetëm kur të dhënat kanë të njëjtën vlerë nga fushat e përbashkët që lidhen tabelat.
- Outer Join - një lidhje që bashkon dy tabela që kthen të gjitha të dhënat nga një tabele dhe nga tabela e dytë vetëm ato të dhënat në të cilat ekziston një vlerë që përputhet me fushën në të cilën tabelat janë të bashkuar.

INNER JOIN

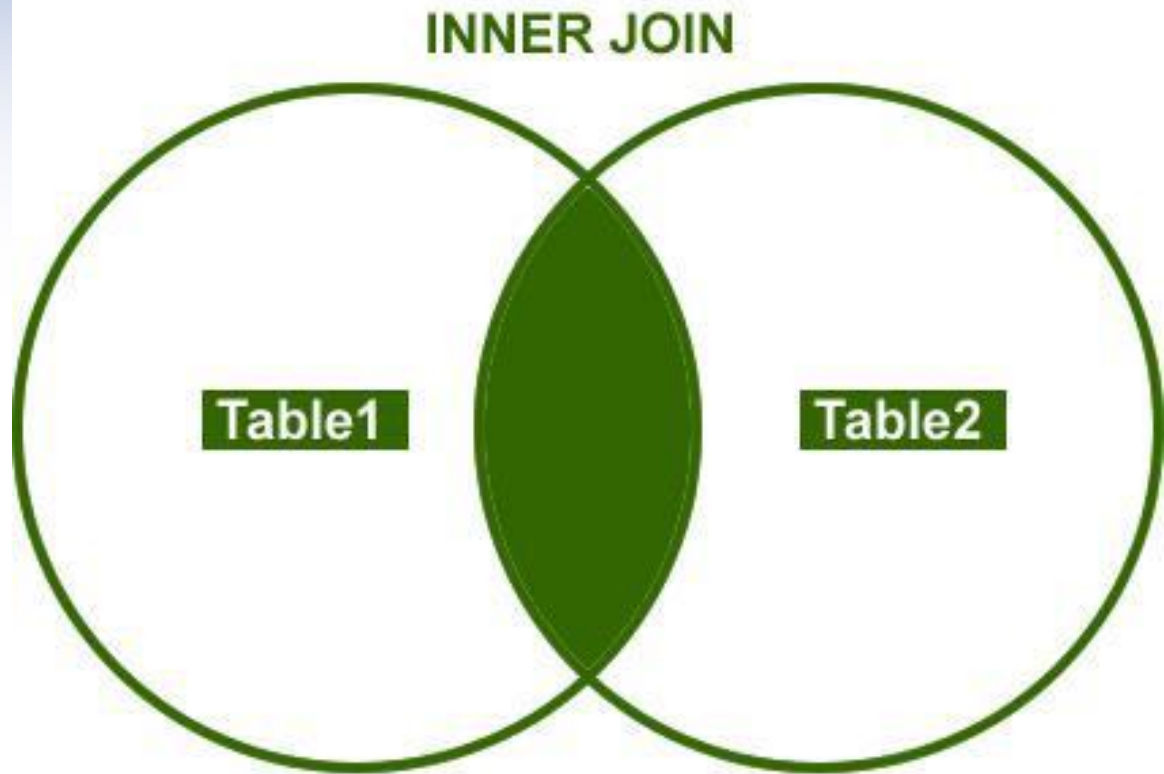
- Operacioni INNER JOIN ka formën e përgjithshme
 - Relacioni1 INNER JOIN Relacioni2 ON condition
- ku Relacioni1, Relacioni2 janë relacione dhe condition kushti i cili determinon formimin e rezultatit.
- Rezultati i komandës përbëhet nga n-shet e kombinuara prej n-sheve të Relacioni1 dhe Relacioni2 për n-shet të cilat plotësojnë kushtin condition

INNER JOIN

```
SELECT * FROM  
  A INNER JOIN B  
  ON <condition>
```

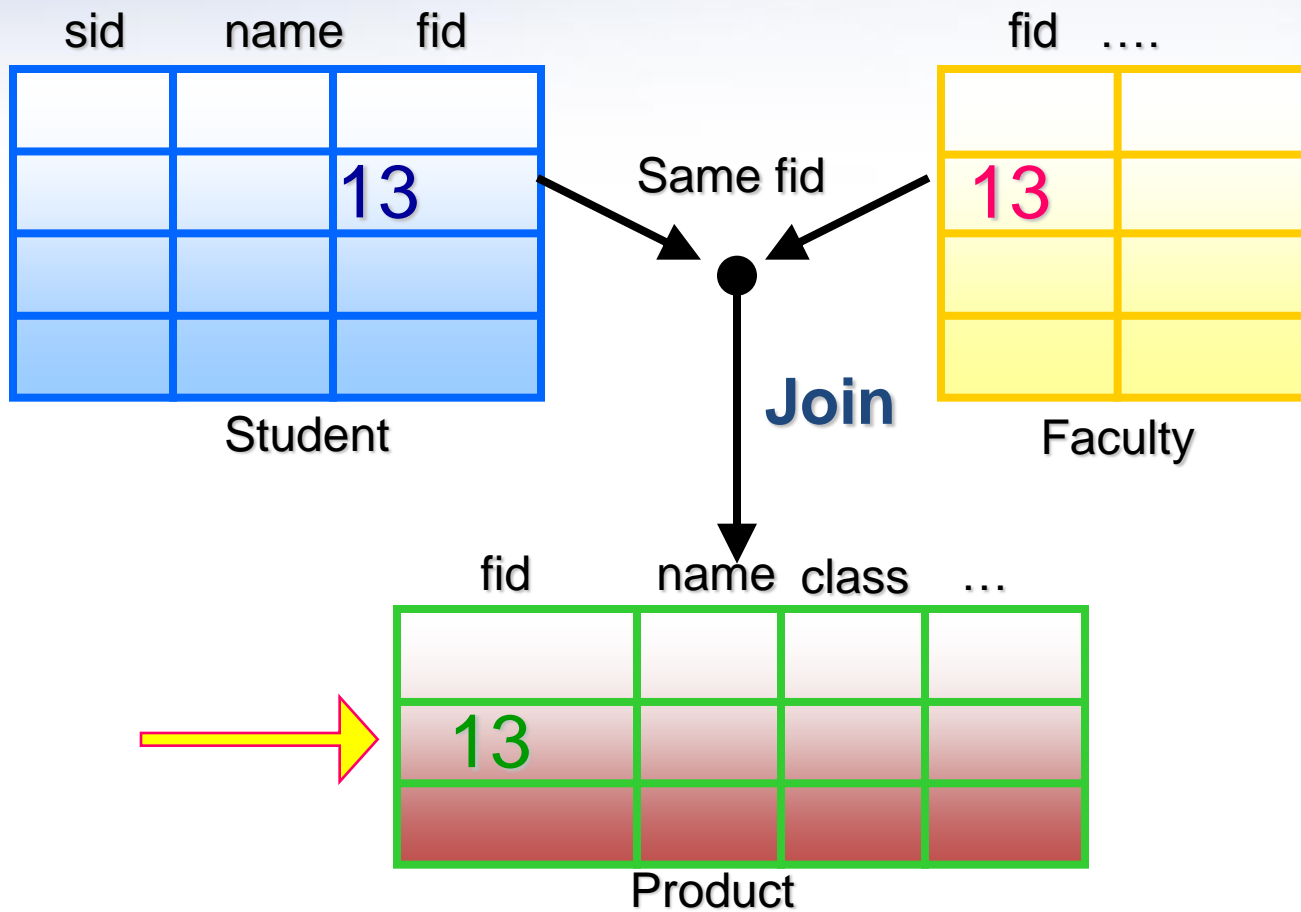
- Eshte e njejta

```
SELECT * FROM A, B  
  WHERE <condition>
```



```
SELECT *  
FROM Table1 t1  
INNER JOIN Table2 t2  
  ON t1.Col1 = t2.Col1
```

INNER JOIN



Shembulli

STUDENT

| S_ID | S_LAST | S_FIRST | F_ID |
|---------------|---------------|---------------|---------------|
| <i>Number</i> | <i>String</i> | <i>String</i> | <i>Number</i> |
| 1 | Miller | Sarah | 1 |
| 2 | Umato | Brian | 1 |
| 3 | Black | Daniel | 1 |
| 4 | Mobley | Amanda | 2 |
| 5 | Sanchez | Ruben | 4 |
| 6 | Connoly | Michael | 3 |

Select *
From Student inner join
Faculty on
Student.FID = Faculty.FID

Shared key values

FACULTY

| F_ID | F_LAST |
|---------------|---------------|
| <i>Number</i> | <i>String</i> |
| 1 | Cox |
| 2 | Blanchard |
| 3 | Williams |
| 4 | Sheng |
| 5 | Brown |

| S_id | S_Last | S_First | F_ID | F_ID | F_Last |
|---------------|---------------|---------------|---------------|---------------|---------------|
| <i>Number</i> | <i>String</i> | <i>String</i> | <i>Number</i> | <i>Number</i> | <i>String</i> |
| 1 | Miller | Saraha | 1 | 1 | Cox |
| 2 | Umato | Brian | 1 | 1 | Cox |
| 3 | Black | Daniel | 1 | 1 | Cox |
| 4 | Mobley | Amanda | 2 | 2 | Blanchard |
| 5 | Sanchez | Ruben | 4 | 4 | Sheng |
| 6 | Connoly | Michael | 3 | 3 | Williames |



Shembulli

Shembulli - Të gjendet lista e shifrave të marinarëve dhe shifrave të anijeve të cilët i kanë rezervuar ndonjëher.

Zgjidhja e këtij shembulli do të jepet ashtu që së pari do të bëjmë operacionin e lidhjes në mes të relacionit **Sailors** dhe relacionit **Reserver** sipas atributit shifra e marinarit dhe pastaj bëjmë projektion në shifrën e marinarit dhe shifrën e anijes.

```
SELECT R.sid, R.bid
```

```
FROM Sailors S INNER JOIN Reserves R on S.sid = R.sid
```

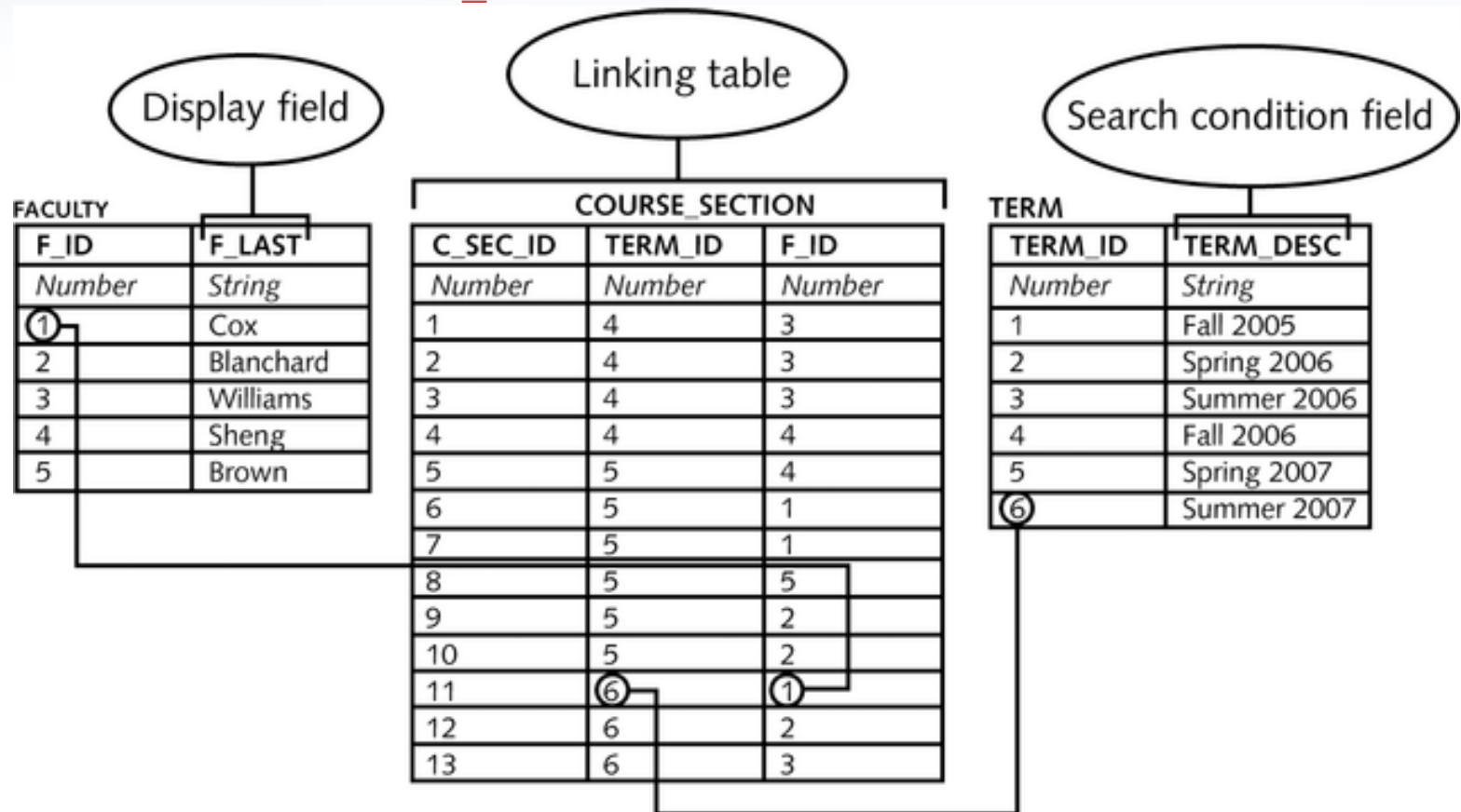
| sid | sname | rating | age |
|-----|---------|--------|-----|
| 2 | Ariana | 7 | 45 |
| 4 | Saranda | 5 | 34 |
| 7 | Vjollca | 7 | 19 |

| sid | bid | day |
|-----|-----|------------|
| 2 | 101 | 01/10/2012 |
| 7 | 103 | 21/12/2012 |

| sid | bid |
|-----|-----|
| 2 | 101 |
| 7 | 103 |

Lidhja e tri tabelave

```
Select * from Faculty inner join Course_Section
on Faculty.FID = Course_Section.FID
inner join Term
on Course_Section.TermID= Term.TermID
```



OUTER JOIN

- Një problem në lidhjen inner join është se vetëm rreshtat që përputhen në mes tabelave mund të paraqiten, mirpo është e mundur që një përdorues mund të jetë i interesuar ti paraqet rreshtat që nuk përputhen me tabelë tjetër.
- Gjetja e rreshtave pa përputhje mund të realizohet me lidhjet **outer join**

OUTER JOIN

- Operacioni OUTER JOIN mund të paraqitet në tre forma, siç janë:
 - i. LEFT OUTER JOIN (ose vetëm LEFT JOIN)
 - ii. RIGHT OUTER JOIN (ose vetëm RIGHT JOIN)
 - iii. FULL OUTER JOIN (ose vetëm FULL JOIN)

dhe forma e përgjithshme e përdorimit

Relacioni1 [LEFT | RIGHT | FULL] OUTER JOIN Relacioni2 ON
condition

Rezultati i komandës përbëhet nga n-shet e kombinuara prej n-sheve të Relacioni1 dhe Relacioni2 për n-shet të cilat plotësojnë kushtin condition dhe i plotësuar me n-shet nga:

- nëse përdoret LEFT - Relacioni1 të cilët nuk kanë pjesë nga Relacioni2 sipas kushtit por ajo pjesë plotësohet me NULL vlera,
- nëse përdoret RIGHT – Relacioni2 të cilët nuk kanë pjesë nga Relacioni1 sipas kushtit por ajo pjesë plotësohet me NULL vlera dhe
- nëse përdoret FULL - Relacioni1 dhe Relacioni2 të cilët nuk kanë pjesë nga relacioni tjetër sipas kushtit por ajo pjesë plotësohet me NULL vlera.

Outer join

| | | |
|--|--|--|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

Natural
Join

| | | |
|--|--|--|
| | | |
| | | |
| | | |

No Match

LEFT

Outer Join

RIGHT

Left outer join



(C) <http://blog.SQLEAuthority.com>

Left outer join

- **Shembulli 2.** Të gjendet lista e shifrave të marinarëve (për të gjithë marinarët) dhe shifrave të anijeve të rezervuara ndonjëherë nga ata.

Zgjidhja është

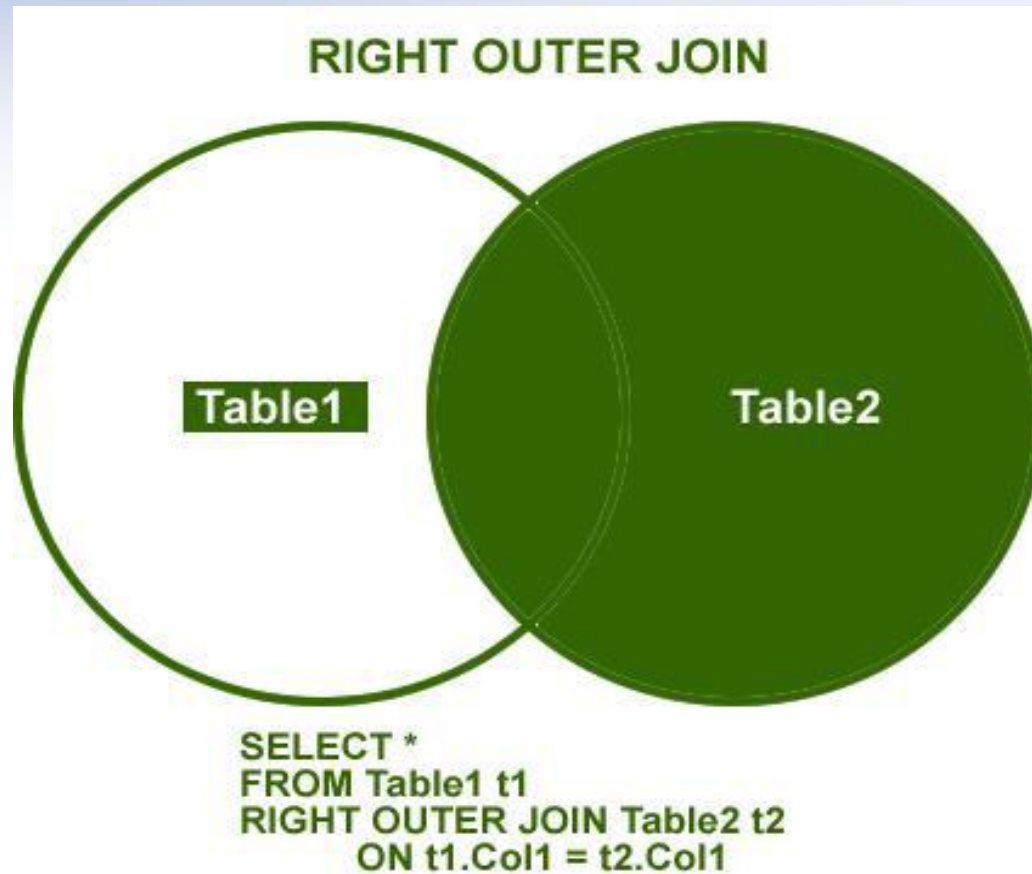
```
SELECT S.sid, R.bid
```

```
FROM Sailors S LEFT JOIN Reserves R on S.sid=R.sid
```

| sid | bid |
|-----|------|
| 22 | 1 |
| 4 | Null |
| 15 | 4 |

- Në rezultatin e fituar vërejmë në rreshtin e tretë kemi NULL vlerë.
- Kështu komanda më lartë i merr të gjitha n-shet nga relacioni Sailors (nga ana e majt) ndërsa nga relacioni Reserves (nga ana e djatht) i merr vetëm n-shet sipas kushtit S.sid=R.sid.
- NULL vlera në rreshtin e tretë do të thotë se marinari me shifër 4 nuk ka rezervuar asnjëherë anije.

Right outer join



(C) <http://blog.SQLAuthority.com>

Right outer joinn

- Të gjendet lista e shifrave të marinarëve dhe shifrave të anijeve të rezervuara ndonjëherë nga ata (për të gjithë anijet).
- Shembulli është i ngjashëm me shembullin paraprakë dhe ilustron operacioni RIGHT OUTER JOIN, zgjidhja është:

SELECT R.sid, B.bid

FROM Reserves R RIGHT outer JOIN Boat B on R.bid = B.bid

| sid | bid |
|------|-----|
| 22 | 3 |
| null | 4 |
| 34 | 12 |

- Në rezultatin e fituar vërejmë në rreshtin e dyte kemi NULL vlerë.
- Kështu komanda më lartë i merr të gjitha n-shet nga relacioni **Boat** (nga ana e djatht) ndërsa nga relacioni **Reserves** (nga ana e majt) i merr vetëm n-shet sipas kushtit **R.bid = B.bid**
- NULL vlera në rreshtin e dyte do të thotë se anija me shifër 4 nuk është rezervuar asnjëherë nga marinarët.

Lidhja CROSS

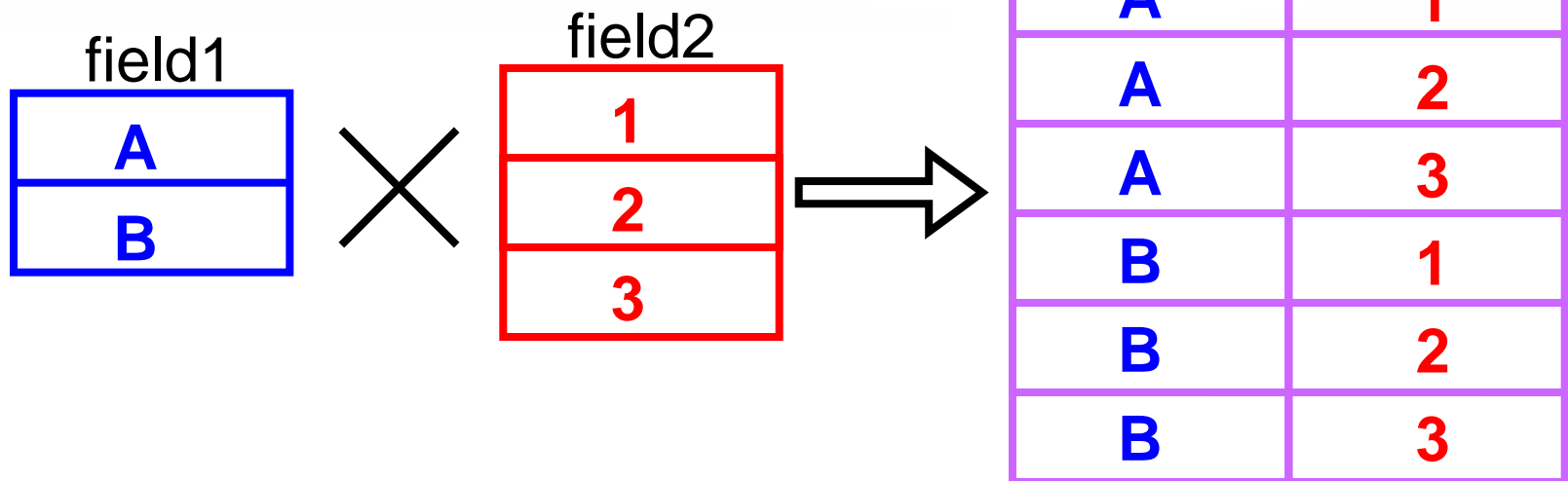
- CROSS JOIN klauzolë prodhon cross-produktin e dy tabelave.
- Kjo është e njëjtë si një produkt Kartezian mes dy tabelave.

```
SELECT * FROM  
    A CROSS JOIN B
```

- Eshte e njejta

```
SELECT * FROM A, B
```

Produkti Kartezian



Lidhja CROSS

Student

| ID | Name |
|-----|------|
| 123 | John |
| 124 | Mary |
| 125 | Mark |
| 126 | Jane |

Enrolment

| ID | Code |
|-----|------|
| 123 | DBS |
| 124 | PRG |
| 124 | DBS |
| 126 | PRG |

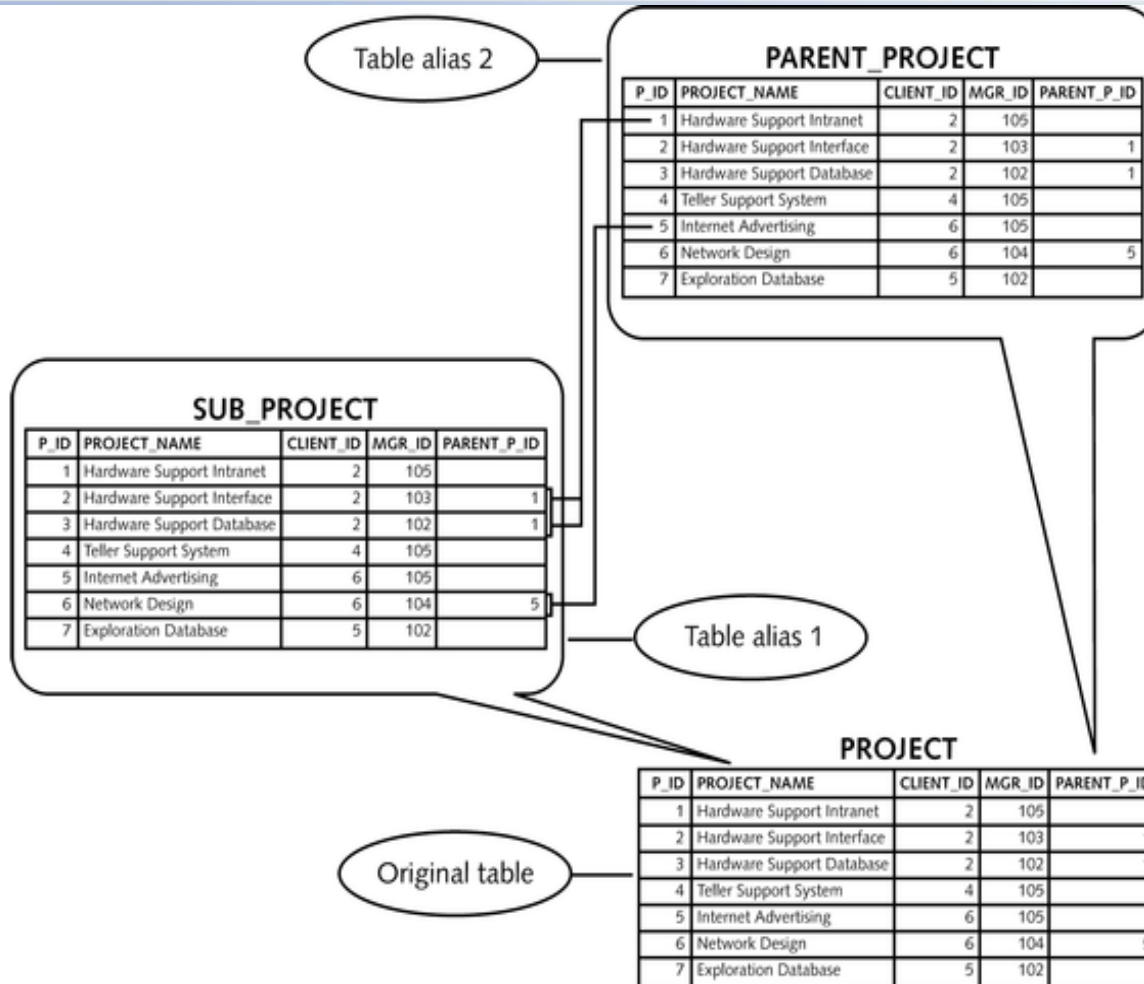
```
SELECT * FROM
Student CROSS JOIN
Enrolment
```

| ID | Name | ID | Code |
|-------|------|-----|------|
| 123 | John | 123 | DBS |
| 124 | Mary | 123 | DBS |
| 125 | Mark | 123 | DBS |
| 126 | Jane | 123 | DBS |
| 123 | John | 124 | PRG |
| 124 | Mary | 124 | PRG |
| 125 | Mark | 124 | PRG |
| 126 | Jane | 124 | PRG |
| 123 | John | 124 | DBS |
| 124 | Mary | 124 | DBS |
| | | | |

Lidhja me vetëveten (Self-Join)

- Në queri lidhet tabelen me vetëveten nëse paraqitet nevoja që ajo tabel te ketë disa role në një relacion,
- Për shembull tabela Puntori mund të lidhet me vete për të gjetur puntorin qe eshte udhëheqës.
- Përdoret kur një tabelë ka një marrëdhënie të huaj të rëndësishëm në vetëvete (zakonisht marrëdhënie prind-fëmijë)
- Përdor pseodenime të ndryshme në mënyrë që të dallohet tabela prind dhe tabela fëmijë

Lidhja me vetë veten (Self-Join)

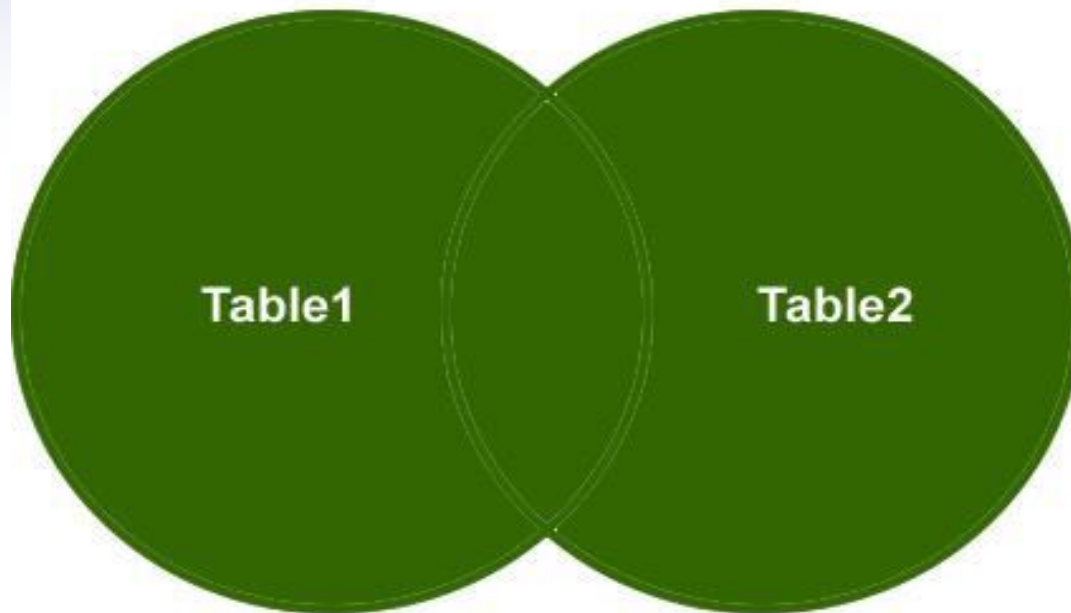


Full joinn

- Në një lidhje nëse duam që të përfshijë rreshtave në të dy tabelat që nuk mund të gjeni ndeshje në tabelat e kundërta?
- Ajo mund të realizohet me komandën full joinn.
- Deri tani, e vetmja mënyrë për të arritur një bashkim të plotë (që vlerat mungojnë në të dy anët e një queri) ishte unioni i rezultateve të dy outerjoin të majtë dhe outerjoin të djathtë.

Full joinn

FULL OUTER JOIN



```
SELECT *  
FROM Table1 t1  
FULL OUTER JOIN Table2 t2  
ON t1.Col1 = t2.Col1
```

Full joinn

- Të paraqiten të gjitha të dhënat nga tabela Departamenti dhe Puntori
- **Select * from Puntori p
full joinn Departamenti d
on p.DID=d.DepID**

| P_id | Emri | Mbiemri | Adres | DID | DID | Pershkrimi |
|------|--------------|---------|-----------|------|------|------------|
| 1122 | Kasanistriot | Gashi | Prishtine | 1 | 1 | DIT |
| 2213 | Anita | Hasani | Gjakove | 2 | 2 | DGJ |
| 6672 | Labinot | Hyseni | Prizeren | Null | Null | Null |
| Null | Null | Null | Null | Null | 3 | DIV |

Lidhje të suksesshëme

- ❑ Planifikoni lidhjen tuaj:
 - Vizatoni një mini-ERD për të treguar se cilat tabelat janë të përfshirë.

- ❑ Të numërohen tabelat që përfshihen në queri.

- ❑ Numri i lidhjeve është gjithmonë një më pak se numri i tabelave në queri.

- ❑ Shiko emrat e kolonave që paraqiten.

?