

# SQL

## *Structured Query Language*

# SQL

- **SELECT** fields                      seleziona i campi (colonne) da visualizzare  
**FROM** table                        specifica la tabella da cui leggere i dati  
**WHERE** condizione                seleziona i record (righe) da visualizzare
  
- **SELECT** expression                è possibile specificare espressioni  
**FROM** table                        (contenenti operazioni e funzioni di calcolo)
  
- **SELECT** table.fields                occorre specificare la tabella di origine  
**FROM** tables                        di ciascun campo  
quando i dati provengono da più tabelle
  
- **SELECT** [field name]                se il nome di un campo o di una tabella è  
**FROM** [table name]                una KEYWORD riservata: va messa tra [ ]  
(o contengono spazi o altri caratteri speciali)

# SQL

## ■ WHERE

- WHERE Code = 2
- WHERE QID = 'Q0004' AND [Date] > #12/1/2005#
- WHERE QID = 'Q0004' OR QID = 'Q0013'
- WHERE Active = True
  
- WHERE QID IN ("Q0003", "Q0004", "Q0011", "Q0044")
  
- WHERE QID LIKE "Q000?"
- WHERE QID LIKE "Q00??"
- WHERE QID LIKE "Q0\*"

## ■ LIKE: metacaratteri (*wildcard*)

- \* zero o più caratteri qualsiasi (\* in MS Access, % in SQL standard)
- ? un singolo carattere qualsiasi (\_ in SQL standard)
- # una singola cifra (0-9)
- [charlist] qualsiasi singolo carattere in *charlist*
- [!charlist] qualsiasi singolo carattere NON in *charlist*

# SQL

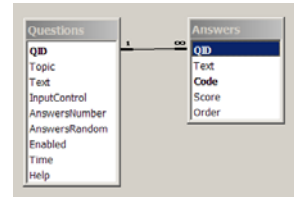
## ■ PARAMETRIZZAZIONE DI UNA QUERY

- Per generalizzare una query (ad es. una ricerca) è possibile utilizzare come operando in una espressione, al posto di un valore fisso, un **PARAMETRO** variabile.
- Esempi:
  - WHERE QID = ???
  - WHERE QID = pippo
  - WHERE QID = [Inserisci il QID da cercare:]
  - WHERE [Text] LIKE [Stringa di ricerca (con metacaratteri):]
- Qualsiasi nome non riconosciuto dall'interprete SQL come nome di campo (o come keyword del linguaggio stesso) viene interpretato come parametro: il valore del parametro viene richiesto interattivamente quando la query viene mandata in esecuzione.

## JOIN in SQL

### ■ JOIN INTERNO (INNER):

```
SELECT Questions.QID, Questions.Text, Answers.Text
FROM Questions INNER JOIN Answers
ON Questions.QID = Answers.QID
```



### ■ JOIN ESTERNO (LEFT/RIGHT):

```
SELECT Questions.QID, Questions.Text, Answers.Text
FROM Questions LEFT JOIN Answers
ON Questions.QID = Answers.QID
```

EQUI-JOIN = PRODOTTO CARTESIANO + SELEZIONE:

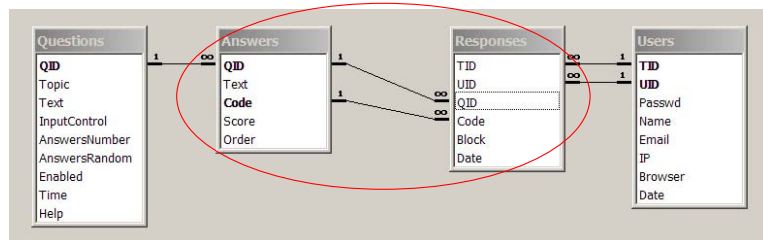
```
SELECT Questions.QID, Questions.Text, Answers.Text
FROM Questions, Answers
WHERE Questions.QID = Answers.QID
```

## SQL

### ■ INNER JOIN:

```
SELECT table1.fields, table2.fields
FROM table1 INNER JOIN table2 ON table1.field=table2.field
```

```
SELECT table1.fields, table2.fields
FROM table1 INNER JOIN table2
ON table1.field1=table2.field1 AND table1.field2=table2.field2
```



### ■ Esempio:

```
SELECT Responses.QID, Answers.Score
FROM Answers INNER JOIN Responses
ON Answers.QID=Responses.QID AND Answers.Code=Responses.Code
```

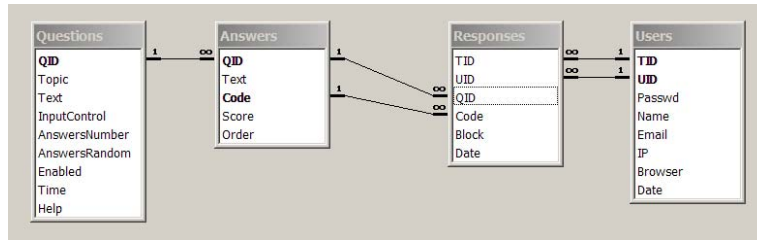
## SQL

- JOIN SEMPLICE: tra due tabelle

```
SELECT table1.fields, table2.fields
FROM table1 INNER JOIN table2 ON table1.field=table2.field
```

- JOIN DOPPIO: tra tre tabelle

```
SELECT table1.fields, table2.fields, table3.fields
FROM table1 INNER JOIN
(table2 INNER JOIN table3 ON table2.field=table3.field)
ON table1.field=table2.field
```



## SQL

- QUERY CON ORDINAMENTO:

```
SELECT fields
FROM tables
ORDER BY fields DESC
```

- SELECT Questions.QID, Questions.Text, Answers.Text, Answers.Score  
FROM Questions INNER JOIN Answers ON Questions.QID=Answers.QID  
ORDER BY Questions.QID
- SELECT Questions.QID, Questions.Text, Answers.Text, Answers.Score  
FROM Questions INNER JOIN Answers ON Questions.QID=Answers.QID  
ORDER BY Questions.QID, Answers.Code

## SQL

- QUERY DI AGGREGAZIONE:

```
SELECT fields, funzione_aggregazione(field)
FROM tables
GROUP BY fields
```

- SELECT Questions.QID, COUNT(Answers.QID) AS NR  
FROM Questions INNER JOIN Answers ON Questions.QID=Answers.QID  
GROUP BY Questions.QID

- SELECT Questions.QID, FIRST(Questions.Text) AS T, COUNT(\*) AS N  
FROM Questions INNER JOIN Answers ON Questions.QID=Answers.QID  
GROUP BY Questions.QID

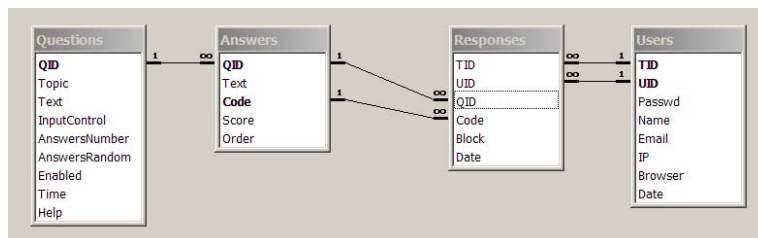
- FUNZIONI DI AGGREGAZIONE:

COUNT, SUM, AVG, STDEVP, FIRST, LAST, MAX, MIN, ...

## SQL

- Esempio: Correzione dei compiti

```
SELECT UID, TID, COUNT(Responses.QID) AS N, SUM(Answers.Score) AS V
FROM Responses INNER JOIN Answers
ON Responses.QID=Answers.QID AND Responses.Code=Answers.Code
WHERE Responses.TID = "June2003"
GROUP BY Responses.UID, Responses.TID
ORDER BY UID, TID
```



## SQL

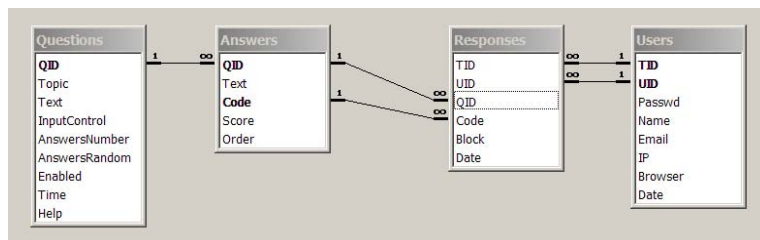
- QUERY NIDIFICATE
- SELECT fields  
FROM (SELECT fields FROM table ...)
- SELECT table.fields, query.fields  
FROM table INNER JOIN (SELECT ...) AS query  
ON table.field = query.field
- SELECT fields  
FROM table  
WHERE field IN (SELECT ...)

## SQL

- Esempio: Correzione dei compiti (aggiungiamo i nomi degli studenti)
 

```

SELECT Users.Name, Results.*
FROM Users LEFT JOIN
(SELECT UID, TID, COUNT(Responses.QID) AS N, SUM(Answers.Score) AS V
FROM Responses INNER JOIN Answers
ON Responses.QID=Answers.QID AND Responses.Code=Answers.Code
GROUP BY Responses.UID, Responses.TID) AS Results
ON Results.UID=Users.UID AND Results.TID=Users.TID
ORDER BY Users.Name
```

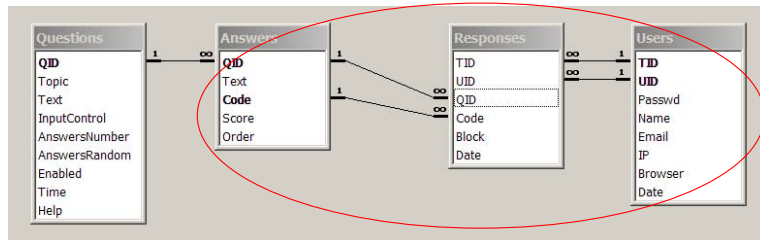


## SQL

- Esempio: Correzione dei compiti (soluzione alternativa)

```

SELECT  FIRST(Users.Name), Users.QID, Users.TID,
        Count(Responses.QID) AS N, SUM(Answers.Score) AS V
FROM    Users LEFT JOIN
        (Answers INNER JOIN Responses
         ON Answers.QID=Responses.QID AND Answers.Code=Responses.Code)
ON      Users.UID=Responses.UID AND Users.TID=Responses.TID
GROUP BY Users.UID, Users.TID
ORDER BY Users.Name
  
```



## SQL

- CROSSTAB QUERY: query di riepilogo a campi incrociati

**TRANSFORM** *funzione\_aggregazione*(field)

```

SELECT  fields
FROM    tables
GROUP BY fields
PIVOT  field
  
```

- **TRANSFORM** Count(Code)  
 SELECT UID, Count(Code)  
 FROM Responses  
 GROUP BY UID  
 ORDER BY UID  
 PIVOT TID
- PIVOT Year([Date])
- PIVOT Format([Date], "yymm")

# SQL

- La documentazione migliore si trova on-line (in inglese)
  
- Microsoft Jet SQL
  - MSDN Library:  
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnacc2k/html/acfundsql.asp>
  - Office:  
<http://office.microsoft.com/en-us/assistance/CH062526881033.aspx>
  
- Molti altri siti, es:
  - Devguru:  
[http://www.devguru.com/Technologies/jetsql/quickref/jet\\_sql\\_list.html](http://www.devguru.com/Technologies/jetsql/quickref/jet_sql_list.html)
  - Wikipedia (italiano):  
<http://it.wikipedia.org/wiki/Sql>