

Utilizzo delle viste

Per rendere più facile la lettura e la comprensione di query molto complesse è utile l'utilizzo delle "VISTE". Vediamo di implementare le viste nella nostra query finale:

Per prima cosa creiamo le tre viste di base: "chiuso", numero_errato" e "inviata_mail", che non fanno altro che effettuare i conteggi delle ricorrenze per gli stati.

```
CREATE VIEW chiuso AS
(
SELECT data, COUNT(*) AS chiusi
FROM utenti_ebay
WHERE stato = 'Chiuso'
GROUP BY data
)
```

```
CREATE VIEW numero_errato AS
(
SELECT data, COUNT(*) AS errati
FROM utenti_ebay
WHERE stato = 'Numero errato'
GROUP BY data
)
```

```
CREATE VIEW inviata_mail AS
(
SELECT data, COUNT(*) AS mail
FROM utenti_ebay
WHERE stato = 'Inviata mail'
GROUP BY data
)
```

Successivamente creiamo altre tre viste relative ai JOIN tra le viste appena create:

```
CREATE VIEW chiuso_join AS
(
SELECT chiuso.data,
       COALESCE(chiusi,0) AS chiusi,
       COALESCE(errati,0) AS errati,
       COALESCE(mail,0) AS mail
FROM chiuso
LEFT OUTER JOIN
numero_errato
ON chiuso.data = numero_errato.data
LEFT OUTER JOIN
inviata_mail
ON chiuso.data = inviata_mail.data
)
```

```

CREATE VIEW numero_errato_join AS
(
  SELECT numero_errato.data,
    COALESCE(chiusi,0) AS chiusi,
    COALESCE(errati,0) AS errati,
    COALESCE(mail,0) AS mail
  FROM chiuso
  RIGHT OUTER JOIN
numero_errato
  ON chiuso.data = numero_errato.data
  LEFT OUTER JOIN
inviata_mail
  ON numero_errato.data = inviata_mail.data
  WHERE chiuso.data IS NULL
)

```

```

CREATE VIEW inviata_mail AS
(
  SELECT inviata_mail.data,
    COALESCE(chiusi,0) AS chiusi,
    COALESCE(errati,0) AS errati,
    COALESCE(mail,0) AS mail
  FROM chiuso
  RIGHT OUTER JOIN
    (numero_errato
  RIGHT OUTER JOIN
    inviata_mail
    ON numero_errato.data = inviata_mail.data)
  ON chiuso.data = inviata_mail.data
  WHERE chiuso.data IS NULL AND numero_errato.data IS NULL
)

```

Create queste tre viste, possiamo ancora crearne una che faccia l'unione di queste tre e le ordini per data:

```

CREATE VIEW union_all AS
(
  (SELECT * FROM chiuso_join)
  UNION ALL
  (SELECT * FROM numero_errato_join)
  UNION ALL
  (SELECT * FROM inviata_mail_join)
  ORDER BY data;
)

```

Fatta questa preparazione su MySQL, da PHP è sufficiente una query che selezioni la vista “union_all” per avere i risultati voluti:

```

$query = "SELECT * FROM union_all";
$res = mysql_query($query);

```