**Web Developer**

**Unità Formativa (UF): Fondamenti di base di dati**

**Docente: Adelchi Valenti**

**Titolo argomento: SQL – Subquery**

**SUBQUERY**

Una subquery è una query nidificata all'interno di un'istruzione SELECT, INSERT, UPDATE o DELETE o all'interno di un'altra subquery. La subquery viene eseguita prima della query.

Una subquery scalare restituisce una sola colonna con una sola riga. Una subquery non scalare restituisce più righe.

**Subquery scalare**

La subquery in questo caso restituisce un solo risultato, ovvero l’incrocio tra una riga ed una colonna della tabella.

SELECT DISTINCT f.genre\_id,

(SELECT g.genre\_name FROM genre g WHERE g.genre\_id = 2) as genre\_name

FROM film f

WHERE f.genre\_id = 2;

**Subquery non scalare**

Le istruzioni che includono una subquery non scalare vengono in genere formulate in uno dei formati seguenti:

* … WHERE expression [NOT] IN (subquery)
* … WHERE expression comparison\_operator [ANY | ALL] (subquery)
* … WHERE [NOT] EXISTS (subquery)

**[NOT] IN**

Se volessi sapere tutti i customer che hanno effettuato noleggi con il rispettivo conteggio dei noleggi stessi:

SELECT r.customer\_id, count(r.customer\_id) AS tot\_rental

FROM rental r

WHERE r.customer\_id **IN** (

SELECT c.customer\_id

FROM customer c

)

GROUP BY r.customer\_id;

**ANY**

Se volessi sapere tutti i customer che hanno effettuato noleggi con il rispettivo conteggio dei noleggi stessi per tutte le occorrenze in cui customer\_id è compreso tra 0 e 100.

SELECT r.customer\_id, count(r.customer\_id) AS tot\_rental

FROM rental r

WHERE r.customer\_id = **ANY** (

SELECT c.customer\_id

FROM customer c WHERE c.customer\_id > 0 AND c.customer\_id < 101

)

GROUP BY r.customer\_id;

**ALL**

Se volessi sapere tutti i customer che hanno effettuato noleggi con il rispettivo conteggio dei noleggi stessi per tutte le occorrenze in cui customer\_id è maggiore di 100.

SELECT r.customer\_id, count(r.customer\_id) AS tot\_rental

FROM rental r

WHERE r.customer\_id > **ALL** (

SELECT c.customer\_id

FROM customer c WHERE c.customer\_id > 0 AND c.customer\_id < 101

)

GROUP BY r.customer\_id;

**EXISTS**

L'operatore EXISTS controlla se la riga della subquery corrisponde a una riga della query esterna. Se non ci sono dati corrispondenti, l'operatore EXISTS restituisce FALSE.

Se volessi sapere tutti i language\_id per i quali esiste almeno un film girato in quella lingua:

SELECT DISTINCT f.language\_id

FROM film f

WHERE EXISTS (

SELECT l.language\_id

FROM language l

);