

# Seconda Prova - soluzioni

## Requisiti Generali / Punti di attenzione

1. I nomi di entità o attributi utilizzati nella modellazione devono essere obbligatoriamente corrispondenti a parole nel testo descrittivo.
2. L'obbligatorietà e la cardinalità delle associazioni deve essere indicata accuratamente.
3. L'eventuale brutta deve essere barrata lungo una diagonale della pagina.

## Il Quadri delle Avanguardie Pittoriche

Il catalogo delle avanguardie pittoriche elenca i Quadri di Pittori ospitati nei Musei di arte moderna di diverse Città. Ogni quadro è il prodotto di un solo pittore, e viene attribuito come Influenzato da diverse Correnti artistiche (decadentismo, dadaismo, cubismo, impressionismo, ...).

Si desidera progettare una base di dati relazionale allo scopo di registrare tutte le informazioni rilevanti. Inoltre la base di dati sarà gestita da una applicazione web utilizzata per il data entry e la produzione di diversi report.

- I titolo e l'anno dei quadri di Picasso [ordinati per anno]

```
SELECT Q.TITOLO, Q.ANNO
FROM QUADRI AS Q, PITTORI AS P
WHERE Q.ID_P=P.ID_P
AND P.NOME='Picasso'
[ORDER BY Q.ANNO]
```

- Quanti sono i quadri Impressionisti a Torino?

```
SELECT COUNT(Q.ID_Q) AS TOT_QUADRI
FROM QUADRI AS Q, CORRENTI AS CA, INFLUENZA AS I, MUSEI AS M, CITTA AS C
WHERE Q.ID_Q=I.ID_Q
AND I.ID_CA=CA.ID_CA
AND Q.ID_M=M.ID_M
AND M.ID_C=C.ID_C
AND CA.NOME='IMPRESSIONISMO'
AND C.NOME='TORINO'
```

- Elenco dei quadri cubisti o impressionisti del museo MOMA

```
SELECT Q.*
FROM QUADRI AS Q, CORRENTI AS CA, INFLUENZA AS I, MUSEI AS M
WHERE I.ID_Q=Q.ID_Q
AND I.ID_CA=CA.ID_CA
AND Q.ID_M=M.ID_M
AND (CA.NOME='IMPRESSIONISMO'
OR CA.NOME='CUBISMO')
AND M.NOME='MOMA'
```

## I musei che espongono Van Gogh

```
SELECT DISTINCT M.*
FROM Museo AS M INNER JOIN Quadro AS Q ON (Q.id_m = M.id_m)
      INNER JOIN Pittori AS P ON (Q.id_p = P.id_p)
WHERE P.nome = 'Van Gogh'
```

## Il Negozio Online

Un negozio online vende diversi Item (articoli) ad Utenti registrati. Ogni Venditore censisce i suoi item che rende disponibili alla vendita, ciascuno con un prezzo ed una quantità.

L'ordine può Comprendere diversi item, ciascuno acquistabile con il prezzo di vendita del momento, in una quantità a piacere purché essa sia disponibile.

L'utente, identificato da un nome, ha un indirizzo email, diverso per ogni utente, ed è autenticato da una password, e viene registrata la data della sua iscrizione.

Si desidera progettare una base di dati relazionale allo scopo di registrare tutte le informazioni rilevanti. Inoltre la base di dati sarà gestita da una applicazione web utilizzata per la registrazione degli utenti e i loro ordini, la consultazione dei cataloghi, il data entry di venditori e item, e la produzione di diversi report.

- Gli ordini di LadyGaga ordinati per data di spedizione

```
SELECT U.NOME, O.*
FROM UTENTI AS U, ORDINI AS O
WHERE U.NOME=O.NOME_U
AND U.NOME='LadyGaga'
ORDER BY O.DATA_SP
```

- Gli item del venditore Scarpe&Ciabatte al momento non disponibili

```
SELECT V.NOME, I.*
FROM ITEM AS I, VENDITORE AS V
WHERE I.ID_V=V.ID_V
AND V.NOME='Scarpe&Ciabatte'
AND I.QTA=0
```

- La spesa totale degli utenti che si sono disiscritti dal sito nel 2019 [ordinati per utente].

```
SELECT U.username, SUM(O.prezzo) AS prezzo_totale
FROM Utente AS U INNER JOIN Ordine AS O
      ON (O.username = U.username)
WHERE U.data_disiscrizione BETWEEN '1/1/2019' AND '31/12/2019'
[ORDER BY U.username, U.data_disiscrizione]
```

- I venditori con l'item più costoso del negozio

```
SELECT V.*, I.nome, I.prezzo
FROM Venditore V INNER JOIN Item I ON (I.id_venditore = V.id_venditore)
WHERE I.prezzo = (SELECT MAX(Item.prezzo) FROM item)
```

## Il Blog

Un blog consente ai suoi Utenti registrati di esprimere Commenti su qualsiasi Pagina del blog. Il commento di pagine del sito è possibile solo dopo che l'utente ha effettuato l'accesso al sistema, in Sessioni utente autenticate, e il sistema registra in quale sessione una pagina viene commentata. L'utente, identificato da un nome, ha un indirizzo email, diverso per ogni utente, ed è autenticato da una password, e viene registrata la data della sua iscrizione. Le pagine sono composte da un titolo libero, modificabile anche successivamente, un testo e altri attributi. Le sessioni possono essere chiuse per logout dell'utente o per inattività, controllata da un timer gestito dal blog.

Si desidera progettare una base di dati relazionale allo scopo di registrare tutte le informazioni rilevanti. Inoltre la base di dati sarà gestita da una applicazione web utilizzata per la registrazione degli utenti, dei loro commenti, e la produzione di diversi report sull'attività degli utenti.

- il numero delle sessioni attualmente aperte

```
SELECT COUNT(S.ID_S) AS TOT_SESSIONI
FROM SESSIONI AS S, UTENTI AS U
WHERE S.CLOSED_BY IS NULL
AND S.ID_U=U.ID_U
```

- La media delle sessioni per utente

```
SELECT AVG(S.ID_S) AS MEDIA_S, U.NOME
FROM SESSIONI AS S, UTENTI AS U
WHERE S.ID_U=U.ID_U
GROUP BY U.ID_U
```

- Il testo dei commenti dell'utente Rossi sulla pagina dal titolo "Joyce's Ulysses", ordinati per data di creazione.

```
SELECT C.TESTO, C.DATA_CREAZIONE
FROM PAGINE AS P, COMMENTI AS C, SESSIONI AS S, UTENTI AS U
WHERE P.ID_P=C.ID_P
AND C.ID_S=S.ID_S
AND S.ID_U=U.ID_U
AND U.NOME='Rossi'
AND P.TITOLO='Joyce' 's Ulysses'
ORDER BY C.DATA_CREAZIONE
```

- I titoli delle pagine commentate da utenti che si sono disiscritti nel 2019

```
SELECT P.TITOLO
FROM PAGINE AS P, COMMENTI AS C, SESSIONI AS S, UTENTI AS U
WHERE P.ID_P=C.ID_P
AND C.ID_S=S.ID_S
AND S.ID_U=U.ID_U
AND U.DATA_DISINSCRIZIONE > '2019/01/01'
AND U.DATA_DISINSCRIZIONE < '2019/12/31'
```

- In quante sessioni è stata commentata la pagina dal titolo "L'infinito leopardiano"?

```
SELECT COUNT(S.ID_S) AS NUM_SESSIONI
```

```
FROM PAGINE AS P, COMMENTI AS C, SESSIONI AS S
WHERE P.ID_P=C.ID_P
AND C.ID_S=S.ID_S
AND P.TITOLO='L''infinito leopardiano'
```

- **Elenco delle pagine con il numero dei commenti [ordinate da quelle più commentate]**

```
SELECT COUNT(C.id) AS n_comm, P.titolo -- i commenti per pagina
FROM Commento C INNER JOIN Pagina P ON (C.id_p = P.id_p)
GROUP BY P.id_p
[ORDER BY n_comm DESC]
```