

SQL - QUERY

1. Trovare i dati di tutti i passeggeri con la fedina penale sporca.

```
SELECT Passeggero.*  
  
FROM Passeggero  
  
WHERE Passeggero.Fedina_penale='Sporca';
```

2. Trovare gli identificatori dei fornitori di carburante con quantità maggiore di 100.

```
SELECT IVA  
  
FROM Fornitore  
  
WHERE Tipo_fornitura='Carburante' AND Quantita>100;
```

3. Trovare i voli di aerei che hanno eseguito, almeno una volta, una tratta siberiana.

```
SELECT *  
  
FROM Volo V  
  
WHERE V.Numero NOT IN (  
  
                SELECT Numero  
  
                FROM Volo  
  
                WHERE Tratta<>'Siberiana');
```

4. Trovare il codice bagaglio per quelle prenotazioni, con la compagnia aerea Rayaner e Lufthansa, che presentano imbarco prioritario.

```
SELECT B.PIR  
  
FROM Prenotazione P INNER JOIN Bagaglio B ON P.PNR=B.PNR_prenotazione  
  
WHERE (P.Compagnia_aerea='Rayanair' OR P.Compagnia_aerea='Lufthansa') AND  
  
P.Imbarco_prioritario='SI';
```

5. Trovare i passeggeri che fanno scalo, diretti all'aeroporto di Dublino, raggruppati per codice dell'aereo.

```
SELECT P.*  
  
FROM Passeggero P INNER JOIN Scalo S ON P.N_sala_attesa_scalo=S.N_sala_attesa  
  
WHERE S.Aero_destinazione='Dublino';
```

6. Trovare gli identificatori dei dipendenti, nati dopo il 1970, che hanno effettuato non più di 4 corsi di aggiornamento, con presenza obbligatoria.

```
CREATE VIEW Dipendenti_70 AS (  
  
SELECT P.Codice_staff
```

```

FROM Personale P
WHERE P.Data_nascita>TO_DATE('01-01-1970', 'DD-MM-YYYY')
);

```

```

SELECT D_70.Codice_staff
FROM (PROVA.Dipendenti_70 D_70 INNER JOIN Frequenza F ON
D_70.Codice_staff=F.Codice_staff_personale) INNER JOIN Corso_agg C ON
F.Codice_istruzione_corso_agg=C.Codice_istruzione
WHERE C.Presenza='SI'
GROUP BY D_70.Codice_staff
HAVING COUNT(D_70.Codice_staff)<=4;

```

7. Trovare gli aeroporti che hanno emesso una ed una sola volta un comunicato di “potenziale attacco terroristico”.

```

CREATE VIEW Due_attacchi AS(
SELECT DISTINCT C1.Nome_aeroporto_mandante
FROM Comunicato C1 INNER JOIN Comunicato C2 ON C1.ID<>C2.ID AND
C1.Nome_aeroporto_mandante=C2.Nome_aeroporto_mandante
WHERE C1.Tipologia='Emergenza' AND C2.Tipologia='Emergenza');

```

```

CREATE VIEW Nessuno AS(
SELECT DISTINCT C.Nome_Aeroporto_Mandante
FROM Comunicato C
WHERE C. Nome_Aeroporto_Mandante NOT IN (SELECT DISTINCT C1. Nome_Aeroporto_Mandante
FROM Comunicato C1
WHERE C1.Tipologia='Emergenza'));

```

```

SELECT DISTINCT Nome_aeroporto_mandante
FROM Comunicato
WHERE Nome_aeroporto_mandante NOT IN (SELECT Nome_aeroporto_mandante FROM
Due_attacchi) AND Nome_aeroporto_mandante NOT IN (SELECT
Nome_aeroporto_mandante FROM Nessuno);

```

8. Per i servizi aggiuntivi che pagano un affitto del suolo minore di 3200, trovare il nome del servizio ed il responsabile.

```
SELECT S.nome_servizio, C.Responsabile_add
```

```
FROM Servizio_add S INNER JOIN Contratto_ente C ON C.IVA_add=S.IVA
```

```
WHERE C.Affitto<=3200;
```

9. Trovare il numero totale di piloti, di grado "tre strisce", assunti dalla compagnia aerea Turkish Airline.

```
SELECT COUNT(D.Codice_staff_personale) AS Piloti_veterani
```

```
FROM (Di_aria D INNER JOIN Personale P ON D.Codice_staff_personale=P.Codice_staff) INNER JOIN Compagnia C ON D.Responsabile_compagnia=C.Responsabile AND D.Nome_compagnia=C.Nome
```

```
WHERE P.Ruolo='Pilota' AND D.Grado='Tre strisce' AND D.Nome_compagnia='Turkish Airline';
```

10. Visualizzare le informazioni delle città raggiunte, da ogni mezzo su gomma, della società Flixbus.

```
SELECT C.*
```

```
FROM (Gomma G INNER JOIN Collegamento_G CG ON G.Targa=CG.Targa_gomma) INNER JOIN Città C ON C.Nome=CG.Nome_città AND C.Cap=CG.Cap_città
```

```
WHERE G.Società='Flixbus';
```

11. Visualizzare i voli a tratta Siberiana, per l'aeroporto di Manchester, ed eliminarli perché non più erogabili.

```
CREATE VIEW Volo_Sibe_Manchester AS(
```

```
SELECT V.Numero
```

```
FROM Volo V
```

```
WHERE V.Tratta='Siberiana' AND V.Aero_destinazione='Manchester');
```

```
DROP VIEW Volo_Sibe_Manchester;
```

12. Visualizzare nome e cognome dei passeggeri che non hanno mai subito una perquisizione.

```
SELECT Passeggero.Nome, Passeggero.Cognome
```

```
FROM Passeggero
```

```
WHERE Passeggero.CI NOT IN (
```

```
SELECT P.CI
```

```
FROM Passeggero P INNER JOIN Check_in C ON P.Postazione_check_in=C.Postazione
```

```
WHERE C.Perquisizione='ACTION');
```

13. Tipologia di servizio aggiuntivo e nome dei servizi che hanno ottenuto almeno due volte il rinnovo, per il periodo invernale, da almeno un aeroporto.

```
SELECT S.Nome_servizio, S.Tipo_servizio
```

```
FROM Servizio_add S
```

```
WHERE S.IVA IN (
```

```

SELECT C1.IVA_add
FROM Contratto_ente C1 INNER JOIN Contratto_ente C2 ON
        C1.Codice_contratto<>C2.Codice_contratto AND
        C1.IVA_add=C2.IVA_add AND C1.Periodo=C2.Periodo
WHERE C1.Periodo='Invernale'

```

);

14. Nome e cognome dei dipendenti che non hanno mai partecipato a corsi di aggiornamento, erogati dall'Ente "Imparando insieme".

```

SELECT Personale.Nome, Personale.Cognome
FROM Personale
WHERE Personale.Codice_staff NOT IN (
        SELECT P.Codice_staff
        FROM (Personale P INNER JOIN Frequenza F ON
                P.Codice_staff=F.Codice_staff_personale)
                INNER JOIN Corso_agg C ON
                F.Codice_istruzione_corso_agg=C.Codice_istruzione
        WHERE C.Ente='Ente_4'

```

);

15. Trovare il nome dell'aeroporto dove è stato prenotato il biglietto aereo (distinto) più costoso.

```

SELECT A.Nome
FROM (Aeroporto A INNER JOIN Terminal T ON T.Nome_aeroporto=A.Nome) INNER JOIN Passeggero P
ON T.Codice=P.Codice_terminal
WHERE P.PNR_prenotazione IN (
        SELECT Prenotazione.PNR
        FROM Prenotazione
        WHERE Prenotazione.Prezzo IN (
                SELECT MAX(DISTINCT P1.Prezzo)
                FROM Prenotazione P1 INNER JOIN Prenotazione P2 ON
                        P1.PNR<>P2.PNR
                WHERE P1.Prezzo>P2.Prezzo
        ));

```

16. Inserimento di un nuovo bagaglio nella stiva dell'aereo di targa I-BIMD

```
INSERT INTO Bagaglio (PIR, PNR_prenotazione, Targa_aereo, Assicurazione, Peso, Altezza, Larghezza, Profondita, Tipo, Capacita)
```

```
VALUES ('MKO789QWE4', 'P62KFM', 'I-ABCD', 'SI', '11', '5', '10', '10', 'Oggetto', '162254');
```

17. Inserimento di un nuovo contratto per un nuovo dipendente.

```
CREATE VIEW Nuovo_dipendente AS (
```

```
SELECT *
```

```
FROM Personale P INNER JOIN Contratto C ON P.FCodice_contratto=C.Codice_contratto_dipendente
```

```
WHERE TRUE);
```

```
INSERT INTO Nuovo_dipendente
```

```
VALUES ('CP001', 'Pepper', 'Pops', 'PPRPPS74L401W', 'Sicurezza', '12/23/1974', 'Bergamo', 'CCD11', 'inglese, sloveno', 25, 10, 1300, 'SOSPESO', 'SI');
```

18. Aggiornamento dati di un Check-in.

```
UPDATE Check_in
```

```
SET N_agenti='3'
```

```
WHERE Postazione='4';
```

19. Aggiornamento dati di una compagnia aerea.

```
UPDATE Compagnia
```

```
SET N_aerei_aero='2', Sportello='6', Responsabile='Battinozzi'
```

```
WHERE Nome='Levato';
```

20. Eliminazione dati di una città.

```
DELETE FROM Citta
```

```
WHERE Nome='Torino';
```