

Progettazione database Compagnia Aerea *Air Italy*

Lorenzo Comis X81000848

Introduzione

Si vuole progettare il sistema informativo di una piccola compagnia aerea di linea. Gli elementi principali di ogni compagnia aerea sono i voli tra i vari aeroporti serviti, gli aeromobili posseduti, i piloti che compongono la crew e ovviamente i clienti viaggiatori che prenotano i viaggi offerti dalla compagnia.

Analisi delle specifiche

Ogni volo di linea è normalmente caratterizzato da un codice del volo (es. V1234), dall'aeroporto di partenza, quello di arrivo, dalla distanza e dalla durata del viaggio.

I vari viaggi vengono effettuati con un aeromobile e sono caratterizzati dalla data in cui sono o saranno effettuati e quindi da uno stato che indica se il velivolo è attualmente in volo oppure no.

I viaggiatori hanno la possibilità di prenotare differenti viaggi. Di ogni cliente si ha informazione del codice fiscale, nome, cognome, data di nascita e numero di cellulare.

Si ha inoltre un elenco degli aeroporti serviti dalla compagnia. Essi sono identificati da un codice di tre lettere (es. MXP o CTA), dalla città e dalla nazione in cui esso è collocato. Un aeroporto può essere inoltre considerato un hub per la compagnia oppure no.

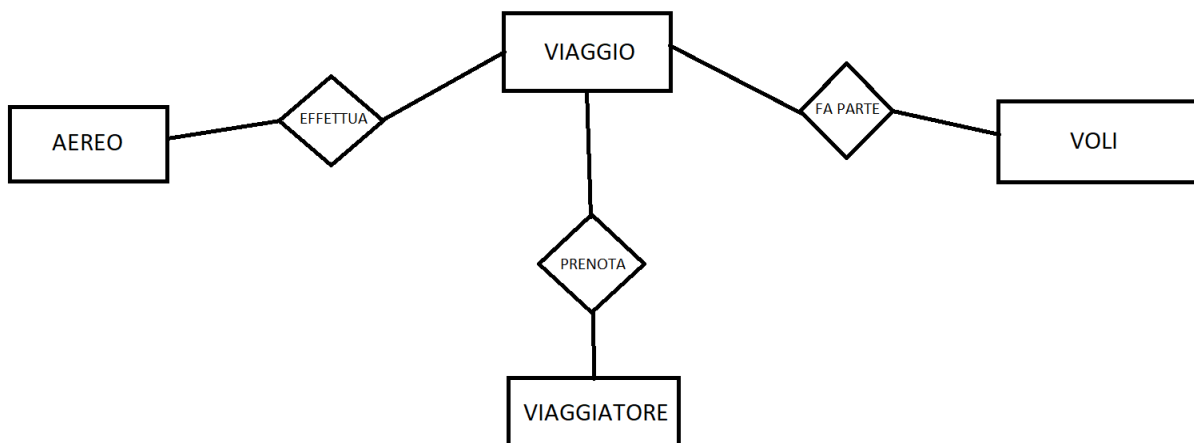
I velivoli che possiede la compagnia sono caratterizzati univocamente dalla matricola, da una marca, modello e anno di acquisto.

I piloti che lavorano all'interno della compagnia sono identificati da un numero di matricola interno, dal nome e cognome e dalla data di nascita.

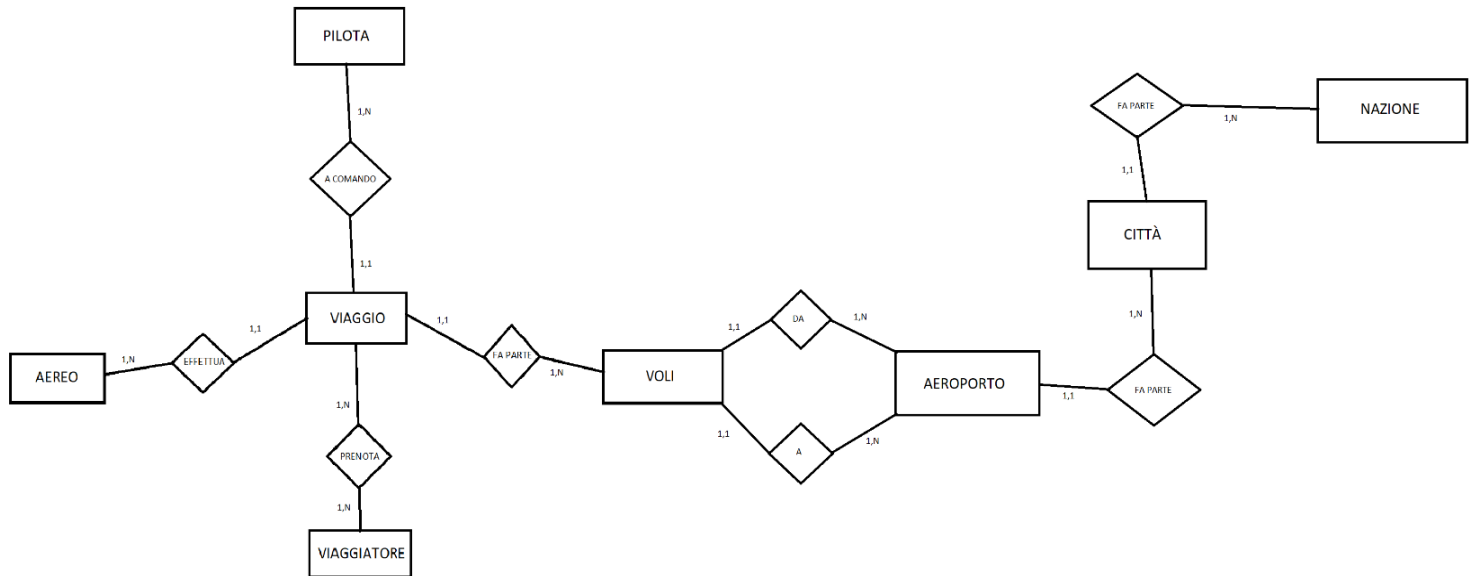
Dizionario dei termini

Termine	Descrizione	Sinonimi	Collegamenti
Volo	Tratta che offre la compagnia		Aeroporto (di partenza e arrivo)
Viaggio	I viaggi effettuati o da effettuare prenotabili dai viaggiatori		Viaggiatore, volo, aereo, pilota
Aeroporto	Aeroporti serviti dalla compagnia, rappresentano i punti di partenza e arrivo dei voli		Voli
Aereo	Velivoli appartenenti alla compagnia utilizzati per i viaggi	Velivolo, mezzo	Viaggio
Pilota	Conducente dell'aeromobile		Viaggio
Viaggiatore	Cliente della compagnia che prenota i viaggi	Cliente	Viaggio

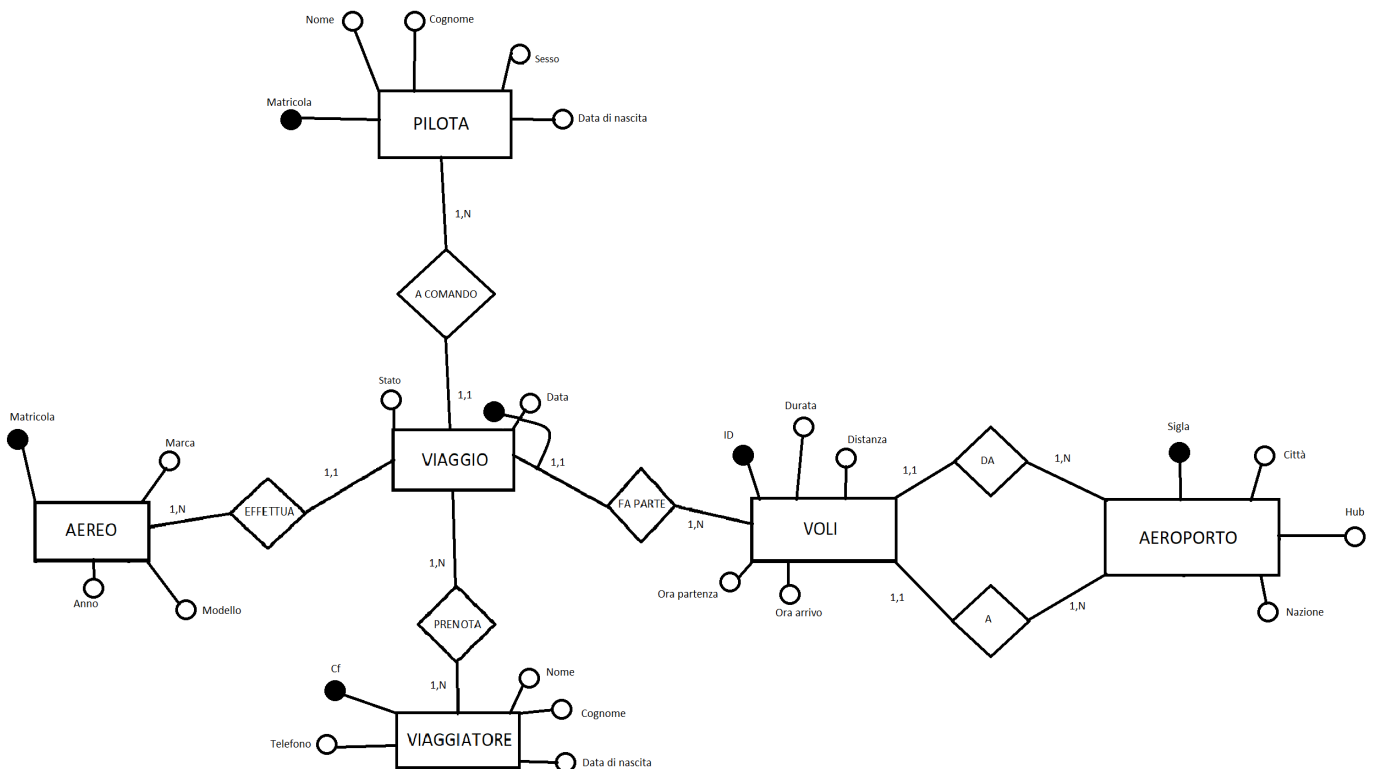
Schema scheletro



Schema intermedio



Schema finale



Dizionario dei dati

Entità

Nome entità	Attributi	Chiave primaria
Voli	ID, durata, distanza, ora partenza, ora arrivo	ID
Viaggio	ID (voli), Data, stato	ID (voli), data
Aeroporto	Sigla, città, hub, nazione	Sigla
Aereo	Matricola, marca, modello, anno	Matricola
Pilota	Matricola, nome, cognome, data di nascita	Matricola
Viaggiatore	Cf, nome, cognome, data di nascita, telefono	Cf

Relazioni

Nome relazione	Entità coinvolte	Descrizione
Effettua	Aereo, viaggio	Associa un viaggio all'aereo che lo effettua. Un aereo può partecipare a più viaggi (ma non nello stesso giorno)
A comando	Pilota, viaggio	Associa un viaggio al pilota che ne è a comando. Un pilota può essere a comando di diversi viaggi (ma non nello stesso giorno)
Prenota	Viaggiatore, viaggio	Associa il viaggiatore alla prenotazione di un viaggio (futuro o passato)
Fa parte	Voli, viaggio	Associa un volo ai vari viaggi che vengono organizzati ognuno in date differenti
Da	Voli, aeroporto	Indica da quale aeroporto parte un volo
A	Voli, aeroporto	Indica in quale aeroporto sarà destinato quel volo

Tabella dei volumi

Concetto	Tipo	Volume
Volo	E	20
Viaggio	E	50
Aeroporto	E	25
Aereo	E	30
Pilota	E	30
Viaggiatore	E	50
Prenotazione	I	20

Tabella operazioni

Operazione	Descrizione	Frequenza	Tipo
O1	Iscrizione di un nuovo viaggiatore nel database	10/giorno	I
O2	Prenotazione di un viaggio di un volo esistente	100/giorno	I
O3	Apertura di una nuova tratta (nuovo volo)	2/giorno	I
O4	Preparare una relazione dei viaggi condotti da ogni pilota.	1/mese	B
O5	Prepara una relazione delle prenotazioni effettuate da ogni cliente.	1/mese	B

Analisi delle ridondanze

Dopo una prima analisi evidenziamo la presenza dell'attributo derivabile "durata" nella tabella voli. Esso potrebbe essere ricavabile calcolando semplicemente la differenza tra l'orario di partenza e l'orario di arrivo del volo utilizzando la funzione TIMEDIFF. Nel nostro caso decidiamo di mantenere l'attributo derivabile "durata" al fine di ottenere un accesso più rapido al dato pagando però uno spazio di memoria aggiuntivo.

Progettazione Logica

Procediamo dopo quanto detto alla progettazione logica della nostra base di dati.

All'interno della progettazione logica le chiavi primarie vengono indicate in **grassetto** mentre quelle esterne sono state sottolineate.

VOLO (**ID**, Durata, Distanza, OraPartenza, OraArrivo, AeroportoPartenza, AeroportoArrivo)

VIAGGIO (**IDVolo**, **DataV**, Stato, Aeromobile, Pilota)

AEROPORTO (**Sigla**, Città, Hub, Nazione)

AEREO (**Matricola**, Marca, Modello, Anno)

PILOTA (**Matricola**, Nome, Cognome, DataNascita)

VIAGGIATORE (**Cf**, Nome, Cognome, DataNascita, Telefono)

PRENOTAZIONE (**IDPrenotazione**, CfViaggiatore, IDVolo, DataV)

Progettazione Fisica

Per la progettazione fisica è stato utilizzato il software MySQL Workbench.

In allegato alla relazione vi sono altri tre file:

- Esame.sql che contiene il codice per la creazione del database;
- Inserimenti.txt che contiene il codice delle query per popolare il database con i dati;
- Query.txt che contiene il codice per alcune operazioni da effettuare sul database.

In basso vi sono riportati degli screenshot riguardanti il codice per la creazione delle tabelle, dell'inserimento dei dati e alcune operazioni sul database.

1. Query per la creazione della tabella "aeroporto"

```
CREATE TABLE IF NOT EXISTS aeroporto (  
    sigla char(3) NOT NULL,  
    città varchar(20) NOT NULL,  
    hub boolean NOT NULL,  
    nazione varchar(20) NOT NULL,  
    PRIMARY KEY (sigla)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

2. Query per la creazione della tabella “volo”. Vi sono due chiavi esterne che fanno riferimento alla chiave primaria della tabella “aeroporto”

```
CREATE TABLE IF NOT EXISTS volo (  
    id char(5) NOT NULL,  
    durata time NOT NULL,  
    distanza int NOT NULL,  
    oraPartenza time NOT NULL,  
    oraArrivo time NOT NULL,  
    aeroportoPartenza char(3) NOT NULL,  
    aeroportoArrivo char(3) NOT NULL,  
    PRIMARY KEY (id),  
    FOREIGN KEY (aeroportoPartenza) REFERENCES aeroporto (sigla),  
    FOREIGN KEY (aeroportoArrivo) REFERENCES aeroporto (sigla)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

3. Creazione della tabella “viaggio”. La tabella contiene l’attributo “stato” di tipo ENUM. Vi sono state definiti infatti i tre tipi di stato del quale un viaggio può essere considerato. Sono state inoltre espresse due coppie di chiavi uniche. Ciò è dovuto dal fatto che un pilota e un aeromobile non possono partecipare a più di un viaggio nello stesso giorno (vincolo stabilito dalla compagnia aerea). Questo impedisce infatti la creazione di un nuovo viaggio avente un pilota o un aereo che abbiano già partecipato a un viaggio in quella data.

```
CREATE TABLE IF NOT EXISTS viaggio (  
    idVolo char(5) NOT NULL,  
    dataV date NOT NULL,  
    stato enum ('concluso', 'in corso', 'da effettuare'),  
    aeromobile char(4) NOT NULL,  
    pilota char(4) NOT NULL,  
    PRIMARY KEY (dataV, idVolo),  
    FOREIGN KEY (idVolo) REFERENCES volo (id),  
    FOREIGN KEY (pilota) REFERENCES pilota (matricola),  
    FOREIGN KEY (aeromobile) REFERENCES aereo (matricola),  
    UNIQUE KEY (dataV, pilota),  
    UNIQUE KEY (dataV, aeromobile)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

4. Creazione della tabella “prenotazione”. La tabella prenotazione associa un viaggiatore al viaggio che egli ha prenotato. È stata definita una chiave unica al fine di impedire che vi siano più prenotazioni dello stesso viaggiatore per uno stesso viaggio.

```
CREATE TABLE IF NOT EXISTS prenotazione (  
    idPrenotazione char(5) NOT NULL,  
    cfViaggiatore char(16) NOT NULL,  
    idVolo char(5) NOT NULL,  
    dataV date NOT NULL,  
    PRIMARY KEY (idPrenotazione),  
    FOREIGN KEY (cfViaggiatore) REFERENCES viaggiatore (cf),  
    FOREIGN KEY (idVolo) REFERENCES viaggio (idVolo),  
    FOREIGN KEY (dataV) REFERENCES viaggio (dataV),  
    UNIQUE KEY (cfViaggiatore, idVolo, dataV)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

5. Inserimento di due nuovi viaggiatori all’interno della tabella “viaggiatore” (Operazione 1).

```
INSERT INTO viaggiatore (cf, nome, cognome, dataNascita, telefono)  
VALUES  
    ('FGHJKL12M34N5670', 'Marco', 'Rossi', '1985-05-12', '3485689456'),  
    ('QWERTY67U89I1230', 'Laura', 'Neri', '1993-07-22', '3397845632');
```

6. Inserimento di tre nuovi voli all’interno del database (Operazione 3).

```
INSERT INTO volo (id, durata, distanza, oraPartenza, oraArrivo, aeroportoPartenza, aeroportoArrivo)  
VALUES  
    ('V0001', '01:30:00', 500, '07:30', '09:00', 'FCO', 'MXP'),  
    ('V0002', '02:15:00', 1000, '11:00', '13:15', 'AMS', 'CDG'),  
    ('V0003', '00:45:00', 250, '13:30', '14:15', 'NCE', 'PSA');
```


7. Query che visualizza la matricola degli aeromobili condotti dal pilota con matricola "P009" nei suoi viaggi.

```
1 • SELECT DISTINCT aeromobile
2 FROM viaggio
3 WHERE pilota = 'P009';
```

Result Grid		Filter Rows:
aeromobile		
A005		
A015		
A012		
A014		

8. Query che visualizza i viaggi che sono al momento in corso.

```
1 • SELECT *
2 FROM viaggio
3 WHERE stato = 'in corso';
```

result Grid

Filter Rows:

Export

idVolo	dataV	stato	aeromobile	pilota
V0002	2023-02-28	in corso	A002	P002
V0003	2023-02-28	in corso	A003	P003
V0008	2023-02-28	in corso	A008	P008
V0009	2023-02-28	in corso	A029	P017
V0010	2023-02-28	in corso	A001	P021
V0015	2023-02-28	in corso	A015	P014

9. Query che visualizza il numero di voli con partenza da aeroporti italiani.

```
1 • SELECT COUNT(*) AS totaleVoliDaItalia
2 FROM volo
3 JOIN aeroporto ON volo.aeroportoPartenza = aeroporto.sigla
4 WHERE aeroporto.nazione = 'Italia';
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
totaleVoliDaItalia			
12			

10. Query che visualizza i piloti e il numero dei viaggi che hanno condotto nel mese di marzo. I piloti sono poi stati ordinati in modo decrescente in base al numero di viaggi svolti (Operazione 4).

```
1 SELECT p.matricola, p.nome, p.cognome, COUNT(*) AS numeroViaggi
2 FROM viaggio v
3 JOIN pilota p ON v.pilota = p.matricola
4 WHERE MONTH(v.dataV) = 3
5 GROUP BY p.matricola
6 ORDER BY numeroViaggi DESC;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
matricola	nome	cognome	numeroViaggi
P009	Elisa	Conti	3
P001	Mario	Rossi	2
P008	Dario	Gallo	2
P012	Roberta	Romano	2
P013	Simone	Moretti	2
P015	Marco	Ricci	2
P017	Filippo	Lombardi	2
P018	Valentina	Serra	2
P027	Lorenzo	Conti	2

11. Query che visualizza per ogni cliente quante prenotazioni ha effettuato per i viaggi nel mese di marzo (Operazione 5).

```
1 • SELECT v.cf, v.nome, v.cognome, COUNT(*) AS numeroPrenotazioni
2 FROM prenotazione p
3 JOIN viaggiatore v ON p.cfViaggiatore = v.cf
4 WHERE MONTH(p.dataV) = 3
5 GROUP BY v.cf
```

cf	nome	cognome	numeroPrenotazioni
CSTNGN91D21A000R	Angela	Castagna	1
DPPMRA00A41A662P	Mario	Rossi	1
FGHJKL12M34N567A	Sara	Ricci	1
FGHJKL12M34N567O	Marco	Rossi	1
LCCGPP92P54L219V	Giuseppe	Liccardo	1
PLCMMM96T09L219E	Mimmo	Pellecchia	4
RSFVRN88A12L219V	Vincenzo	Rosafio	2
SMTMRA95C12L219W	Alessandra	Santoro	1
VNTMRA92B24E183B	Maria	Ventimiglia	1
ZXVBN12KJ34H567H	Filippo	Serra	1

Lorenzo Comis X81000848

Catania 27/02/2023