

# LABORATORIO BASI DI DATI

## 2022/23

Progetto Laboratorio di basi di dati: ISBD Crociere

Gruppo concettuale logico

Autore: Luca Borrello

Laboratorio di basi di dati, Anno Accademico: 2022/2023

II Semestre

# **INDICE:**

1. INTRODUZIONE
2. RELAZIONE CONCETTUALE
  - 2.1 SCHEMA CONCETTUALE
  - 2.2 NARRATIVA SCHEMA CONCETTUALE
  - 2.3 DESCRIZIONE ASSOCIAZIONI
3. SCHEMA RELAZIONALE
  - 3.1 SCHEMA RELAZIONALE LOGICO
  - 3.2 DALLO SCHEMA CONCETTUALE A QUELLO RELAZIONALE
  - 3.3 IDENTIFICAZIONE CHIAVI PRIMARIE
  - 3.3 RAPPRESENTAZIONI ASSOCIAZIONI 1:1
  - 3.4 RAPPRESENTAZIONI ASSOCIAZIONI 1:N
  - 3.5 RAPPRESENTAZIONI ASSOCIAZIONI N:N
  - 3.6 RAPPRESENTAZIONI GERARCHIE
4. SCHEMA RELAZIONALE
  - 4.1 SCHEMA LOGICO TESTUALE
5. CREAZIONE DELLE TABELLE NELLA BASI DI DATI

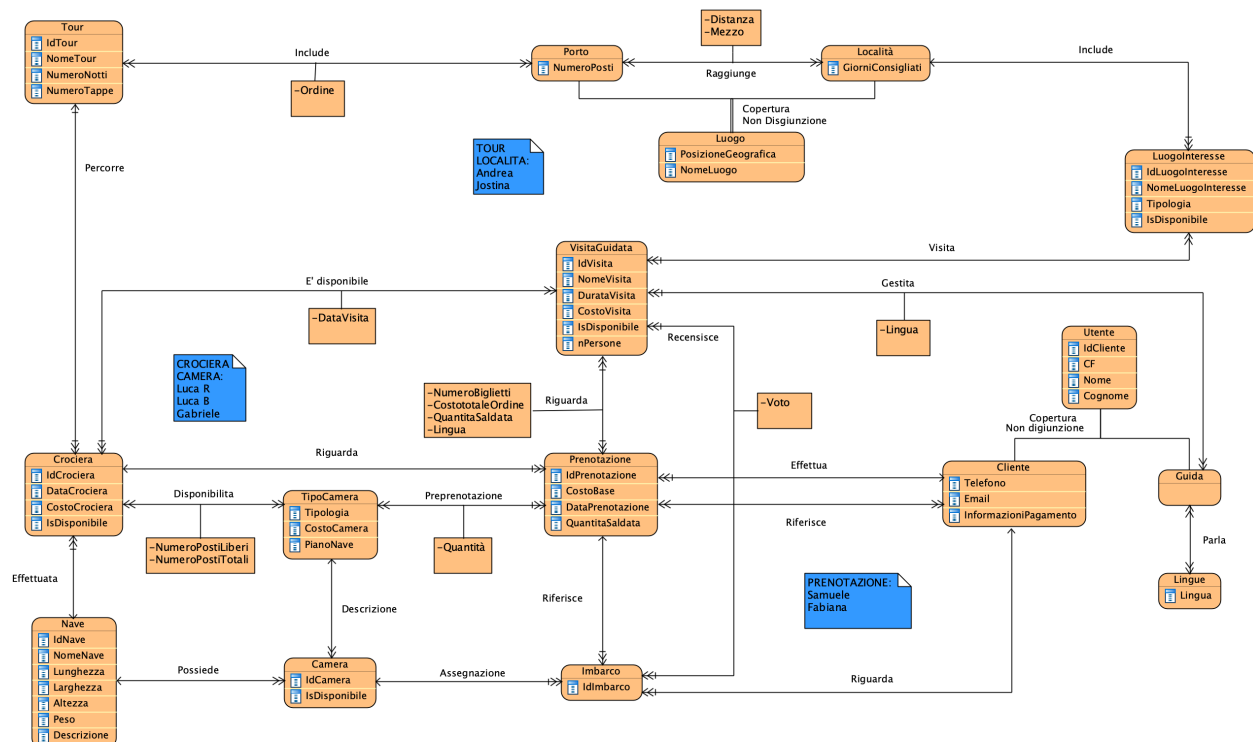
## 1. INTRODUZIONE:

L'azienda "ISBD Crociere" ha commissionato un'applicazione web che interagisce con una base di dati per gestire l'organizzazione di crociere e per permettere a clienti di prenotare comodamente crociere e visite in luoghi di interesse artistici.

Il sistema offre ai clienti un elenco di crociere prenotabili e mostra le possibili visite che si possono effettuare partendo dalle varie tappe della crociera.

## 2. RELAZIONE CONCETTUALE

## 2.1 SCHEMA CONCETTUALE:



## 2.2 NARRATIVA SCHEMA CONCETTUALE:

Questa sezione sarà dedicata interamente alla descrizione delle tabelle dello schema concettuale, mentre nel successivo paragrafo analizziamo più tecnicamente i dettagli degli attributi e i loro eventuali vincoli.

1. **Utente**(IdCliente, CF, Nome, Cognome):
  - a. utente che vuole usufruire del servizio di prenotazioni di crociere
  - b. questa tabella contiene tutte le informazioni relative a un singolo utente.
2. **Guida**(IdCliente, CF, Nome, Cognome):
  - a. contiene le stesse informazioni della tabella padre Utente
  - b. identifica una persona che gestisce una precisa guida per una visita
3. **Cliente**(IdCliente, CF, Nome, Cognome, Telefono, Email, InformazioniPagamento):
  - a. contiene alcune informazioni più del padre, come Telefono Email e il metodo usato per pagare
  - b. identifica la persona che prenota e paga una crociera
4. **Prenotazione**(IdPrenotazione, CostoBase, DataPrenotazione, QuantitàSaldata):
  - a. prenotazione che l'utente effettua
  - b. il cliente è identificato tramite la chiave esterna "codice fiscale" nel logico.
  - c. la crociera è identificata tramite la chiave esterna IdCrociera nel logico.
  - d. "QuantitàSaldata" identifica il pagamento parziale della crociera. La prenotazione si considera valida solo se il pagamento parziale equivale al costo base della crociera, entro sette giorni dalla data partenza.
5. **Imbarco** (IdImbarco):
  - a. identifica che una prenotazione è stata saldata ed effettivamente abbiamo la carta di imbarco per salire sulla nave.
6. **Camera** (IdCamera, IsDisponibile):
  - a. identifica le camere all'interno di una nave e se la cabina è disponibile o no.
7. **TipoCamera** (Tipologia, Costo, PianoNave):
  - a. identifica la tipologia di camera e il relativo costo, indipendentemente dalla nave su cui è presente la cabina.
8. **Nave** (IdNave, NomeNave, Lunghezza, Larghezza, Altezza, Peso, Descrizione):
  - a. identifica una nave, con le relative informazioni e dati.
9. **Crociera** (IdCrociera, DataCrociera, CostoCrociera, IsDisponibile):
  - a. identifica una crociera con la relativa data di partenza.

- b. `isDisponibile` identifica se è ancora disponibile prenotare la crociera.
- 10. **Tour** (`IdTour`, `NumeroNotti`, `NumeroTappe`):
  - a. identifica il tour e il numero di notti in cui viene effettuato e in quante tappe consiste.
- 11. **Porto** (`PosizioneGeografica`, `NomePorto`, `NumeroPosti`):
  - a. indica un porto, con la sua posizione geografica e il numero di navi massimo che può ospitare.
- 12. **Località** (`PosizioneGeografica`, `Nome`, `GiorniConsigliati`):
  - a. indica una località, specificando nome e giorni di permanenza consigliati.
- 13. **Luogo di Interesse** (`PosizioneGeografica`, `NomeLuogoInteresse`, `Tipologia`, `IsDisponibile`)
  - a. rappresenta una lista di luoghi di interesse visitabili presenti in una località, specificandone il tipo
  - b. `IsDisponibile` indica se quel luogo di interesse può essere meta di una visita guidata o non è disponibile per lavori o altro.
- 14. **VisitaGuidata** (`IdVisita`, `NomeVisita`, `DurataVisita`, `CostoVisita`, `IsDisponibile`, `nPersone`):
  - a. indica tutte le visite guidate offerte, con il costo aggiuntivo da pagare e la durata.
  - b. `IsDisponibile` indica se la visita può essere prenotata o no perchè un luogo di interesse non è disponibile.
  - c. `nPersone` rappresenta il numero massimo di persone che possono partecipare alla visita
- 15. **Guida** (`IdGuida`, `Nome`, `Cognome`):
  - a. indica tutte le guide disponibile dall'organizzazione ISBD, con le loro informazioni
- 16. **Lingue**(`Lingua`):
  - a. indica tutte le lingue disponibili che vengono parlate dalle guide.

## 2.3 DESCRIZIONE ASSOCIAZIONI:

Questa sezione sarà dedicata alla descrizione delle associazioni tra tabelle e attributi delle associazioni.

- 1. Include (Tour - Porto):
  - possono essere raggiunti uno o più porti da un tour e un porto è raggiunto in uno o più tour;
  - permette di tenere traccia di tutti i porti che vengono raggiunti da un preciso tour
    - Ordine: informazione che tiene traccia dell'ordine in cui i porti vengono raggiunti, per facilitare anche la tracciabilità di precise informazioni

2. Raggiunge (Porto - Località):

- possono essere raggiunte una o più località da un porto e un porto è raggiunto da una o più località;
- indica tutte le località che si possono raggiungere da un porto. E' possibile raggiungere stesse località da diversi porti
  - Distanza: indica la distanza delle diverse località dal porto
  - Mezzo: indica il mezzo con cui può avvenire lo spostamento

3. Include (Località - LuogoInteresse):

- una località include nessuno o più luoghi di interesse e un luogo di interesse è incluso solo in una località
- indica tutti i luoghi di interesse inclusi in una località

4. Visita (LuogoInteresse - VisitaGuidata):

- una visita guidata visita uno o più luoghi di interesse e un luogo di interesse è visitato da molte visite guidate o nessuna, se ad esempio non è stato ancora assegnato nessun luogo
- indica tutti i luoghi di interesse che vengono visitati da una precisa visita guidata

5. Gestita (VisitaGuidata - Guida):

- una visita guidata è gestita da una o più guide e una guida gestisce una o più visite guidate
- indica tutte le visite guidate che una guida gestisce
  - Lingua: indica l'informazione in cui viene svolta una visita guidata dalla guida

6. Parla (Guida - Lingue):

- una lingua è parlata da una o più guide e una guida parla una lingua o più lingue
- indica tutte le lingue che parla una guida

7. Effettua (Cliente - Prenotazione):

- un cliente effettua nessuna o molte prenotazioni in quanto può non prenotare nonostante sia registrato e una precisa prenotazione è effettuata solo da un cliente
- indica tutte le prenotazioni che vengono fatte dal cliente

8. Riferisce (Cliente - Prenotazione):

- una prenotazione fa riferimento a uno o molti clienti e un cliente riferisce ad una o molte prenotazioni
- indica tutti i clienti a cui fa riferimento una prenotazione

9. Riguarda (Cliente - Imbarco):

- ad un cliente viene accettato nessuno o più imbarchi, in caso non paghi la quota e un imbarco riferisce solo ad un cliente
- indica tutti gli imbarchi intesi come successi di prenotazioni dei clienti

10. Recensisce (Imbarco - VisitaGuidata):

- un cliente in imbarco può recensire nessuna o più visite, in caso non partecipi alle visite, e una visita guidata è recensita da nessuno o molti
- indica tutte le recensioni alle visite che vengono fatte da persone effettivamente imbarcate
  - Voto: un numero che indica il voto della recensione

11. Riguarda (Prenotazione - VisitaGuidata):

- una prenotazione riguarda nessuna, se un cliente non acquista nulla, o molte visite guidate e una visita guidata riguarda nessuna, se non è stata acquistata da nessuno, o molti
- indica tutte le prenotazioni di visite che vengono effettuate con dettagli
  - NumeroBiglietti: quantità di biglietti che vengono acquistati per una visita
  - CostoTotaleOrdine: somma totale del dovuto da un cliente
  - QuantitàSaldata: indica se un cliente può partecipare alla visita o no se ha saldato
  - Lingua: indica la lingua in cui viene svolta la visita guidata

12. E' disponibile (Crociera - VisitaGuidata):

- una visita guidata è disponibile per nessuna crociera, se è stata appena organizzata, o molte e una crociera ha disponibile una o più visite guidate
- indica tutte le visite disponibili per una crociera
  - DataVisita: indica una informazione aggiuntiva che informa quando viene effettuata la visita

13. Percorre (Tour - Crociera):

- una crociera percorre nessuno tour, se non è stato ancora definito o esattamente uno e un tour è percorso da nessuna crociera se per esempio non è stata scelta la nave o molte
- indica la relazione tra la crociera e il relativo tour percorso

14. Riguarda (Crociera - Prenotazione):

- una prenotazione riguarda una sola crociera e una crociera riguarda nessuna prenotazione, se è stata appena assegnata, o molte
- indica tutte le prenotazioni che fanno riferimento ad una precisa crociera

15. Disponibilità (Crociera - TipoCamera):

- una crociera fa riferimento a uno o più tipi di camera e un tipo di camera fa riferimento a una o più crociera
- indica tutti i tipi di cabine disponibili per ogni crociera
  - NumeroPostiLiberi: informazione che ci permette di tenere traccia del riempimento della crociera
  - NumeroPostiTotali: informazione che indica la capienza massima per una precisa crociera ricordando che una crociera può essere la stessa nave che percorre un tour in diversi periodi

#### 16. Effettuata (Crociera - Nave):

- una crociera è effettuata da una sola nave e una nave effettua nessuna crociera, se è stata appena creata o in manutenzione, o molte
- indica tutte le crociere che effettua una nave

#### 17. Possiede (Nave - Camera):

- una nave possiede una o molte camere e una camera riguarda una sola precisa nave
- indica tutte le camere disponibili di una nave

#### 18. Descrizione (TipoCamera - Camera):

- una camera riguarda un solo tipo di camera e un tipo di camera riferisce a uno o molte camere
- indica la descrizione di ogni camera come tipologia e costo

#### 19. Prenotazione (TipoCamera - Prenotazione):

- un tipo di camera viene assegnata nessuna prenotazione, se ancora nessuno ha prenotato, o molte e una prenotazione riferisce a una o più tipi di camera
- indica tutte le tipologie di camere che fanno riferimento ad una prenotazione
  - Quantità: informazione per tenere traccia della quantità di tipologia di camera prenotate dalla relativa prenotazione del cliente

#### 20. Assegnazione (Camera - Imbarco):

- ad un imbarco è assegnata una sola camera e una camera è assegnata a nessun imbarco, se ancora non è disponibile la nave, o a molti
- indica la rispettiva camera di una prenotazione che fa riferimento all'imbarco

#### 21. Riferisce (Imbarco - Prenotazione):

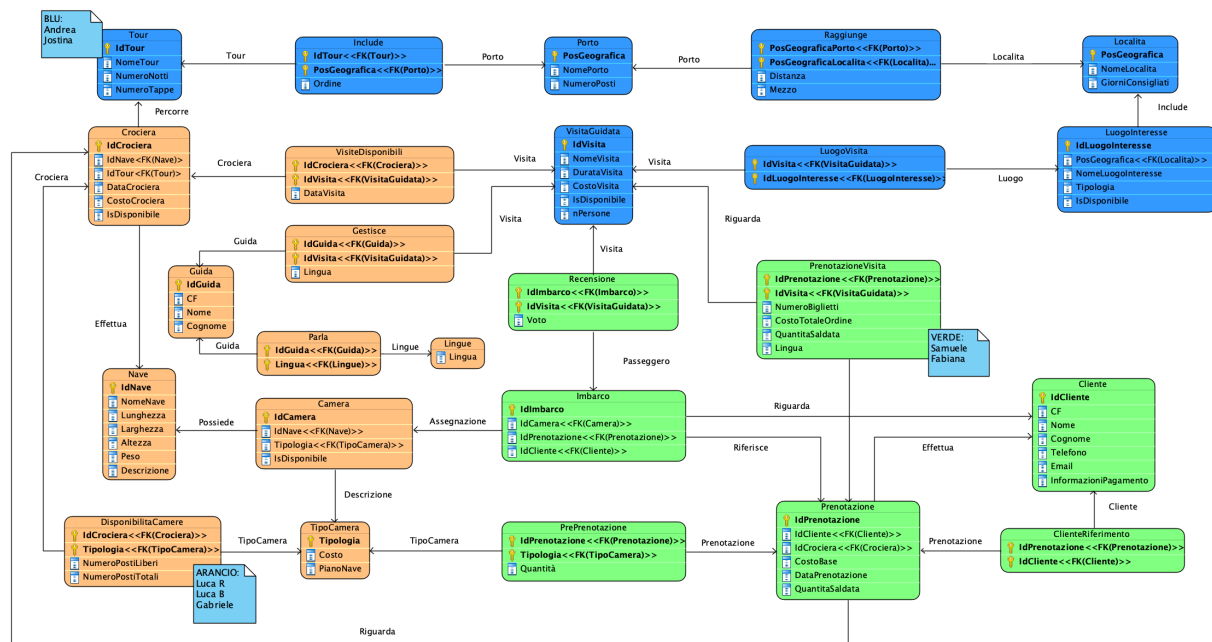
- un imbarco riferisce ad una prenotazione sola e una prenotazione riferisce a nessun imbarco, se non viene saldato l'importo, o molti, se si prenota anche per qualcun'altro
- indica tutti i riferimenti di imbarco relativi ad una precisa prenotazione



### 3. SCHEMA RELAZIONALE:

Passiamo alla descrizione della progettazione relazionale logico.

#### 3.1 SCHEMA RELAZIONALE LOGICO:



#### 3.2 DALLO SCHEMA CONCETTUALE A QUELLO LOGICO:

Per la trasformazione dello schema concettuale nel relativo schema logico relazionale si è tenuto conto dei seguenti aspetti fondamentali:

- Identificazione chiavi primarie;
- Rappresentazione delle associazioni uno ad uno e uno molti;
- Rappresentazione delle associazioni molti a molti;
- Rappresentazioni delle gerarchie;

### 3.3 IDENTIFICAZIONE CHIAVI PRIMARIE

Per la scelta delle chiavi primarie è necessario individuare un attributo capace di identificare univocamente ogni record di una certa entità, ciò avviene scegliendo da un certo numero di chiavi dette candidate quella che si ritiene essere la migliore, una chiave primaria deve infatti possedere alcune buone proprietà fra queste se ne citano alcune.

Una chiave primaria deve essere:

- univoca
- semplice (in termini di usabilità)
- breve

### 3.3 RAPPRESENTAZIONI ASSOCIAZIONI 1:1

Nella rappresentazione delle associazioni uno ad uno è possibile aggiungere la chiave esterna ad una qualunque delle due entità considerate nell'associazione, la nostra scelta è stata quella di inserire la chiave esterna nell'entità verso la quale l'associazione è parziale:

Nel nostro caso in particolare, non esistono associazioni di molteplicità 1:1 quindi non tratteremo questo caso.

### 3.4 RAPPRESENTAZIONI ASSOCIAZIONI 1:N

Per la rappresentazione delle associazioni uno a molti si è provveduto ad aggiungere agli attributi della relazione rispetto a cui l'associazione è univoca una chiave esterna che riferisce l'altra relazione.

Vedere per esempio l'associazione uno a molti fra: NAVE ed CAMERA, l'entità CAMERA contiene una chiave

esterna NOMENAVE che riferisce la chiave primaria della **NAVE** ad essa associata.

In questo modo in CAMERA abbiamo tutte le informazioni di cui abbiamo bisogno per tenere traccia dei dati. IdCamera e NomeNave insieme formano Chiave Primaria per CAMERA, in quanto solo l'id della camera non è sufficiente poichè ci sono ripetizioni di numero tra le camere.

### 3.5 RAPPRESENTAZIONI ASSOCIAZIONI N:N

La rappresentazione delle associazioni molti a molti avviene aggiungendo allo schema una nuova entità che contiene le due chiavi esterne che riferiscono le entità coinvolte.

Ad esempio notiamo come sia stato necessario implementarlo per l'associazione tra CROCIERA e VISITAGUIDATA.

Abbiamo introdotto l'entità VISITEDISPONIBILI per tenere traccia di quali visite (NomeVisita) sono disponibili avendo come informazioni NomeNave, NomeTour e DataCrociera.

## 3.6 RAPPRESENTAZIONI GERARCHIE

Le gerarchie per come sono definite a livello concettuale non sono rappresentabili su uno schema relazionale, per l'implementazione delle gerarchie su uno schema relazionale ci sono varie possibilità, se ne citano alcuni solo per informazione:

- Rappresentazione unica;
- Partizionamento verticale;
- Partizionamento orizzontale.

A partire dallo schema concettuale bisogna osservare che esistono due gerarchie: quella tra :

- UTENTE(superclasse) - CLIENTE(sottoclasse) e GUIDA(sottoclasse);
- LUOGO(superclasse) - LOCALITA'(sottoclasse) e PORTO(sottoclasse).

Per entrambe le gerarchie abbiamo deciso di eliminare il padre, alla base di questa scelta vi è il fatto che non ci fosse alcun vantaggio nel tenere sia il padre che i figli, e non avendo molte informazioni aggiuntive il padre, abbiamo deciso di accorpare tutte le informazioni e farle ereditare ai figli. In questo modo risulta anche semplificata la traduzione delle associazioni poiché in entrambe le gerarchie la classe padre non ha nessuna associazione con altre entità.

Tenere separati località e porto ci dà il vantaggio di poter sapere esattamente chi è l'uno o l'altra, con la possibilità però che in qualche caso vi sarà ripetizione di alcune città.

## 4. SCHEMA RELAZIONALE:

Vengono qui descritte le tabelle finali ottenute e i singoli attributi con eventualmente i vincoli.

### 4.1 SCHEMA LOGICO TESTUALE

Descriviamo qui lo schema logico in forma testuale. Vengono evidenziate le chiavi primarie e chiavi esterne come mostrato nella legenda

LEGENDA: Primary\_Key, Foreign\_Key\*

1. **RECENSIONE**(IdImbarco\*, IdVisita\*, Voto):

- Identifica la recensione di un cliente per una specifica visita, rappresentata da un voto da 1 a 5
- IdImbarco\* + NomeVisita\*: chiave primaria
- IdImbarco\*: chiave esterna per IMBARCO
- NomeVisita\*: chiave esterna per VISITAGUIDATA
- Voto: un intero, indica il voto per la recensione ad una precisa visita.
- VINCOLI:
  - $1 \leq \text{VOTO} \leq 5$

2. **IMBARCO**(IdImbarco, IdCamera\*, IdPrenotazione\*, IdCliente\*):

- IdImbarco: chiave primaria, un intero univoco, indica il numero di imbarco relativo ad una prenotazione
- IdCamera\*: chiave esterna per CAMERA
- NomeNave\*: chiave esterna per CAMERA
- IdPrenotazioneCF\*: chiave esterna per PRENOTAZIONE
- IdCliente\*: chiave esterna per CLIENTE

3. **CLIENTE**(IdCliente, CF, Nome, Cognome, Telefono, Email, InformazioniPagamento):

- IdCliente: un intero univoco indica l'identificativo
- CF: chiave primaria, una stringa, indica il codice fiscale dell'utente
- Nome: stringa, indica il nome dell'utente
- Cognome: stringa, indica il cognome dell'utente
- Telefono: stringa, indica il numero di telefono dell'utente
- Email: stringa, indica il contatto email dell'utente
- InformazioniPagamento: stringa, indica il metodo di pagamento utilizzato e registrato dell'utente

4. **CLIENTERIFERIMENTO**(IdPrenotazione\*, IdCliente\*):

- Identifica tutti i cliente collegati a una prenotazione, un cliente può prenotare una crociera per altre persone e questa tabella permette di identificare quali clienti viaggiano insieme.
- IdPrenotazione\* + IdCliente\*: chiave primaria
- IdPrenotazione\*: chiave esterna per PRENOTAZIONE
- IdCliente\*: chiave esterna per CLIENTE

5. **PRENOTAZIONE**(IdPrenotazione, IdCliente\*, IdCrociera\*, CostoBase, DataPrenotazione, QuantitàSaldata);

- IdPrenotazione: chiave primaria, indica il codice di prenotazione
- IdCliente\*: chiave esterna per CLIENTE
- IdCrociera\*: chiave esterna per CROCIERA
- CostoBase: un intero calcolabile, indica quanto è il prezzo base che l'utente deve saldare per salire a bordo entro i 7 giorni previsti
- DataPrenotazione: un intero (DDMMYYYY) indica la data in cui è stata effettuata la prenotazione. Viene utilizzata per controllare che venga saldata entro i 7 giorni.
- QuantitàSaldata: un intero, indica la spesa saldata dal cliente. Se uguale a CostoBase allora la prenotazione è andata a buon fine
- VINCOLI:
  - $0 \leq \text{QuantitàSaldata} \leq \text{CostoBase}$
  - $\text{DataPrenotazione} \leq \text{DataCrociera}$  (riferimento a quella crociera)

6. **PRENOTAZIONEVISITA**(IdPrenotazione\*, IdVisita\*, NumeroBiglietti, CostoTotaleOrdine, QuantitàSaldata, Lingua);

- Permette di sapere quali visite un utente ha acquistato
- IdPrenotazione\* + NomeVisita\*: chiave primaria
- IdVisita\*: chiave esterna per VISITAGUIDATA
- IdPrenotazione\*: chiave esterna per PRENOTAZIONE
- NumeroBiglietti: un intero, indica quanti biglietti per una precisa visita sono stati comprati
- CostoTotaleOrdine: un intero, indica il costo totale dell'ordine da saldare

- **QuantitàSaldata**: un intero, indica quanto è stato saldato dal cliente. Se uguale a **CostoTotaleOrdine** allora prenotazione andata a buon fine
- **Lingua**: una stringa, indica in che lingua è stata prenotata la visita guidata
- **VINCOLI**:
  - $0 < \text{NumeroBiglietti}$
  - $0 \leq \text{QuantitàSaldata} \leq \text{CostoTotaleOrdine}$
  - $\text{Sum}(\text{NumeroBiglietti}) < n\text{Persone}$

7. **PREPRENOTAZIONE**(IdPrenotazione\*, Tipologia\*, Quantità):

- Associa la prenotazione al tipo di camera desiderato così da sapere quante cabine prenotate per utente
- **IdPrenotazione\* + Tipologia\***: chiave primaria
- **IdPrenotazione\***: chiave esterna per PRENOTAZIONE
- **Tipologia\***: chiave esterna per TIPOCAMERA
- **Quantità**: quantità di camere che vengono prenotate
- **VINCOLI**:
  - $0 < \text{Quantità}$

8. **TIPOCAMERA**(Tipologia, Costo, PianoNave):

- **Tipologia**: chiave primaria, una stringa, indica la tipologia della stanza
- **Costo**: un intero, indica il costo della stanza
- **PianoNave**: un intero, indica il piano della nave
- **VINCOLI**:
  - $0 < \text{Costo}$
  - $0 < \text{PianoNave}$

9. **DISPONIBILITACAMERE**(IdCrociera\*, Tipologia\*, NumeroPostiLiberi, NumeroPostiTotali):

- Permette di avere una panoramica dei posti disponibili per ogni tipologia di camera per ogni crociera non ancora partita.
- **IdCrociera\* + Tipologia\***: chiave primaria

- IdCrociera\*: chiave esterna per NAVE
- Tipologia\*: chiave esterna per TIPOCAMERA
- NumeroPostiLiberi: un intero, indica i posti liberi rimasti su una precisa nave. Si aggiorna per ogni prenotazione andata a buon fine o cancellata
- NumeroPostiTotali: un intero, indica i posti totali disponibili su una nave
- VINCOLI:
  - $0 \leq \text{NumeroPostiLiberi} \leq \text{NumeroPostiTotali}$

10. **CAMERA**(IdCamera, IdNave\*, Tipologia\*, IsDisponibile):

- IdCamera: chiave primaria, un intero, indica il numero di cabina della stanza UNIVOCO.
- IdNave\*: chiave esterna per NAVE
- Tipologia\*: chiave esterna per TIPOCAMERA
- IsDisponibile: indica se la camera della nave è disponibile

11. **NAVE**(IdNave, NomeNave, Lunghezza, Larghezza, Altezza, Peso, Descrizione):

- IdNave: un intero univoco indica l'identificativo.
- NomeNave: chiave primaria, un intero.
- Lunghezza: un intero, indica la lunghezza della nave
- Larghezza: un intero indica la larghezza della nave
- Altezza: un intero indica l'altezza della nave
- Peso: un intero indica l'altezza della nave
- Descrizione: una stringa, indica una descrizione testuale
- VINCOLI:
  - $0 < \text{Lunghezza}, \text{Larghezza}, \text{Altezza}, \text{Peso}$

12. **GUIDA**(IdGuida, CF, Nome, Cognome):

- IdGuida: chiave primaria, un intero univoco indica l'identificativo.
- CF: una stringa, indica il codice fiscale della guida
- Nome: stringa, indica il nome dell'utente

- Cognome: stringa, indica il cognome dell'utente

13. **PARLA**(IdGuida\*, Lingua\*):

- Tutte le lingue parlate dalla guida
- IdGuida + Lingua: chiave primaria
- IdGuida\*: chiave esterna per Guida
- Lingua\*: chiave esterna per Lingue

14. **LINGUE**(Lingua):

- Lingua: chiave primaria, una stringa

15. **GESTISCE**(IdGuida\*, IdVisita\*, Lingua):

- Tutte le visite che una guida gestisce, con la molteplicità per la lingua
- IdGuida\* + IdVisita\* : chiave primaria
- IdGuida\*: chiave esterna per GUIDA
- IdVisita\*: chiave esterna per VISITAGUIDATA
- Lingua: una stringa, indica la lingua per cui è disponibile la visita

16. **LUOGOVISITA**(IdVisita\*, IdLuogoInteresse\*):

- Indica tutte i luoghi di interesse disponibili per una visita
- IdVisita\* + NomeLuogo\*: chiave primaria
- IdVisita\*: chiave esterna per VISITAGUIDATA
- IdLuogoInteresse\*: chiave esterna per LUOGOINTERESSE

17. **VISITAGUIDATA**(IdVisita, NomeVisita, DurataVisita, CostoVisita, IsDisponibile, nPersone):

- IdVisita: chiave primaria, un intero univoco, indica l'identificatore.
- NomeVisita: una stringa, indica il nome della visita guidata messa a disposizione
- DurataVisita: un intero (M) indica il tempo della durata della visita in minuti



- CostoVisita: un intero, indica il costo della visita
- IsDisponibile: un intero (0,1) indica se la visita è disponibile
- nPersone: un intero, indica quante persone massimo possono partecipare alla visita
- VINCOLI:
  - $0 < DurataVisita$
  - $0 < CostoVisita$
  - $0 < nPersone$

18. **VISITEDISPONIBILI**(IdCrociera\*, IdVisita\*, DataVisita):

- Contiene tutte le visite disponibili per una precisa crociera
- IdVisita\* + IdCrociera\*: chiave primaria
- IdCrociera\*: chiave esterna per CROCIERA.
- IdVisita\*: chiave esterna per VISITAGUIDATA.
- DataVisita: un intero (DDMMYYYY) indica la data in cui è viene effettuata la la visita per quella precisa crociera
- VINCOLI:
  - $DataCrociera + NumeroNotti > DataVisita > DataCrociera$

19. **CROCIERA**(IdCrociera, IdNave\*, IdTour\*, DataCrociera, CostoCrociera, IsDisponibile):

- IdCrociera: chiave primaria, un intero, indica l'identificativo UNIVOCO della crociera corrispondente
- IdNave\*: chiave esterna per NAVE
- IdTour\*: chiave esterna per TOUR
- DataCrociera: chiave primaria, un intero (DDMMYYYY) indica la data in cui si svolge la crociera
- CostoCrociera: un intero calcolato, indica quanto costa la crociera (variabile in base alla stagione, cioe data, tour e nave)
- IsDisponibile: un intero (0,1) indica se la crociera è disponibile
- VINCOLI:
  - $0 < CostoCrociera$

20. **TOUR**(IdTour, NomeTour, NumeroNotti, NumeroTappe):

- IdTour: chiave primaria, un intero, indica l'identificativo
- NomeTour: chiave primaria, una stringa, indica il nome del tour univoco
- NumeroNotti: un intero, indica in quanti giorni il tour viene effettuato
- NumeroTappe: un intero, indica quante tappe totali fa il relativo tour. Utile per trovare le ultime tappe comuni
- VINCOLI:
  - $0 < \text{NumeroNotti}$
  - $0 < \text{NumeroTappe}$

21. **INCLUDE**(IdTour\*, PosGeografica\*, Ordine):

- Contiene le informazioni di tutti porti raggiunti dal tour
- IdTour\* + PosizioneGeografica\*: chiave primaria
- IdTour\*: chiave esterna per TOUR
- PosGeografica\*: chiave esterna per PORTO
- Ordine: un intero, indica l'ordine con cui i porti vengono raggiunti durante un preciso tour. Serve per tenere traccia delle tappe
- VINCOLI:
  - $0 < \text{Ordine}$

22. **PORTO**(PosGeografica, NomePorto, NumeroPosti):

- PosGeografica: chiave primaria, una stringa, indica la posizione assoluta ed univoca di un luogo
- NomePorto: una stringa, indica il nome del porto dove la nave si ferma
- NumeroPosti: un intero, indica quante navi è predisposto a far attraccare un determinato porto
- VINCOLI:
  - $0 < \text{NumeroPosti}$

23. **RAGGIUNGE**(PosGeografica\*, PosGeografica\*, Distanza, Mezzo):

- PosGeografica\* + PosizioneGeografica\*: chiave primaria

- PosGeografica\*: chiave esterna per PORTO
- PosGeografica\*, chiave esterna per LOCALITA'
- Distanza: un intero, indica la distanza in chilometri (km) tra la località e il porto dove è attraccata la nave
- Mezzo: una stringa, indica mezzo di trasporto previsto per lo spostamento
- VINCOLI:
  - $0 < \text{Distanza}$

24. **LOCALITA**(PosGeografica, NomeLocalità, GiorniConsigliati):

- PosGeografica: chiave primaria, una stringa, indica la posizione assoluta e univoca in cui si trova
- NomeLocalità: una stringa, indica il nome della località raggiunta da un porto (possibile porto=località)
- GiorniConsigliati: un intero, indica il numero di giorni consigliati per quella precisa località (considerando le attrazioni da visitare)
- VINCOLI:
  - $0 < \text{GiorniConsigliati}$

25. **LUOGOINTERESSE**(IdLuogoInteresse, PosGeografica\*, NomeLuogoInteresse, Tipologia, IsDisponibile):

- IdLuogoInteresse: chiave primaria, un intero univoco per identificare.
- PosGeografica\*: chiave esterna per LOCALITA'
- NomeLuogoInteresse: chiave primaria, una stringa, indica il nome del luogo di interesse (attrazione)
- Tipologia: una stringa, indica la specifica di tipologia di luogo di interesse
- IsDisponibile: un intero (0,1) indica se il luogo di interesse è disponibile

## 5. CREAZIONE DELLE TABELLE NELLA BASE DI DATI:

Qui di seguito è presente il codice per la eliminazione, creazione di tutte le tabelle necessarie e sequenze per il progetto “Crociere”:

```

-- @Author : LucaBorrello - AndreaFerrari

drop table CAMERA CASCADE CONSTRAINTS;
drop table CLIENTERIFERIMENTO CASCADE CONSTRAINTS;
drop table CROCIERA CASCADE CONSTRAINTS;
drop table GESTISCE CASCADE CONSTRAINTS;
drop table GUIDA CASCADE CONSTRAINTS;
drop table IMBARCO CASCADE CONSTRAINTS;
drop table INCLUDE CASCADE CONSTRAINTS;
drop table LINGUE CASCADE CONSTRAINTS;
drop table LOCALITA CASCADE CONSTRAINTS;
drop table LUOGOVISITA CASCADE CONSTRAINTS;
drop table LUOGOINTERESSE CASCADE CONSTRAINTS;
drop table NAVE CASCADE CONSTRAINTS;
drop table PARLA CASCADE CONSTRAINTS;
drop table PORTO CASCADE CONSTRAINTS;
drop table RAGGIUNGE CASCADE CONSTRAINTS;
drop table PRENOTAZIONE CASCADE CONSTRAINTS;
drop table PRENOTAZIONEVISITA CASCADE CONSTRAINTS;
drop table PREPRENOTAZIONE CASCADE CONSTRAINTS;
drop table DISPONIBILITACAMERE CASCADE CONSTRAINTS;
drop table RECENSIONE CASCADE CONSTRAINTS;
drop table TIPOCAMERA CASCADE CONSTRAINTS;
drop table TOUR CASCADE CONSTRAINTS;
drop table VISITAGUIDATA CASCADE CONSTRAINTS;
drop table VISITEDISPONIBILI CASCADE CONSTRAINTS;

-- @Author : Gabriele

drop table RUOLOPERMESSO CASCADE CONSTRAINTS;
drop table UTENTERUOLO CASCADE CONSTRAINTS;
drop table EMAIL CASCADE CONSTRAINTS;
drop table TELEFONO CASCADE CONSTRAINTS;
drop view SESSIONE CASCADE CONSTRAINTS;
drop view Utente CASCADE CONSTRAINTS;
drop table BE_SESSIONE CASCADE CONSTRAINTS;
drop table BE_UTENTE CASCADE CONSTRAINTS;
drop table RUOLO CASCADE CONSTRAINTS;
drop table PERMESSO CASCADE CONSTRAINTS;

-- @Author : Gabriele

CREATE TABLE be_UTENTE(
    IDUtente INT PRIMARY KEY,
    CF CHAR(16) NOT NULL UNIQUE,
    Nome VARCHAR2(255) NOT NULL,
    Cognome VARCHAR2(255) NOT NULL,
    InformazioniPagamento VARCHAR2(255) NOT NULL,
    EPrivilegiato NUMBER(1) NOT NULL,
    SALT RAW(32) NOT NULL,
    username VARCHAR2(64) NOT NULL UNIQUE,
    password RAW(32) NOT NULL
);

CREATE TABLE Ruolo (
    IDRuolo INT PRIMARY KEY,
    Nome VARCHAR2(32) NOT NULL
);

```

```

CREATE TABLE Permesso (
    Nome VARCHAR2(32) PRIMARY KEY
);

CREATE TABLE be_SESSIONE(
    IDSessione INT PRIMARY KEY,
    IDUtente INT NOT NULL,
    EPrivilegiato NUMBER(1) NOT NULL,
    TStamp TIMESTAMP(4) NOT NULL,
    PEPPER RAW(32) NOT NULL,
    FOREIGN KEY(IDUtente) REFERENCES BE_Utente(IDUtente)
);

CREATE TABLE Telefono (
    IDUtente INT NOT NULL,
    Telefono VARCHAR2(32) PRIMARY KEY,
    FOREIGN KEY(IDUtente) REFERENCES BE_Utente(IDUtente)
);

CREATE TABLE Email (
    IDUtente INT NOT NULL,
    Email VARCHAR2(255) PRIMARY KEY,
    FOREIGN KEY(IDUtente) REFERENCES BE_Utente(IDUtente)
);

CREATE TABLE UtenteRuolo (
    IDUtente INT NOT NULL,
    IDRuolo INT NOT NULL,
    FOREIGN KEY(IDRuolo) REFERENCES Ruolo(IDRuolo),
    FOREIGN KEY(IDUtente) REFERENCES BE_Utente(IDUtente),
    PRIMARY KEY(IDRuolo, IDUtente)
);

CREATE TABLE RuoloPermesso (
    IDRuolo INT NOT NULL,
    NomePermesso VARCHAR2(32) NOT NULL,
    FOREIGN KEY(IDRuolo) REFERENCES Ruolo(IDRuolo),
    FOREIGN KEY(NomePermesso) REFERENCES Permesso(Nome),
    PRIMARY KEY(IDRuolo, NomePermesso)
);

CREATE VIEW Utente AS
    SELECT idutente, username, CF, Nome, Cognome, INFORMAZIONIPAGAMENTO, EPRIVILEGIATO
    FROM be_UTENTE
;

CREATE VIEW Sessione AS
    SELECT IDSESSIONE, IDUTENTE, EPRIVILEGIATO, TSTAMP
    FROM be_SESSIONE
;

-- @Author : LucaBorrello - AndreaFerrari

Create table GUIDA (
    IDGuida int PRIMARY KEY,
    CF CHAR(16) not null,
    Nome VARCHAR2(225) not NULL,

```

```

        Cognome VARCHAR2(225) not NULL
    );

create table LINGUE (
    Lingua VARCHAR2(255) not null PRIMARY KEY
);

create table PARLA (
    IDGuida int not null,
    Lingua VARCHAR2(255) not null,
    FOREIGN KEY (IDGuida) REFERENCES GUIDA(IDGuida),
    FOREIGN KEY (Lingua) REFERENCES LINGUE(Lingua),
    PRIMARY KEY (IDGuida, Lingua)
);

CREATE TABLE VISITAGUIDATA (
    IDVisita int PRIMARY KEY,
    NomeVisita varchar2(225) not null,
    DurataVisita int not NULL,
    CostoVisita int not NULL,
    isDisponibile NUMBER(1) not NULL,
    nPersone int not NULL,
    check (0 < DurataVisita and 0 < CostoVisita and 0 < nPersone)
);

CREATE TABLE NAVE(
    IDNave int PRIMARY KEY,
    NomeNave varchar2(255) not null,
    Lunghezza int not null,
    Larghezza int not null,
    Altezza int not null,
    Peso int not null,
    Descrizione varchar2(255) not null,
    check (0<Lunghezza and 0<Larghezza and 0 < Altezza and 0 < Peso)
);

CREATE TABLE TOUR (
    IDTour int PRIMARY KEY,
    NomeTour VARCHAR2(225) not null,
    numeroNotti int not NULL,
    numeroTappe int not null,
    check (0 < numeroNotti and 0 < numeroTappe)
);

CREATE TABLE PORTO (
    PosGeografica VARCHAR2(255) not null PRIMARY KEY,
    NomePorto VARCHAR2(255) not NULL,
    NumeroPosti int not NULL,
    check(0 < NumeroPosti)
);

CREATE TABLE LOCALITA(
    PosGeografica VARCHAR2(255) PRIMARY KEY,
    NomeLocalita VARCHAR2(255) not NULL,
    GiorniConsigliati int not null,
    check(0 < GiorniConsigliati)
);

```

```

CREATE TABLE LUOGOINTERESSE (
    IDLuogointeresse int PRIMARY KEY,
    PosGeografica VARCHAR2(255) REFERENCES LOCALITA(PosGeografica),
    NomeLuogointeresse VARCHAR2(255),
    Tipologia VARCHAR2(255) not NULL,
    isDisponibile NUMBER(1) not null
);

CREATE TABLE GESTISCE (
    IDGuida int not null REFERENCES Guida(IDGuida),
    IDVisita int not null REFERENCES VisitaGuidata(IDVisita),
    lingua VARCHAR2(225) not NULL,
    PRIMARY KEY (IDGuida, IDVisita)
);

CREATE TABLE LUOGOVISITA (
    IDVisita int not null REFERENCES VisitaGuidata(IDVisita),
    IDLuogointeresse int not null REFERENCES LUOGOINTERESSE(IDLuogointeresse),
    PRIMARY KEY (IDVisita, IDLuogointeresse)
);

CREATE TABLE CROCIERA(
    IDCrociera int not null primary key,
    IDNave int not null REFERENCES NAVE(IDNave),
    IDTour int not null REFERENCES TOUR(IDTour),
    DataCrociera date not null,
    CostoCrociera int not null,
    isDisponibile NUMBER(1) not null,
    check (0 < CostoCrociera )
);

CREATE table VISITEDISPONIBILI(
    IDCrociera int not null REFERENCES CROCIERA(IDCrociera),
    IDVisita int not null REFERENCES VISITAGUIDATA(IDVisita),
    DataVisita date not null,
    PRIMARY KEY (IDCrociera, IDVisita)
);

CREATE TABLE INCLUDE (
    IDTour int REFERENCES TOUR(IDTour),
    PosGeografica VARCHAR2(255) REFERENCES PORTO(PosGeografica),
    Ordine int not null,
    PRIMARY KEY (IDTour, PosGeografica),
    check(0 < Ordine)
);

CREATE TABLE RAGGIUNGE(
    PosGeograficaPorto VARCHAR2(255) REFERENCES PORTO(PosGeografica),
    PosGeograficaLocalita VARCHAR2(255) REFERENCES LOCALITA(PosGeografica),
    Distanza int not NULL,
    Mezzo VARCHAR2(255) not null,
    PRIMARY KEY (PosGeograficaPorto, PosGeograficaLocalita),
    check(0 <= Distanza)
);

CREATE TABLE PRENOTAZIONE(
    IdPrenotazione int not null primary key,

```

```

        IDCliente int NOT NULL references BE_UTENTE(IDUTENTE),
        IDCrociera int not null REFERENCES CROCIERA(IDCrociera),
        CostoBase int not null,
        DataPrenotazione date not null,
        QuantitaSaldata int not null,
        check (QuantitaSaldata >= 0 and QUANTITASALDATA <= CostoBase)
    );

CREATE TABLE TIPOCAMERA(
    Tipologia varchar2(255) not null primary key,
    Costo int not null,
    PianoNave int not null,
    check (0 < Costo and 0 < PianoNave)
);

CREATE TABLE CAMERA(
    IdCamera int not null primary key,
    IDNave int not null references NAVE(IDNave),
    Tipologia varchar2(255) not null references TIPOCAMERA(Tipologia),
    isDisponibile int not null
);

CREATE TABLE IMBARCO(
    IdImbarco int not null primary key,
    IdCamera int not null references CAMERA(IdCamera),
    IdPrenotazione int not null references PRENOTAZIONE(IdPrenotazione),
    IDCliente int not null references BE_UTENTE(IDUTENTE)
);

CREATE TABLE PRENOTAZIONEVISITA(
    IdPrenotazione int not null references PRENOTAZIONE(IdPrenotazione),
    IDVisita int not null references VISITAGUIDATA(IDVisita),
    NumeroBiglietti int not null,
    CostoTotaleOrdine int not null,
    QuantitaSaldata int not null,
    LINGUA VARCHAR2(255) NOT NULL REFERENCES LINGUE(LINGUA),
    primary key(IdPrenotazione, IDVisita),
    check (0<NumeroBiglietti and QuantitaSaldata<=CostoTotaleOrdine and 0<=QuantitaSaldata)
);

CREATE TABLE PREPRENOTAZIONE(
    IdPrenotazione int not null references PRENOTAZIONE(IdPrenotazione),
    Tipologia varchar2(255) not null references TIPOCAMERA(Tipologia),
    Quantita int not null,
    primary key(IdPrenotazione, Tipologia),
    check (0 < Quantita)
);

CREATE TABLE RECENSIONE(
    IdImbarco int not null references IMBARCO(IdImbarco),
    IDVisita int not null references VISITAGUIDATA(IDVisita) ,
    Voto int not null,
    primary key(IdImbarco, IDVisita),
    check (1<=Voto and Voto <= 5)
);

CREATE TABLE CLIENTERIFERIMENTO (
    IdPrenotazione int NOT NULL references PRENOTAZIONE(IdPrenotazione),
    IDCliente int NOT NULL references BE_UTENTE(IDUTENTE),

```



```
        primary key(IdPrenotazione, IDCliente)
    );

CREATE TABLE DISPONIBILITACAMERE(
    IDCrociera int not null references CROCIERA(IDCrociera),
    Tipologia varchar2(255) not null references TIPOCAMERA(Tipologia),
    NumeroPostiLiberi int not null,
    NumeroPostiTotali int not null,
    primary key(IDCrociera, Tipologia),
    check (NumeroPostiLiberi<=NumeroPostiTotali and NumeroPostiLiberi>=0)
);
```