



PROGETTO BASI DI DATI: GESTIONE DEI MARVEL PUB

Università degli Studi di Napoli Parthenope – Facoltà di Scienze e Tecnologie

Corso di Laurea in Informatica – Il anno – A.A. 2021/2022

Progetto a cura di
MADONNA ENRICO - 0124002279
NOCERINO GIORGIO – 0124002311
IZZO ANTONIO - 0124002447

Data di consegna: 20/07/2022
Docente di Basi di Dati – MARATEA ANTONIO

RELAZIONE DEL PROGETTO

INDICE

1. Raccolta e Analisi dei Requisiti

- 1.1. Raccolta Requisiti
- 1.2. Glossario
- 1.3. Specifiche sulle Operazioni di Base
- 1.4. Specifiche sulle Operazioni di Utente

2. Progettazione Concettuale

- 2.1. Creazione dello Schema EE-R
- 2.2. Documentazione dello Schema EE-R
 - 2.2.1. Dizionario delle Entità
 - 2.2.2. Dizionario delle Associazioni
- 2.3. Regole di Vincolo
 - 2.3.1. Vincoli d'Integrità Statici
 - 2.3.2. Vincoli d'Integrità Dinamici

3. Progettazione Logica

- 3.1. Tavola dei volumi a regime
- 3.2. Tavola degli Utenti
- 3.3. Tavola delle Operazioni
- 3.4. Tavole dei descrittori delle Operazioni
- 3.5. Use Case (Casi d'Uso)

4. **Modello Relazionale**

4.1. Dal Modello EE-R al Relazionale

4.2. Elenco delle Relazioni del Modello Logico Relazionale

4.3. Verifica delle forme normali

4.3.1. Prima forma normale

4.3.2. Seconda forma normale

4.3.3. Terza forma normale

4.3.4. Forma normale di Boyce e Codd (BCNF)

5. **Implementazione SQL**

5.1. Creazione delle Tabelle

5.2. Creazione delle Viste

5.3. Creazione degli Utenti

5.4. Creazione dei Trigger

5.5. Procedure PL/SQL

5.6. Popolamento fittizio del DB

5.7. Scheduler

1. Raccolta ed Analisi dei Requisiti

1.1 Raccolta Requisiti

Si vuole realizzare una base di dati che registra il funzionamento di una catena di pub, chiamata Marvel Pub. Tale catena, registra i clienti che effettuano una prenotazione al tavolo o che richiedono una consegna.

Dei **clienti** interessa il recapito telefonico per identificarli, il nome e il cognome, mentre della prenotazione si vuole sapere l'orario e la data. Inoltre, i clienti possono essere sia coloro che si trovano al pub (*sul posto, bloccano un tavolo*) o coloro che sono nelle rispettive abitazioni (*a domicilio*).

Un cliente può possedere una **tessera** identificata da un codice che dà diritto ad uno sconto sulla consumazione.

Ogni **sede del pub** ha il proprio indirizzo, il proprio numero di telefono e degli orari di apertura e chiusura. Ogni pub possiede delle sale che contengono i rispettivi **tavoli** che i clienti prenotano e dai quali partono le ordinazioni.

Ogni pub ha al massimo 12 **dipendenti** di cui interessa sapere nome, cognome, data di nascita, sesso (che può essere registrato come Maschio, Femmina, Non Specificato) e codice fiscale. Tale dipendente può essere un *cameriere*, un *cuoco* o un *rider* (che effettua le consegne ai clienti la richiedono).

Ovviamente, tutti i dipendenti registrano i propri **turni di lavoro** di cui interessa sapere la data, l'ora e quanto dura (per legge, non più di 8 ore) e poi ricevono uno **stipendio** minimo di 800 euro (di quest'ultimo interessa sapere la data dell'importo e le trattenute) e che per politiche interne non può superare i 1600 euro se il dipendente non ha già lavorato per almeno un anno. Inoltre, ad ogni dipendente spetta un giorno di riposo a settimana e non può lavorare per più di 190 ore al mese.

Invece, ogni rider effettua delle **consegne** ai clienti d'asporto. Tali consegne interessano sapere la data e l'ora, la via e la città del cliente. Ovviamente queste consegne contengono delle "**ordinazioni a domicilio**".

Comunque, i clienti anziché effettuare le ordinazioni a domicilio, possono, in alternativa, recarsi alla sede tramite la **prenotazione** fatta dal cliente stesso. Di tale prenotazione interessa sapere la data e l'ora. Per politica del Pub un cliente non può bloccare più di 5 tavoli alla volta e può effettuare prenotazioni solo per il giorno stesso o per i due successivi.

Interessa sapere che ogni **ordinazione** (di cui serve tenere traccia del **codice ordinativo**) è associata ad un **conto** che il cliente deve pagare tramite contanti o carta.

Inoltre, l'ordinazione va a comprendere più di un **prodotto(menu)** che può essere un **panino** o una **bibita** (della quale vogliamo sapere se è alcolica). Ovviamente ogni panino è composto da ingredienti (**materie prima**). Tali prodotti (materie prime e bibite) compongono le forniture.

Il pub richiede infatti **forniture** necessarie per il continuo delle vendite. Di una fornitura interessa sapere il codice, il nome del fornitore, il prezzo d'acquisto, la quantità dell'acquisto,

la data della richiesta e della consegna fatta dal pub e la data di scadenza della partita richiesta che deve essere garantita almeno per una settimana dopo la richiesta.

Inoltre, un pub può stabilire una partnership con un **parcheggio**, dove i clienti, possono effettuare una sosta ad un prezzo orario vantaggioso. Di tale parcheggio, interessa sapere posizione dove è situato, l'orario di apertura e di chiusura, il nome del parcheggio, quanti posti sono riservati ai clienti, e quanto è distante dal pub.

1.2 Glossario

Termine	Descrizione	Sinonimi	Collegamenti
<i>Cliente</i>	Persona che si reca nel pub per prenotare il tavolo, oppure a domicilio l'ordinazione	Acquirente	Tessera, da Cliente (Sul Posto): Prenotazione; da Cliente (D'Asporto): Ordinazione (A Domicilio)
<i>Tessera</i>	Una scheda che può essere utile al cliente per ricevere lo sconto	Card	Cliente
<i>Conto</i>	Scontrino inviato al cliente una volta pagato la sua ordinazione	Scontrino	Ordinazione, Pub
<i>Pub</i>	Sede dove i clienti (sul posto) prenotati si siedono vicino ai tavoli e dove vengono gestiti dai dipendenti	Sede	Tavolo, Conto, Dipendente, Fornitura, Parcheggio
<i>Dipendente</i>	Sono i personali dove gestiscono i pub, registrano i turni lavorativi e ricevono lo stipendio e si può specializzare in Cameriere, Cuoco o Rider	Personale, Lavoratore	Turno di Lavoro, Stipendio, Pub, da Dipendente (Rider): Consegna-Ordinazione (A Domicilio)
<i>Stipendio</i>	Dove i dipendenti del pub ricevono ogni mese	Paga	Dipendente
<i>Turno di Lavoro</i>	Dove i dipendenti registrano le presenze nel pub	Giorno lavorativo	Dipendente
<i>Ordinazione</i>	Gli ordini fatti dai clienti e possono essere o al tavolo o a domicilio	Ordine	Da Ordinazione (A Domicilio): Cliente (D'Asporto), Consegna-Dipendente (Rider); da Ordinazione (Al Tavolo): Prenotazione

<i>Consegna</i>	Dove i rider effettuano ordinazioni a domicilio	-	Dipendente (Rider), Ordinazione (A Domicilio)
<i>Prenotazione</i>	Fatto dai clienti che si trovano in sede	-	Blocca-Tavolo, Cliente (Sul Posto), Ordinazione (Al Tavolo)
<i>Tavolo</i>	Dove le persone si recano a prenotare per permettere di bloccare il tavolo	-	Pub, Blocca-Prenotazione
<i>Menù</i>	Dove il cliente tramite l'ordinazione sceglie il menù che gli piace dove può scegliere la bibita e/o il panino	Prodotto, Bibita, Panino	Comprende-Ordinazione, da Menù (Bibita): Fornitura; da Menù (Panino): Ingrediente-Materia Prima
<i>Materia Prima</i>	Egli compone come ingrediente del panino	Ingrediente	Fornitura, Ingrediente-Menù (Panino)
<i>Fornitura</i>	Dove il pub fa ogni volta la richiesta	Merce	Pub, Materia Prima, Menù (Bibita)
<i>Parcheggio</i>	Dove i clienti possono effettuare una sosta per recarsi al pub	-	Pub

1.3 Specifiche sulle Operazioni di Base

Ecco un elenco delle operazioni di base:

- **Inserimento Cliente:** Inserimento di un Cliente che si può essere suddiviso in Sul Posto e D'Asporto;
- **Inserimento Cliente Sul Posto:** Inserimento di un Cliente Posto;
- **Inserimento Cliente D'Asporto:** Inserimento di un Cliente D'Asporto;
- **Inserimento Tessera:** Inserimento di una Tessera per il Cliente;
- **Inserimento Conto:** Inserimento di un Conto;
- **Inserimento Pub:** Inserimento di un Pub della catena;
- **Inserimento Dipendente:** Inserimento di un Dipendente che lavora in un Pub. Si può specializzare in Cuoco, Cameriere oppure Rider;
- **Inserimento Cameriere:** Inserimento di un Cameriere;
- **Inserimento Cuoco:** Inserimento di un Cuoco;
- **Inserimento Rider:** Inserimento di un Rider;
- **Inserimento Stipendio:** Inserimento di uno Stipendio per il Dipendente;
- **Inserimento Turno di Lavoro:** Inserimento di un Turno di Lavoro per il Dipendente;
- **Inserimento Ordinazione:** Inserimento di un Ordinazione fatta dal Cliente (Sul Posto/D'Asporto);
- **Inserimento Ordinazione Al Tavolo:** Inserimento di un Ordinazione fatta dal Cliente (Sul Posto);

- **Inserimento Ordinazione A Domicilio:** Inserimento di un Ordinazione fatta dal Cliente (D'Asporto);
- **Inserimento Consegna:** Inserimento di una Consegna;
- **Inserimento Prenotazione:** Inserimento di una Prenotazione fatta dal Cliente (Sul Posto) che blocca un Tavolo;
- **Inserimento Tavolo:** Inserimento di un Tavolo;
- **Inserimento Menù:** Inserimento di un prodotto del Menù;
- **Inserimento Panino:** Inserimento di un Panino;
- **Inserimento Bibita:** Inserimento di un Bibita;
- **Inserimento Materia Prima:** Inserimento di una Materia Prima del Panino;
- **Inserimento Fornitura:** Inserimento di una Fornitura richiesta dal Pub;
- **Inserimento Parcheggio:** Inserimento di un Parcheggio affiliato al Pub.

1.4 Specifiche sulle Operazioni di Utente

Segue invece, un elenco delle operazioni degli utenti.

- **Eliminazione Cliente:** Elimina le informazioni del Cliente se e solo se quest'ultimo non effettua ordinazioni da più di un anno. Coinvolge 5 tabelle: Cliente D'Asporto, Ordinazione A Domicilio, Ordinazione, Conto e Consegna;
- **Aumenta Stipendio:** Aumenta lo stipendio del Dipendente se quest'ultimo ha lo guadagna meno di 1500 euro e ha lavorato più di 160h nell'ultimo mese;
- **Attiva Sconto:** Attiva lo sconto del 10% sul totale del conto se, durante il calcolo dell'importo da pagare il sistema rileva che il cliente possiede una tessera;
- **Disponibilità Parcheggio:** Permette al cliente di sapere se il pub è dotato di un parcheggio affiliato o meno;
- **Calcola Scorte:** Permette al gestore delle forniture di sapere le scorte disponibili in magazzino di un determinato prodotto (materia prima o bibita);
- **Bonus Dipendente:** se nel mese selezionato si raggiunge un obiettivo minimo di vendite, allora i dipendenti del pub avranno un bonus sullo stipendio.

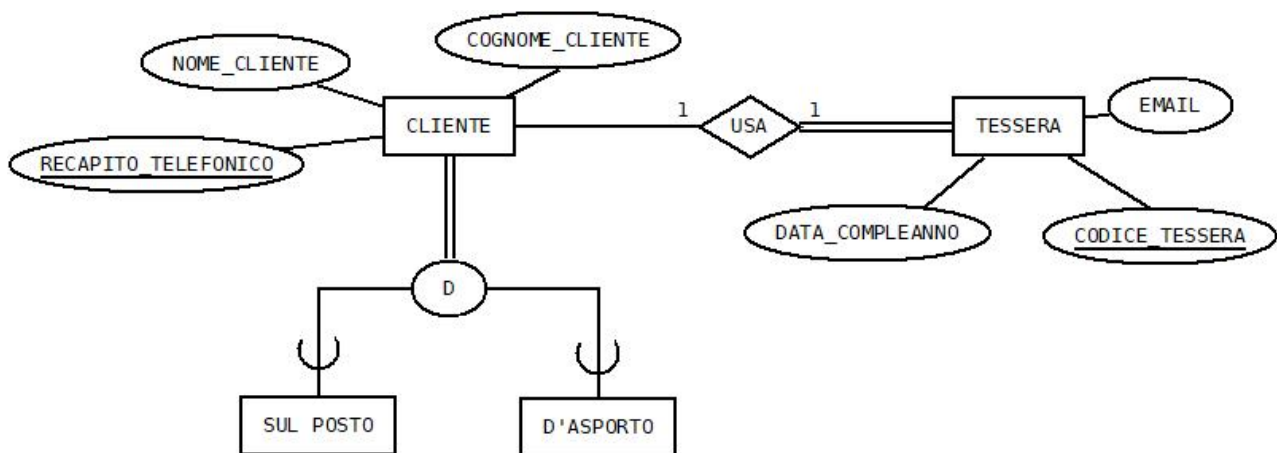
2. Progettazione Concettuale

2.1 Creazione dello Schema EE-R (Extended Entity - Relationship)

CLIENTE E TESSERA

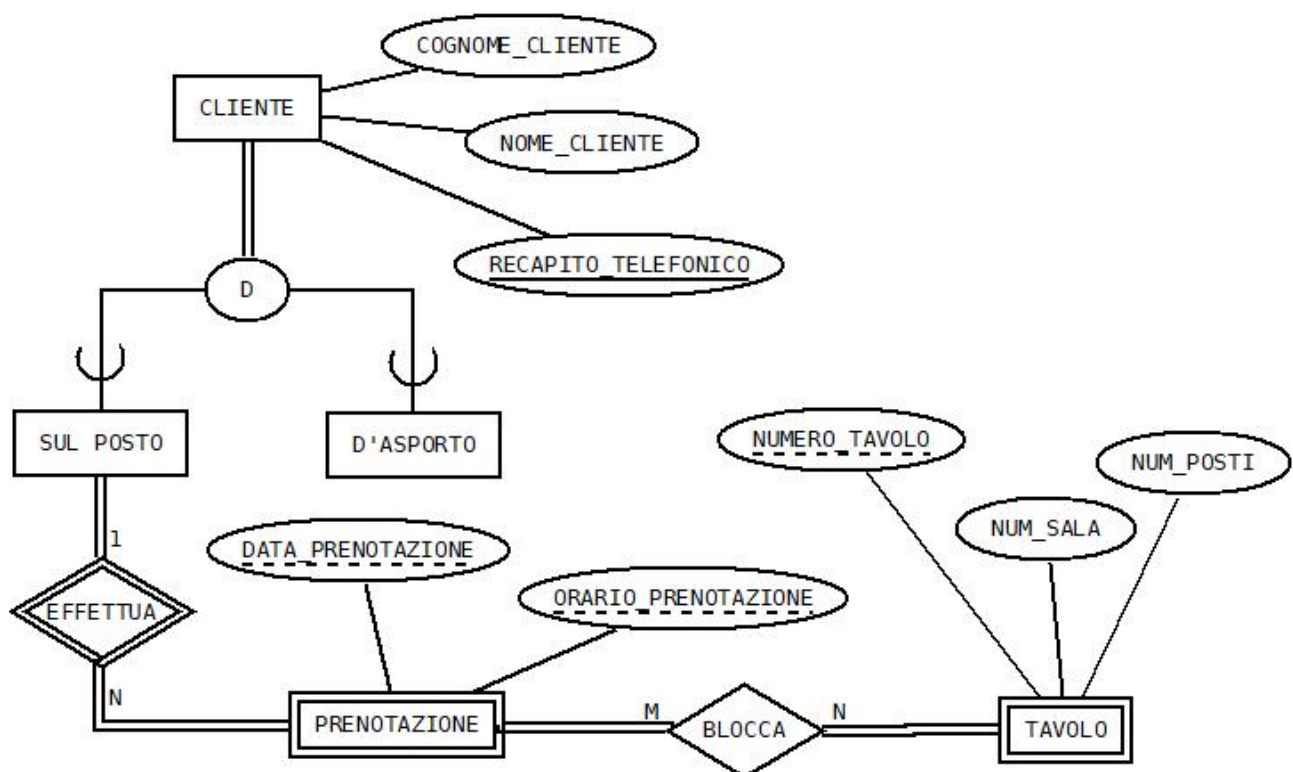
Analizzando lo schema EE-R è possibile riconoscere un primo modello fondamentale: l'attività principale del Marvel Pub. Essa è incentrata sulla figura del Cliente, il quale può essere un Cliente Sul Posto o D'Asporto, e può avere una Tesserà. E come vedremo in seguito le classi figlie del Cliente saranno collegate alle chiavi primarie di Prenotazione e Ordinazione A Domicilio.

A questo punto andremo ad analizzare le entità per descrivere dettagliatamente tutti i requisiti. Si introducono inoltre, per ogni entità e relazione, i relativi attributi. In primo luogo, concentrandosi sull'entità Cliente con la chiave primaria Recapito Telefonico e i suoi attributi: Nome Cliente e Cognome Cliente. È possibile effettuare una specializzazione, derivandola nelle entità: Cliente Sul Posto con la sua chiave Recapito Telefonico Sul Posto che collegherà alla chiave primaria della classe madre e Cliente D'Asporto con la sua chiave Recapito Telefonico D'Asporto che collegherà sempre alla chiave primaria della classe madre. Introduciamo anche l'entità Tesserà con chiave primaria Codice_Tesserà e con gli attributi: Data Compleanno Cliente, Email e Recapito Cliente, che ovviamente non si vede dal punto di vista schematico, ma si relaziona a Cliente tramite l'associazione Usa, con la totalità verso Tesserà poiché tutte le tessere appartengono a un cliente:



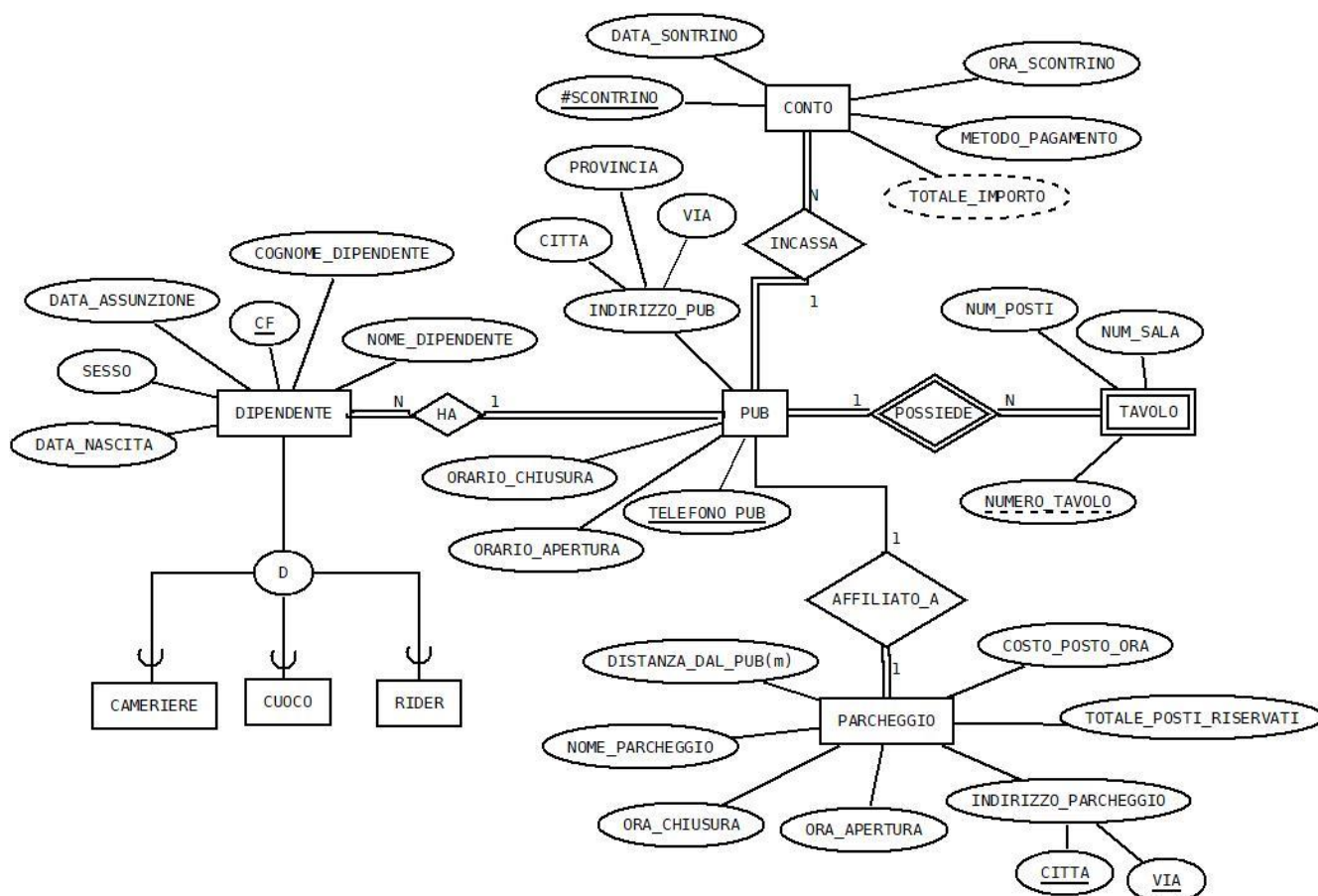
PRENOTAZIONE E TAVOLO

Passiamo a un altro schema. Come detto in precedenza, le classi figlie del Cliente saranno collegate alle chiavi primarie di Prenotazione e Ordinazione A Domicilio. Concentrandoci sull'entità Cliente Sul Posto con la chiave primaria Recapito Telefonico Sul Posto, sarà collegato alla chiave primaria di Prenotazione che sarebbe Data Prenotazione e Ora Prenotazione (che saranno fusi nel modello relazionale) con le sue chiavi esterne (che ovviamente non si vedono nel modello concettuale): Recpt Tel, Num Tavolo e Tf Pub. La prima sarà collegata, tramite l'associazione Effettua, alla chiave della classe figlia del Cliente (Sul Posto), mentre le ultime due, tramite l'associazione Blocca M:N, saranno collegate alla chiave prima di Tavolo.



PUB, CONTO, PARCHEGGIO E DIPENDENTE

Ora ci concentreremo sul Pub. Il pub ha la sua chiave primaria Telefono Pub e i suoi attributi: Orario Apertura e Chiusura, Indirizzo che si scompone in Via, Città e Provincia. Il pub è collegato con alcune entità: tramite l'associazione Possiede, un pub può possedere più Tavoli, con l'associazione Incassa, un pub può appunto incassare più Conti [con chiave primaria ID_Scontrino e i suoi attributi: Data e Ora Scontrino, Metodo Pagamento e due chiavi esterne Recapito_Pub che collega a Pub e ID_Ordine che collegherà a Ordinazione (di questo vedremo di seguito) e infine un attributo proveniente dall'associazione Riguarda (collegamento verso Ordinazione) chiamato Sconto], e infine tramite l'associazione Ha, un pub deve necessariamente avere dei Dipendenti che possono specializzarsi in Cameriere, Cuoco oppure Rider. Dipendente contiene la chiave primaria CF e i suoi attributi: Nome e Cognome Dipendente, Data di Nascita, Data di Assunzione, Sesso e la chiave esterna che collega verso chiave primaria di Pub. Inoltre quest'ultimo, può possedere un Parcheggio, tramite l'associazione Affiliato_a, la cui chiave primaria rappresenta l'indirizzo stesso e i suoi attributi: Costo Posto all'Ora, Orario Apertura e Chiusura, Nome Parcheggio, Totale Posti Riservati, Distanza dal Pub e la chiave esterna che si collegherà al Pub:

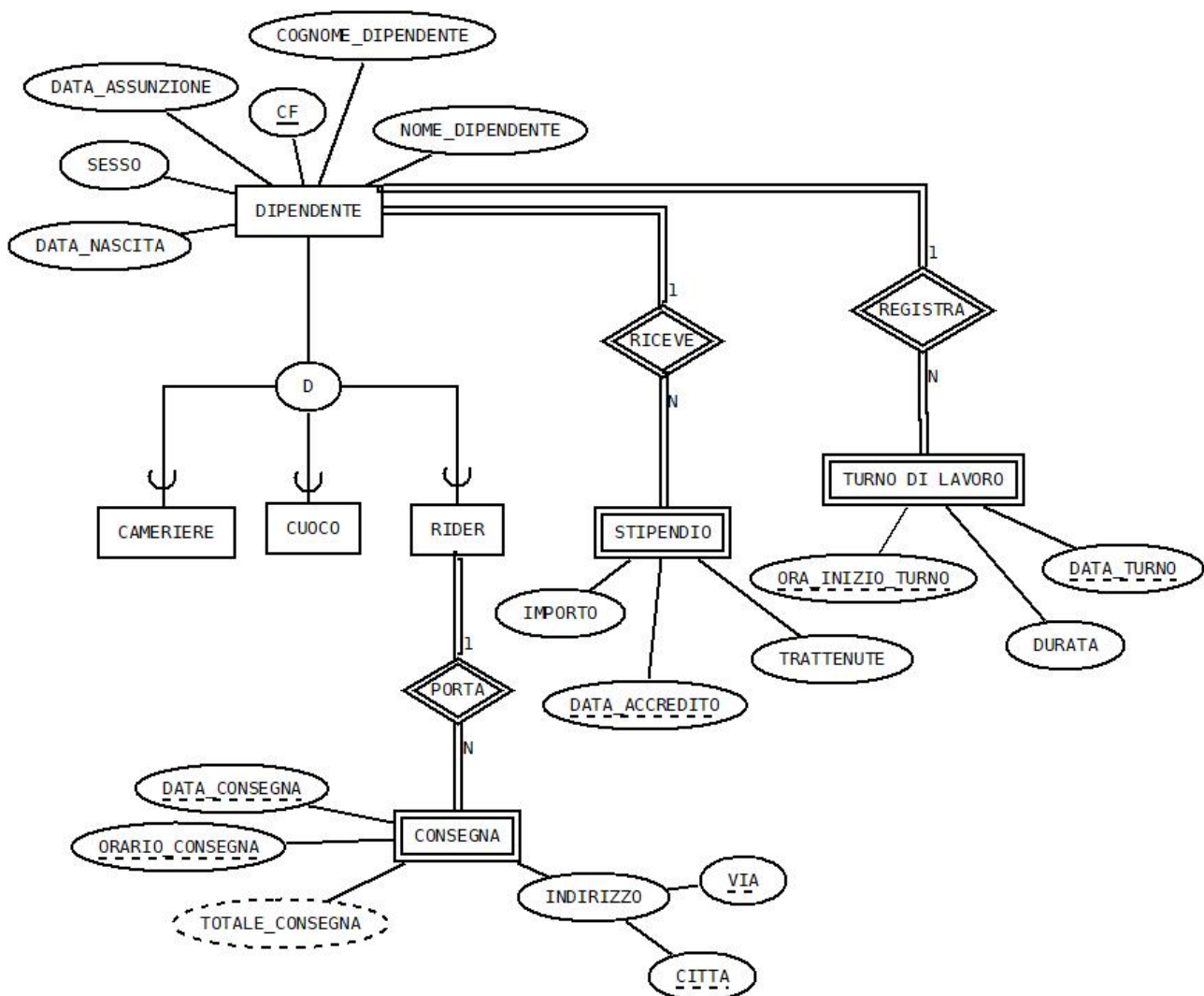


DIPENDENTE, STIPENDIO, TURNO DI LAVORO E CONSEGNA

Osservando Dipendente, notiamo che quest'ultimo può sia registrare il proprio turno di lavoro, ricevere lo stipendio e infine consegnare le ordinazioni (a domicilio) ai clienti (d'asporto). Vediamo prima lo Stipendio, che tramite l'associazione Riceve, ha una chiave primaria, la Data Accredito, che è composta con la chiave esterna del CF del Dipendente, che collega alla chiave primaria del Dipendente, poiché lo Stipendio è un'entità debole. Comunque, ha anche i suoi attributi, ovvero Importo e Trattenute.

Invece Turno di Lavoro, ha la chiave primaria la Data e l'Ora Turno, che è composta, anch'essa con la chiave esterna del CF del Dipendente, che collega alla chiave primaria del Dipendente, poiché il Turno di Lavoro è un'entità debole. Comunque, ha anche un suo attributo, ovvero Durata.

Infine, la Consegna, che tramite l'associazione Porta, ha una chiave primaria, la Data e l'Ora Consegna, Via e Città, che è composta con la chiave esterna del CF del Dipendente (per lo più Rider), che collega alla chiave della classe figlia del Dipendente e Codice Ordinazione che collega all'entità Ordinazione, poiché la Consegna è un'entità debole:



Passiamo adesso a un altro schema. Seguendo tramite l'entità Consegna, passeremo a un altro anello importante dello schema concettuale, ovvero l'Ordinazione. Questa può essere divisa (come in Clienti) in due entità figlie: Ordinazione Al Tavolo che viene collegata alla Prenotazione dei Clienti Sul Posto e Ordinazione A Domicilio che viene richiesta dal Cliente D'Asporto.

Partendo da Consegna, tramite l'associazione Composta da, ha una chiave primaria, la Data e l'Ora Consegna, Via e Città, che è composta con la chiave esterna del CF del Dipendente (Rider)

Tramite l'associazione Paga, abbiamo il collegamento con Conto che ha i suoi attributi: #Scontrino(chiave), Data e Ora Scontrino, Metodo Pagamento e due chiavi esterne Recapito_Pub che collega a Pub e ID_Ordine che collegherà a Ordinazione.

Il diagramma ER illustra la struttura dati di un sistema di gestione di un ristorante. Le entità e i loro attributi sono:

- CLIENTE**: NOME_CLIENTE, COGNOME_CLIENTE, RECAPITO_TELEFONICO.
- CONTO**: #SCONTRINO, DATA_SONTRINO, ORA_SCONTRINO, METODO_PAGAMENTO, TOTALE_IMPORTO (attributo derivato), SCONTO_10%.
- ORDINAZIONE**: COD_ORDINAZIONE.
- PRENOTAZIONE**: ORARIO_PRENOTAZIONE, DATA_PRENOTAZIONE.
- MENU**: NOME_MENU, PREZZO, QUANTITA.
- PANINO**: (entità derivata da MENU).
- BIBITA**: ALCOLICO (attributo derivato da MENU).
- CONSEGNA**: DATA_CONSEGNA, ORARIO_CONSEGNA, TOTALE_CONSEGNA (attributo derivato).

Le relazioni e i vincoli sono:

- PAGA** (1:N): Relazione tra CLIENTE e CONTO.
- EFFETTUA** (1:N): Relazione tra PRENOTAZIONE e ORDINAZIONE.
- RICHIEDE** (1:N): Relazione tra ORDINAZIONE e MENU.
- COMPOSTA DA** (1:1): Relazione tra ORDINAZIONE e CONSEGNA.
- COMPRESA** (M:N): Relazione tra ORDINAZIONE e MENU.

Le cardinalità sono indicate nei diagrammi: 1 per le relazioni PAGA, EFFETTUA, RICHIEDE e COMPOSTA DA; N per le relazioni PAGA, EFFETTUA e COMPRESA; M per la relazione COMPRESA.

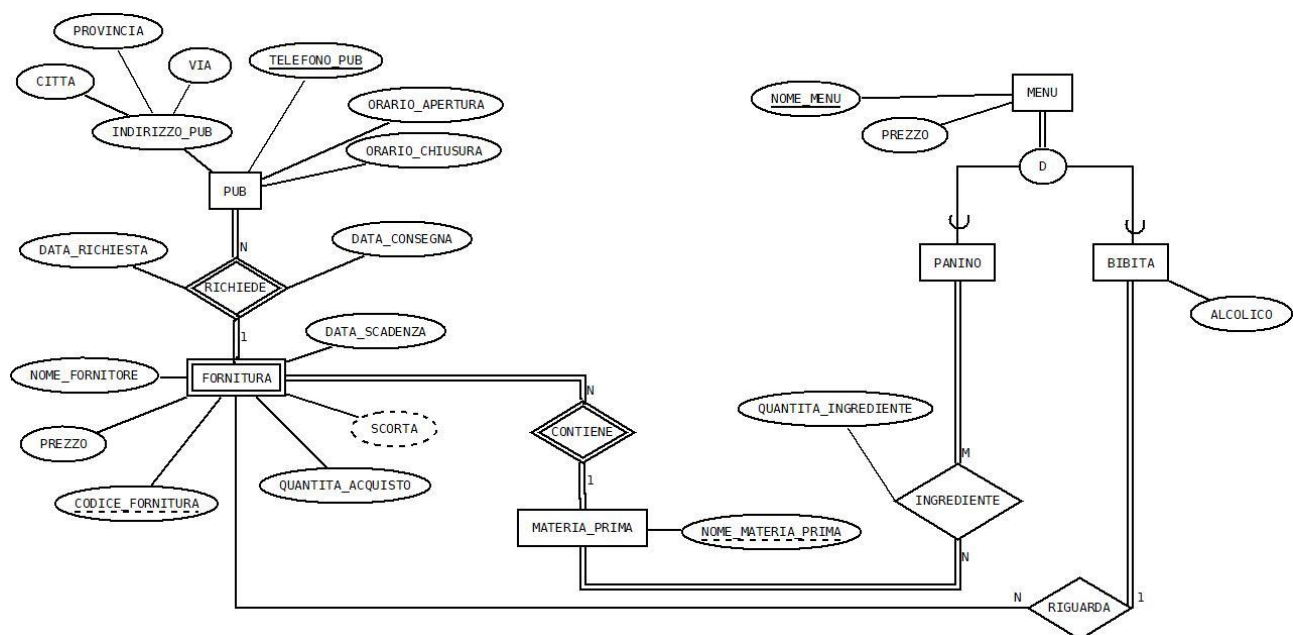
FORNITURA E MATERIA PRIMA

Infine, passiamo all'ultimo anello importante, ovvero la Fornitura. Infatti, come sappiamo, ogni pub può richiedere una fornitura quando ne ha bisogno. Dal punto di vista concettuale, la Fornitura ha la chiave primaria Codice fornitura. I suoi attributi sono: Nome Fornitore, prezzo, data scadenza e le chiavi esterne che collegano rispettivamente alle seguenti entità: Pub, materia prima e bibita (classe figlia del menù).

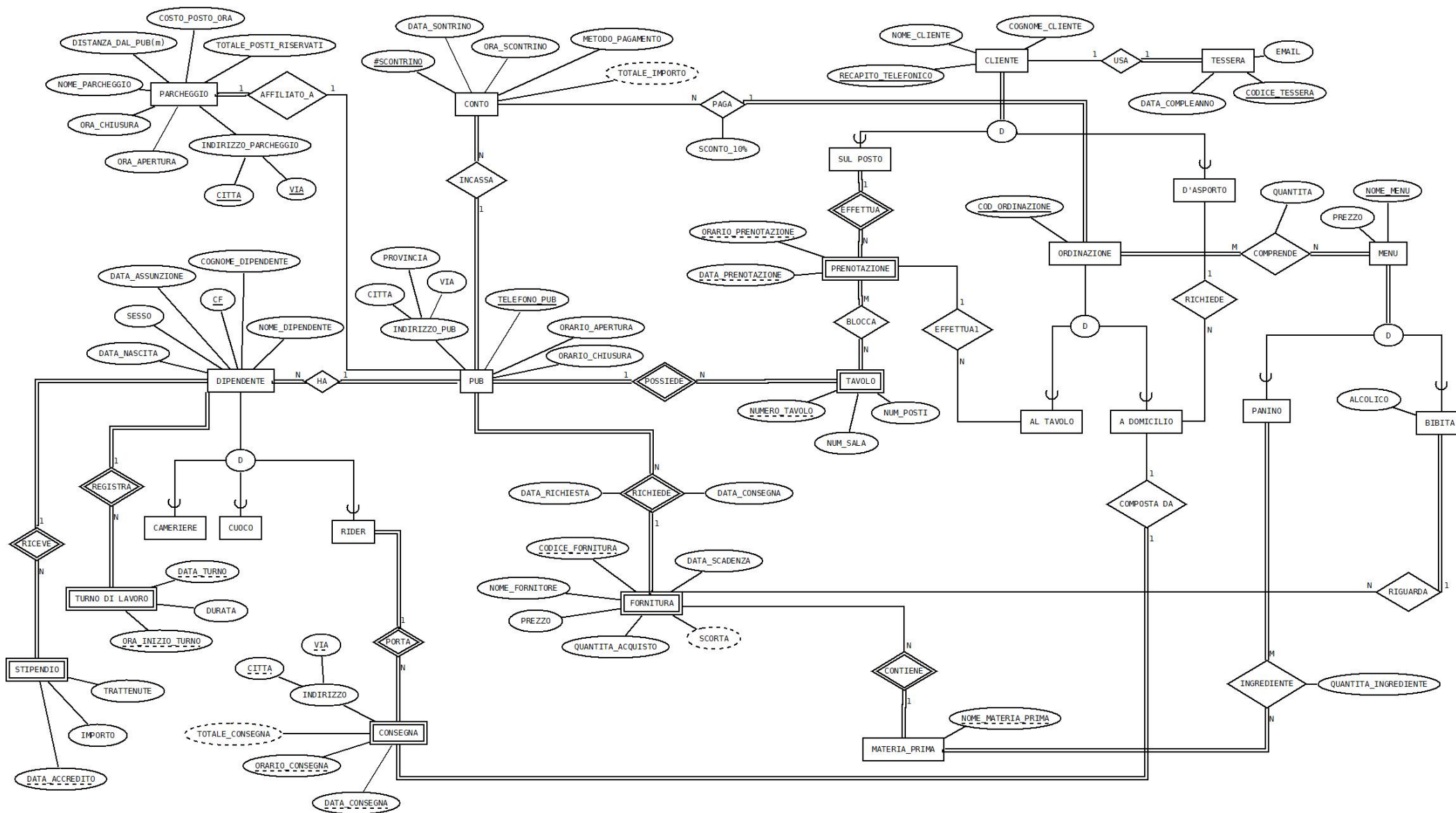
L'entità Materia Prima ha come chiave primaria il nome della materia prima che viene collegata dalla chiave esterna nell'entità fornitura, tramite l'associazione Contiene; come sappiamo una Materia Prima è l'ingrediente di un panino e tramite la associazione Ingrediente, collega la chiave esterna con la chiave della classe figlia Panino del Menu

Infine, una chiave esterna nella Fornitura si collega alla chiave della classe figlia Bibita del Menu, tramite l'associazione Riguarda.

Inoltre, nell'associazione Richiede tra due entità Pub e Fornitura (dove a collegarle ci pensa la chiave esterna nella Fornitura verso la chiave primaria del Pub) esistono due attributi chiamati data richiesta e data consegna:



Analizzato ogni anello importante, questo è lo schema EE-R completo del progetto:



2.2 Documentazione dello Schema EE-R

2.2.1 Dizionario delle Entità

Entità	Descrizione	Attributi	Chiave Primaria
<i>Cliente</i>	Il cliente che prenota l'ordinazione a domicilio oppure prenota il tavolo del pub	Recapito Telefonico, Cognome Cliente, Nome Cliente	Recapito Telefonico
<i>Cliente Sul Posto</i>	Il cliente che si reca nel pub per effettuare una prenotazione	Rec_Telefonico_SulPosto	Rec_Telefonico_SulPosto
<i>Cliente D'Asporto</i>	Il cliente che da casa effettua un'ordinazione	Rec_Telefonico_DAsporto	Rec_Telefonico_DAsporto
<i>Ordinazione</i>	L'ordinazione sia a domicilio e al tavolo fatta dal cliente	Codice Ordinazione, Recet_Tel (A Domicilio)	Codice Ordinazione
<i>Ordinazione Al Tavolo</i>	L'ordinazione fatta dal Cliente Sul Posto	Cod_Ordinazione_Tavolo, D_O_Prenota, Recpt_Tel, Tfn_Pub	Cod_Ordinazione_Tavolo
<i>Ordinazione A Domicilio</i>	L'ordinazione fatta dal Cliente D'Asporto	Cod_Ordinazione_Domicilio, Recet_Tel	Cod_Ordinazione_Domicilio
<i>Prenotazione</i>	La prenotazione del cliente sul posto per bloccare al tavolo	Data_Ora_Prenotazione, Recpt_Tel	Data_Ora_Prenotazione
<i>Tessera</i>	La tessera del cliente dove avrà lo sconto	Codice Tessera, Data_Compleanno_Cliente, Email, Recapito_Cliente	Codice Tessera
<i>Conto</i>	Il conto effettuato dopo che il cliente ha pagato l'ordinazione	ID_Scontrino, Ora_Data_Scontrino, Metodo Pagamento, ID_Ordine, Recapito_Pub, Sconto	ID_Scontrino
<i>Tavolo</i>	Il tavolo che si trova nel pub	Numero Tavolo, Num_Sala, Te_Pub, Num_Posti	Numero Tavolo
<i>Pub</i>	La sede del pub	Telefono Pub, Orario Apertura, Orario Chiusura, Via, Citta, Provincia	Telefono Pub
<i>Dipendente</i>	Il dipendente che lavora nel pub e si specializza in cameriere, cuoco o rider	CF, Nome Dipendente, Cognome Dipendente, Data Nascita, Sesso, Data Assunzione, Telefn_Pub	CF
<i>Cameriere</i>	Colui che serve i tavoli	CF_Cameriere	CF_Cameriere
<i>Cuoco</i>	Colui che prepara i piatti	CF_Cuoco	CF_Cuoco
<i>Rider</i>	Colui che porta le consegne	CF_Rider	CF_Rider

<i>Stipendio</i>	Lo stipendio inviato al dipendente	Data Accredito, Importo, Trattenute, CF_Dip	Data Accredito
<i>Turno di Lavoro</i>	Il turno di lavoro del dipendente	Data_Ora_Turno, CF_Dipendente, Durata	Data_Ora_Turno
<i>Fornitura</i>	La fornitura inviata al pub	Cod_Fornitura, Nome Fornitore, Quantità Acquisto, Prezzo, Data Richiesta, Data Consegna, Data Scadenza, Telef_Pub, Cod_Mat_Pr, N_Bibita	Cod_Fornitura
<i>Parcheggio</i>	Il parcheggio vicino al pub	Via, Citta, Costo_Posto_Ora, Ora Apertura, Ora Chiusura, Nome Parcheggio, Totale Posti Riservati, Distanza Pub, N_Pub_Associato	Via, Citta
<i>Materia Prima</i>	La materia prima dove compone il panino	Nome Materia Prima	Nome Materia Prima
<i>Menu</i>	Il menu dove si divide in bibita e panino scelto dal cliente	Nome Menu, Prezzo, Grandezza (Bibita), Alcolico (Bibita)	Nome Menu
<i>Panino</i>	Panino scelto dal cliente	N_Menu_Panino	N_Menu_Panino
<i>Bibita</i>	Bibita scelta dal cliente	N_Menu_Bibita, Alcolico	N_Menu_Bibita

2.2.2 Dizionario delle Associazioni

Associazione	Descrizione	Entità Coinvolte	Attributi	Molteplicità
<i>Usa</i>	Associa il cliente alla tessera	Cliente, Tessera	-	1:1
<i>Paga</i>	Associa il conto all'ordinazione	Conto, Ordinazione	Sconto	N:1
<i>Incassa</i>	Associa il pub al conto	Pub, Conto	-	1:N
<i>Possiede</i>	Associa il pub al tavolo	Pub, Tavolo	-	1:N
<i>Richiede</i>	Associa la fornitura al pub	Fornitura, Pub	Data Richiesta, Data Acquisto	N:1
<i>Affiliato a</i>	Associa il parcheggio al pub	Parcheggio, Pub	-	1:1
<i>Contiene</i>	Associa la fornitura alla materia prima	Fornitura, Materia Prima	Quantità_MP	N:1
<i>Ingrediente</i>	Associa la materia prima al menu (panino)	Materia Prima, Menu (Panino)	Quantità_Ingrediente	M:N
<i>Comprende</i>	Associa il menu all'ordinazione	Menu, Ordinazione	Quantità	M:N
<i>Riguarda</i>	Associa la fornitura alla bibita	Fornitura, Bibita	-	N:1
<i>Richiede_1</i>	Associa il cliente (d'asporto) all'ordinazione (a domicilio)	Cliente (D'Asporto), Ordinazione (A Domicilio)	-	1:N
<i>Effettua</i>	Associa il cliente (sul posto) alla prenotazione	Cliente (Sul Posto), Prenotazione	-	1:N
<i>Blocca</i>	Associa la prenotazione al tavolo	Prenotazione, Tavolo	-	M:N
<i>Effettua_1</i>	Associa la prenotazione all'ordinazione (al tavolo)	Prenotazione, Ordinazione (Al Tavolo)	-	1:N
<i>Riceve</i>	Associa il dipendente allo stipendio	Dipendente, Stipendio	-	1:N
<i>Ha</i>	Associa il pub al dipendente	Pub, Dipendente	-	1:N
<i>Registra</i>	Associa il dipendente al turno di lavoro	Dipendente, Turno di Lavoro	-	1:N
<i>Porta</i>	Associa il dipendente (rider) alla consegna	Dipendente (Rider), Consegna	-	1:N
<i>Composta_da</i>	Associa la consegna all'ordinazione (a domicilio)	Consegna, Ordinazione (A Domicilio)	-	1:1

2.3 Regole di Vincolo

2.3.1 Vincoli d'Integrità Statici

VS 1 → Il sesso del dipendente deve essere Maschio (M), Femmina (F) o Non Binario (NB);

VS 2 → L'importo dello stipendio del dipendente non deve essere inferiore a 800;

VS 3 → La durata del turno di lavoro del dipendente non deve essere maggiore a 8 ore;

VS 4 → I codici fiscali che compaiono nel database devono essere necessariamente di 16 caratteri;

VS 5 → Il metodo di pagamento nel conto del cliente deve essere CONTANTI o CARTA;

VS 6 → L'attributo Alcolico contenuto in bibita sarà SI oppure NO

2.3.2 Vincoli d'Integrità Dinamici

VD 1 → Non è possibile aggiungere un altro dipendente se il numero per il Pub che lo assumerebbe è al massimo (12);

VD 2 → Il pub non può richiedere la fornitura dove la data di scadenza è prossima a 1 settimana dalla data di richiesta;

VD 3 → Una prenotazione non può bloccare più di 5 tavoli in un determinato giorno e in un determinato pub;

VD 4 → Il dipendente appena assunto, dunque senza abbastanza esperienza, non può avere uno stipendio base(cioè senza considerare eventuali bonus) troppo alto (1600 è il massimo);

VD 5 → Il dipendente non può lavorare 7 giorni su 7;

VD 6 → Il dipendente non può lavorare più di 190 ore al mese;

VD 7 → Il dipendente non può effettuare due turni di lavoro nello stesso giorno;

VD 8 → Quando effettua una prenotazione, il cliente può farlo esclusivamente o per il giorno stesso oppure per i 2 giorni successivi al massimo;

VD 9 → La fine del turno di un dipendente non deve superare un orario prefissato (03:00 in questo caso)

3. Progettazione Logica

3.1 Tavola dei volumi a regime

La seguente tavola è stimata in base al volume che il database potrebbe occupare nel suo ciclo di vita. Poiché stiamo parlando di una catena ristorativa, avremo volumi molto alti, ipotizzando una ciclo di vita almeno 5 anni.

Concetto	Tipo	Volume
<i>Cliente</i>	E	10000
<i>Usa</i>	A	10000
<i>Tessera</i>	E	10000
<i>Pub</i>	E	100
<i>Ha</i>	A	1200
<i>Dipendente</i>	E	1200
<i>Cliente Sul Posto</i>	EF	5000
<i>Effettua</i>	A	10000
<i>Prenotazione</i>	ED	10000
<i>Cliente D'Asporto</i>	EF	5000
<i>Richiede</i>	A	10000
<i>Ordinazione A Domicilio</i>	EF	10000
<i>Effettua</i>	A	10000
<i>Ordinazione Al Tavolo</i>	EF	10000
<i>Incassa</i>	A	20000
<i>Conto</i>	E	20000
<i>Possiede</i>	A	20000
<i>Tavolo</i>	ED	20000
<i>Affiliato_a</i>	A	100
<i>Parcheggio</i>	E	100
<i>Rider</i>	EF	200
<i>Porta</i>	A	10000
<i>Consegna</i>	ED	10000
<i>Ordinazione</i>	E	100000
<i>Comprende</i>	A	100000
<i>Menu</i>	E	20
<i>Panino</i>	EF	10
<i>Ingrediente</i>	A	50
<i>Materia Prima</i>	E	50
<i>Fornitura</i>	ED	200000
<i>Contiene</i>	A	100000
<i>Richiede</i>	A	200000
<i>Riguarda</i>	A	100000
<i>Registra</i>	A	36500
<i>Turno_di_Lavoro</i>	ED	42600
<i>Riceve</i>	A	7200
<i>Stipendio</i>	ED	7200
<i>Composta_da</i>	A	30000

<i>Blocca</i>	A	10000
<i>Paga</i>	A	20000
<i>Bibita</i>	EF	10

Legenda: E = entità, A = associazioni, ED = entità deboli, EF = entità figlie

3.2 Tavola degli Utenti

Anche qui avremo volumi molto alti.

Utente	Categoria	Volume	Privilegi	Tipo
<i>Amministratore</i>	Admin	1	Tutte le tabelle	A
<i>Cliente</i>	Utente Comune	100000	R-V: Menu_Panini, Menu_Bibite, Ingre_Panini, Ingredienti2 R/W-P: Park	NA
<i>Dipendente</i>	Utente Comune	1200	R-V: Conti_ord R/W-P: Delete_Info_Cliente, Sconto_Tes	NA
<i>Responsabile_Pub</i>	Utente Comune	100	R/W-P: Bonus_Mese R/W-P: Aum_Stp	NA
<i>Gestore_Forniture</i>	Utente Comune	200	R-V: Riepilogo_Vendite_MP, Riepilogo_Forniture_MP, Riepilogo_Vendite_Bibite, Riepilogo_Forniture_Bibite; R/W-P: Scorta	NA

3.3 Tavola delle Operazioni

Si ricorda che abbiamo una catena di Pub, quindi anche qui avremo volumi elevati.

Operazione	Tipo	Frequenza	Periodo	Read/Write
<i>Inserimento Cliente</i>	Batch	100	Giorno	W
<i>Inserimento Cliente Sul Posto</i>	Batch	50	Giorno	W
<i>Inserimento Cliente D'Asporto</i>	Batch	50	Giorno	W
<i>Inserimento Tessera</i>	Batch	30	Mese	W
<i>Inserimento Conto</i>	Batch	1000	Giorno	W
<i>Inserimento Dipendente</i>	Batch	20	Mese	W
<i>Inserimento Cameriere</i>	Batch	10	Mese	W
<i>Inserimento Cuoco</i>	Batch	5	Mese	W
<i>Inserimento Rider</i>	Batch	5	Mese	W
<i>Inserimento Pub</i>	Batch	2	Anno	W
<i>Inserimento Stipendio</i>	Batch	120	Mese	W
<i>Inserimento Turno di Lavoro</i>	Batch	100	Giorno	W
<i>Inserimento Ordinazione</i>	Batch	1000	Giorno	W
<i>Inserimento Ordinazione Al Tavolo</i>	Batch	500	Giorno	W
<i>Inserimento Ordinazione A Domicilio</i>	Batch	500	Giorno	W
<i>Inserimento Consegna</i>	Batch	500	Giorno	W
<i>Inserimento Prenotazione</i>	Batch	500	Giorno	W
<i>Inserimento Tavolo</i>	Batch	40	Anno	W
<i>Inserimento Menù</i>	Batch	5	Anno	W
<i>Inserimento Panino</i>	Batch	3	Anno	W
<i>Inserimento Bibita</i>	Batch	2	Anno	W
<i>Inserimento Materia Prima</i>	Batch	10	Anno	W
<i>Inserimento Fornitura</i>	Batch	20	Mese	W
<i>Inserimento Parcheggio</i>	Batch	10	Anno	W
<i>Menu_Panini</i>	Batch	1000	Giorno	R
<i>Menu_Bibite</i>	Batch	1000	Giorno	R
<i>Ingre_Panini</i>	Batch	1000	Giorno	R
<i>Ingredienti2</i>	Batch	1000	Giorno	R
<i>Riepilogo_Forniture_Mp</i>	Batch	10	Settimana	R
<i>Riepilogo_Forniture_Bibite</i>	Batch	10	Settimana	R
<i>Riepilogo_Vendite_Mp</i>	Batch	10	Settimana	R

<i>Riepilogo_Vendite_Bibite</i>	Batch	10	Settimana	R
<i>Delete_Info_Cliente</i>	Batch	1	Settimana	R-W
<i>Aum_Stp</i>	Batch	100	Mese	R-W
<i>Sconto_Tes</i>	Batch	1000	Giorno	R-W
<i>Park</i>	Batch	500	Giorno	R
<i>Scorta</i>	Batch	100	Settimana	R
<i>Bonus_mese</i>	Batch	100	Mese	R-W

3.4 Tavole dei descrittori delle Operazioni

VISTE

Nome	Menu_Panini
<i>Usa</i>	Menu, Menu(Panino)
<i>Esito</i>	View Created – SELECT * FROM MENU_PANINI
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	Menù del panino esistente
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vista sul menù dei panini per il cliente

Nome	Mebu_Bibite
<i>Usa</i>	Menu, Menu(Bibita)
<i>Esito</i>	View Created – SELECT * FROM MENU_BIBITE
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	Menù della bibita esistente
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vista sul menù delle bibite per il cliente

Nome	Conti_ord
<i>Usa</i>	Conto, Comprende, Menu
<i>Esito</i>	View Created – SELECT * FROM CONTI_ORD
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere gli ordini di oggi

Nome	Ingre_panini
<i>Usa</i>	Menu, Panino, Ingrediente
<i>Esito</i>	View Created – SELECT * FROM INGRE_PANINI
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere gli ingredienti dei panini

Nome	Ingredienti2
<i>Usa</i>	Ingrediente, Panino
<i>Esito</i>	View Created – SELECT * FROM INGREDIENTI2
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere gli ingredienti dei panini

Nome	Riepilogo_forniture_mp
<i>Usa</i>	Pub, Fornitura, Materia_prima
<i>Esito</i>	View Created – SELECT * FROM RIEPILOGO_FORNITURE_MP
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere le forniture acquistate di materie prime

Nome	Riepilogo_forniture_bibite
<i>Usa</i>	Pub, Fornitura, Bibita
<i>Esito</i>	View Created – SELECT * FROM RIEPILOGO_FORNITURE_BIBITE
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere le forniture acquistate di Bibite

Nome	Riepilogo_vendite_mp
<i>Usa</i>	Conto, Ordinazione, Comprende,Panino,Ingrediente,Materia_prima
<i>Esito</i>	View Created – SELECT * FROM RIEPILOGO_VENDITE_MP
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere il consumo di materie prime

Nome	Riepilogo_vendite_Bibite
<i>Usa</i>	Conto, Ordinazione, Comprende, Bibita
<i>Esito</i>	View Created – SELECT * FROM RIEPILOGO_VENDITE_BIBITE
<i>Errori</i>	Join non collegate
<i>Pre-Condizioni</i>	-
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Vedere il consumo di materie prime

PROCEDURE

Nome	Delete_Info_Cliente
<i>Usa</i>	Conto, Ordinazione, Ordinazione (A Domicilio), Consegna
<i>Esito</i>	1) Eliminazione informazioni cliente confermata 2) Cliente ha diritto di rimanere nel sistema
<i>Errori</i>	1) Recapito telefonico non valido 2) Ordinazione non valida
<i>Pre-Condizioni</i>	1) Recapito telefonico esistente 2) Ordinazione esistente 3) Data e ora dello scontrino esistente (conto)
<i>Post-Condizioni</i>	Data e ora dello scontrino vecchio
<i>Scopo</i>	Eliminare le informazioni del cliente se quest'ultimo non effettua ordinazioni da più di un anno

Nome	Aum_Stp
<i>Usa</i>	Stipendio, Dipendente, Turno di Lavoro
<i>Esito</i>	1) Modifica dello stipendio al dipendente 2) Stipendio troppo alto/Minore durata del turno lavorativo
<i>Errori</i>	CF non valido
<i>Pre-Condizioni</i>	1) Stipendio esistente 2) CF esistente 3) Turno di lavoro esistente
<i>Post-Condizioni</i>	1) Stipendio minore di 1500 2) Ore lavorative mensili > 160
<i>Scopo</i>	Aumentare lo stipendio del dipendente se quest'ultimo guadagna meno di 1500 euro e ha lavorato più di 160 ore nell' ultimo mese

Nome	Sconto_Tes
<i>Usa</i>	Tessera, Cliente, Cliente (Sul Posto), Prenotazione, Ordinazione (Al Tavolo), Ordinazione, Conto
<i>Esito</i>	1) Viene applicato lo sconto (da ND a SI) 2) Non viene applicato lo sconto (da ND a NO)
<i>Errori</i>	1) Recapito telefonico non valido 2) Ordinazione non valida
<i>Pre-Condizioni</i>	1) Tessera esistente 2) Recapito telefonico esistente 3) Ordinazione esistente
<i>Post-Condizioni</i>	Lo sconto deve essere pre-impostato a ND
<i>Scopo</i>	Il cliente quando pagherà il conto ,se possiede una tessera, avrà lo sconto del 10%

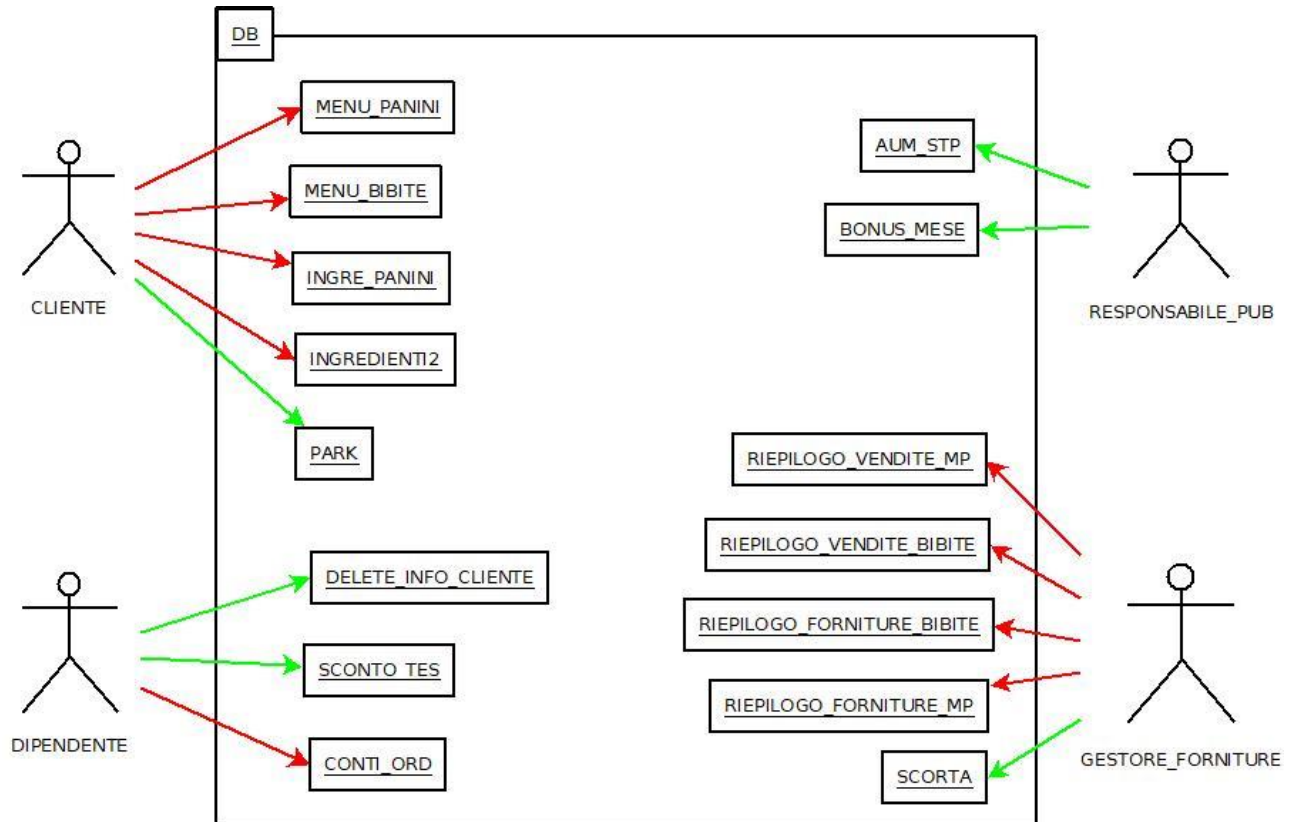
Nome	Park
<i>Usa</i>	Parcheggio, Pub
<i>Esito</i>	1) Il pub non ha un parcheggio 2) Il parcheggio più vicino al pub esiste 3) Il pub selezionato non esiste
<i>Errori</i>	Telefono del Pub non valido
<i>Pre-Condizioni</i>	1) Parcheggio esistente 2) Pub esistente
<i>Post-Condizioni</i>	Non per forza il parcheggio esiste
<i>Scopo</i>	Far sapere al cliente se il pub è dotato di un parcheggio affiliato o meno

Nome	Scorta
<i>Usa</i>	Fornitura, Materia Prima, Bibita, Pub, Panino, Ordinazione, Conto, Comprende
<i>Esito</i>	1)Mostra la scorta della bibita o panino selezionato in un determinato pub
<i>Errori</i>	Telefono del Pub non valido o Prodotto non esistente
<i>Pre-Condizioni</i>	1)Prodotto esistente 2)Pub esistente
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Far sapere al gestore delle scorte se il pub ha bisogno di fare un ordine

Nome	Bonus_mese
<i>Usa</i>	Conto, Ordinazione, Comprende, Menu, Stipendio, Dipendente
<i>Esito</i>	1)il pub ha raggiunto l'obiettivo ed aumento lo stipendio dei suoi dipendenti 2)Il pub non ha raggiunto l'obiettivo
<i>Errori</i>	Telefono del Pub non valido
<i>Pre-Condizioni</i>	1) Pub esistente
<i>Post-Condizioni</i>	-
<i>Scopo</i>	Dare un bonus ai dipendenti se si realizza un obiettivo di vendite

3.5 Use Case (Casi d'Uso)

Questi sono gli Use Case, ovvero i casi d'uso. Le frecce in **rosso** vuol dire che visualizza le **viste**, mentre le frecce in **verde** vuol dire che eseguono le **procedure**:



4. Modello Relazionale

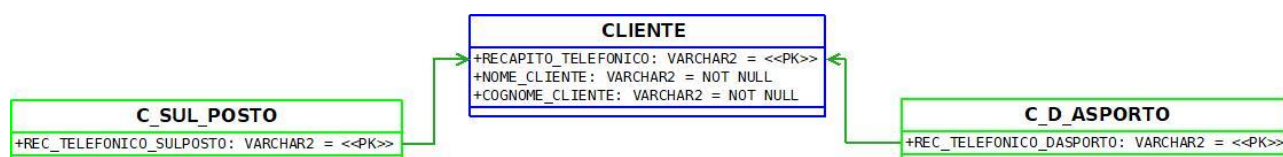
A seguito della costruzione del modello EE-R, si procede con la traduzione dello stesso nel modello logico Relazionale.

4.1 Dal Modello EE-R al Relazionale

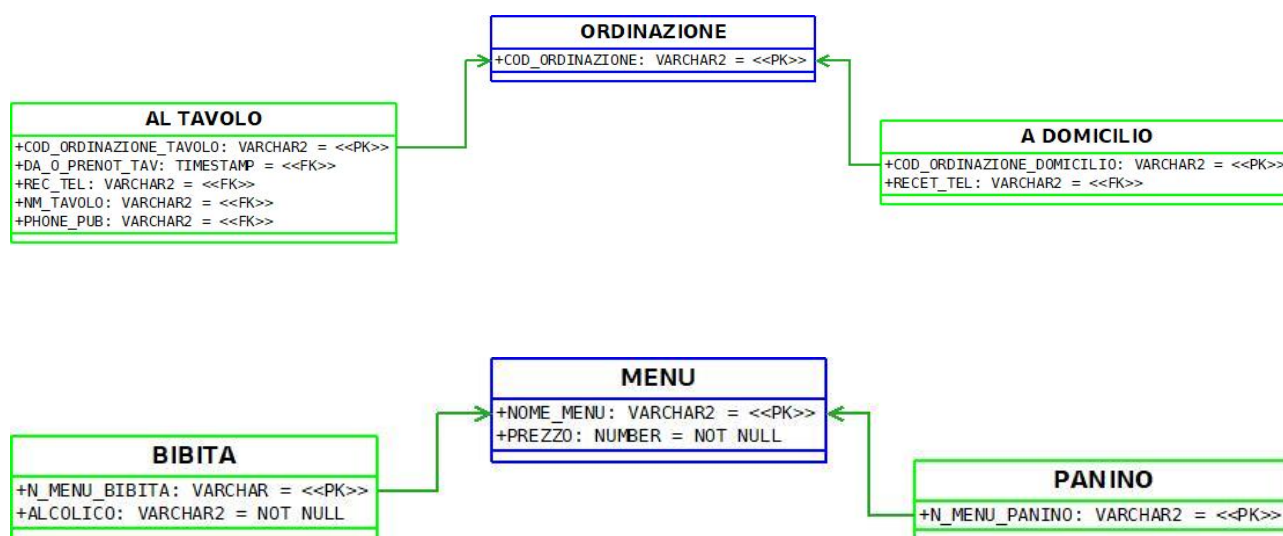
SPECIALIZZAZIONI

Vediamo la traduzione di Cliente e delle sue specializzazioni Sul Posto e D'Asporto. Tutte le specializzazioni si collegano all'entità madre tramite le chiavi esterne che in questo caso sono anche chiave primarie.

Su Cliente Sul Posto e D'Asporto, abbiamo, rispettivamente, la chiave primaria che può essere anche la chiave esterna di Recapito Telefonico Sul Posto e Recapito Telefonico D'Asporto, che, come vedremo in seguito sono collegate alle chiavi primarie di Prenotazione e Ordinazione A Domicilio.



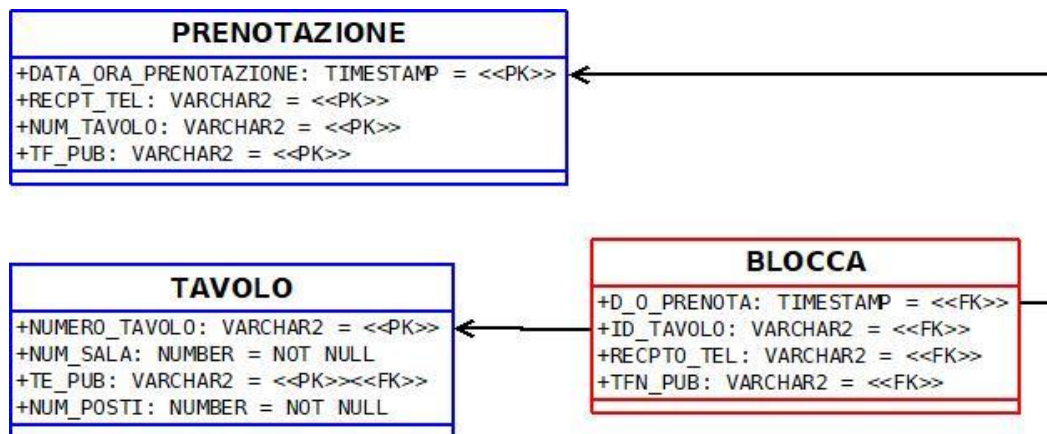
Lo stesso discorso fatto sopra vale anche per Ordinazione e le sue specializzazioni in Ordinazione Al Tavolo e A Domicilio, che per Menù e le sue specializzazioni in Panino e Bibita:



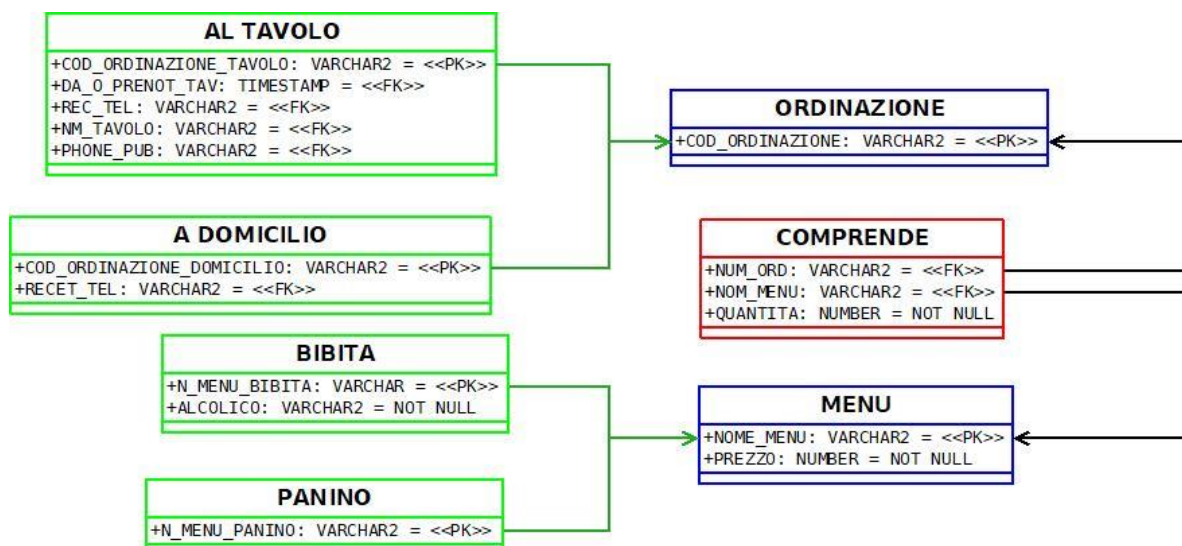
ASSOCIAZIONI M:N

Passiamo ora alla traduzione delle varie associazioni delle singole entità, per la precisione, le associazioni dove troviamo la molteplicità N:M

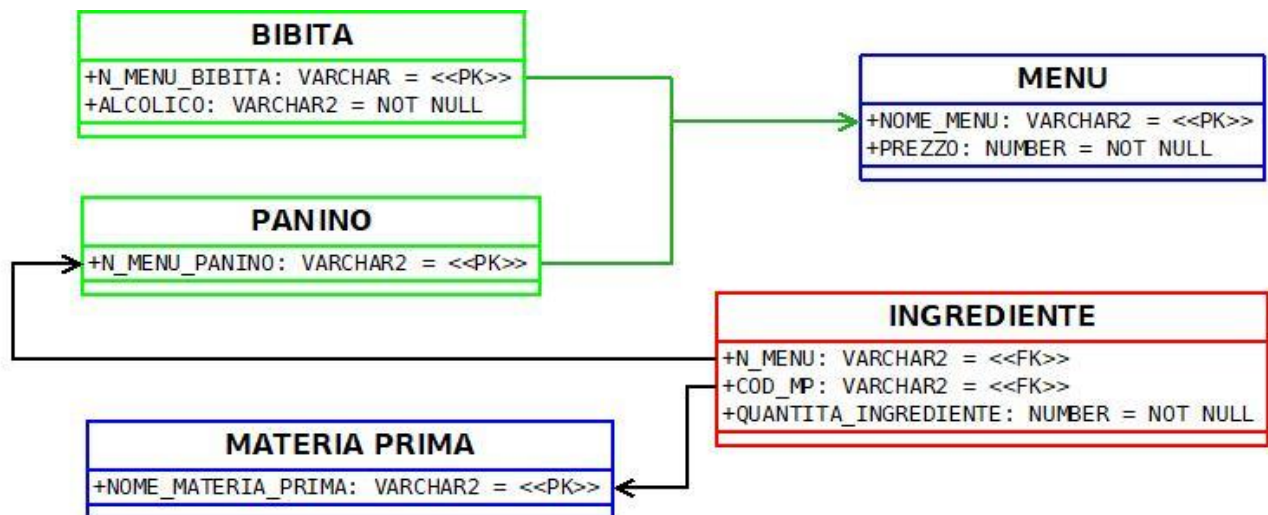
- L'associazione Blocca (N:M) è stata tradotta con l'aggiunta di una nuova tabella, le sue chiavi esterne sono D_O_Prenota, che collegherà alla chiave primaria Data_Ora_Prenotazione di Prenotazione e ID_Tavolo, che collegherà alla chiave primaria Numero_Tavolo di Tavolo:



- L'associazione Comprende (N:M) è stata tradotta con l'aggiunta di una nuova tabella, le sue chiavi esterne sono Num_Ord, che collegherà alla chiave primaria Cod_Ordinazione di Ordinazione e Nom_Menu, che collegherà alla chiave primaria Nome_Menu di Menu:



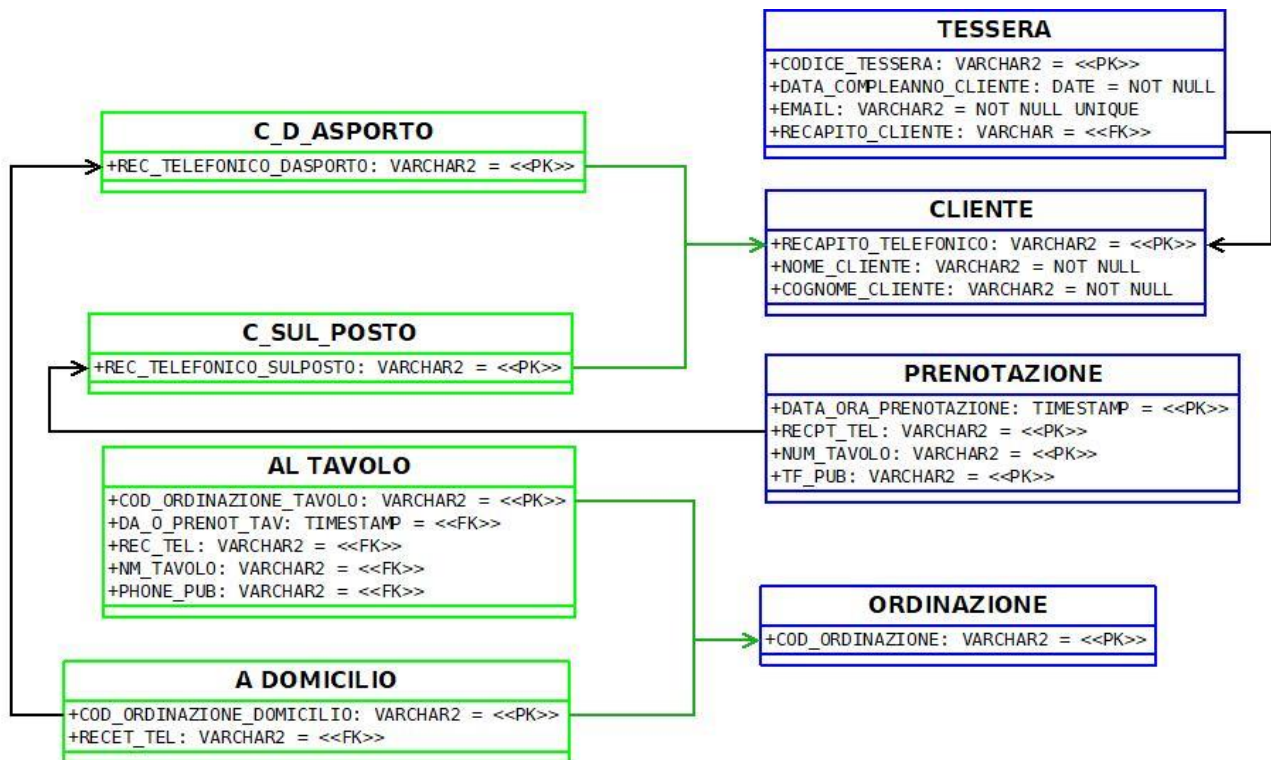
- L'associazione Ingredienti (N:M) è stata tradotta con l'aggiunta di una nuova tabella, le sue chiavi esterne sono N_Menu, che collegherà alla chiave primaria della classe figlia N_Menu_Panino di Panino, che si collegherà di conseguenza alla classe madre Menù e Cod_MP, che collegherà alla chiave primaria Nome_Materia_Prima di Materia Prima:



ASSOCIAZIONI TRA ENTITA

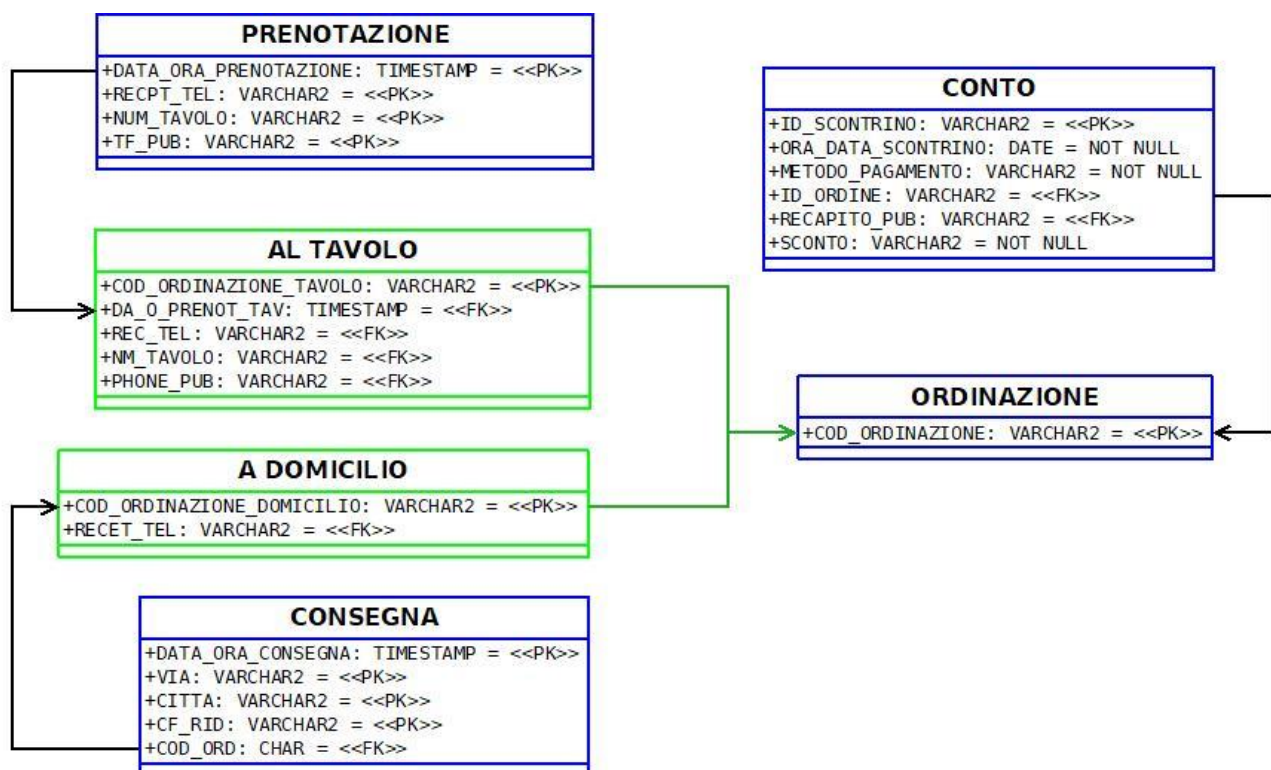
Passiamo ora alla traduzione delle varie associazioni delle singole entità, per la precisione, le associazioni che hanno la molteplicità 1:1 o 1:N, partendo da **Cliente**. Se osserviamo lo schema EE-R, tutte le **Tessere** afferiscono a un solo Cliente e tutti i clienti possono ricevere una sola tessera. Inoltre, un Cliente, quello **D'Asporto**, può richiedere più **Ordinazioni (A Domicilio)**, mentre quello **Sul Posto**, può effettuare più **Prenotazioni**:

- L'associazione Usa (1:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Cliente su Tessera, poiché non tutti i clienti possono ricevere una tessera;
- L'associazione Richiede (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Cliente (D'Asporto) su Ordinazione (A Domicilio), poiché un singolo cliente possono richiedere più ordinazioni;
- L'associazione Effettua (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Cliente (Sul Posto) su Prenotazione, poiché un singolo cliente possono effettuare più prenotazioni:



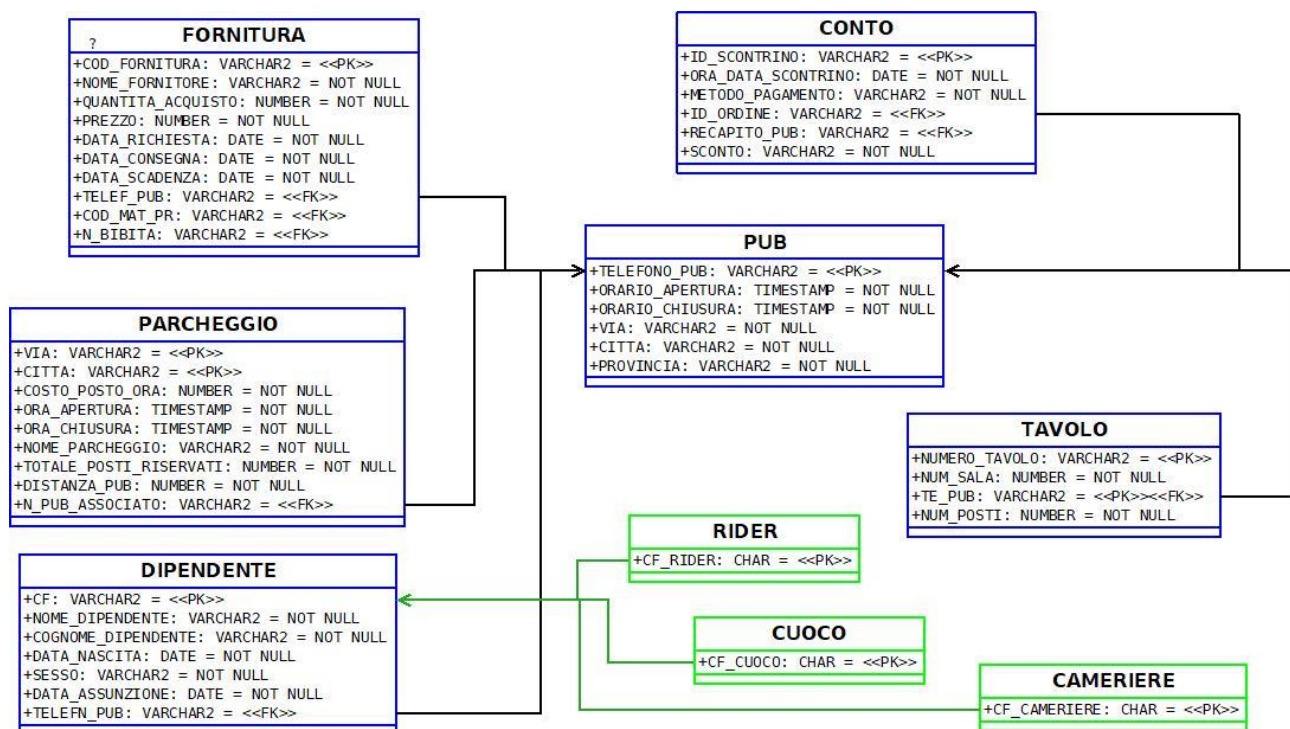
Passiamo ora all'**Ordinazione**. Se vediamo nello schema EE-R, un'Ordinazione, quella **A Domicilio**, va a comporre una **Consegna** alla volta, invece quella **Al Tavolo**, è effettuata da una **Prenotazione** del cliente e inoltre, tutte le ordinazioni vengono pagate su più conti:

- L'associazione Compsta_da (1:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Ordinazione (A Domicilio) su Consegna, poiché una consegna compone un'ordinazione;
- L'associazione Riguarda_2 (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Prenotazione su Ordinazione (Al Tavolo), poiché una prenotazione può riguardare più ordinazioni;
- L'associazione Riguarda (N:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Ordinazione sul Conto, poiché più conti vanno a coinvolgere un'ordinazione, con l'aggiunta dell'attributo Sconto su conto poiché va ad applicare sull'ordinazione;



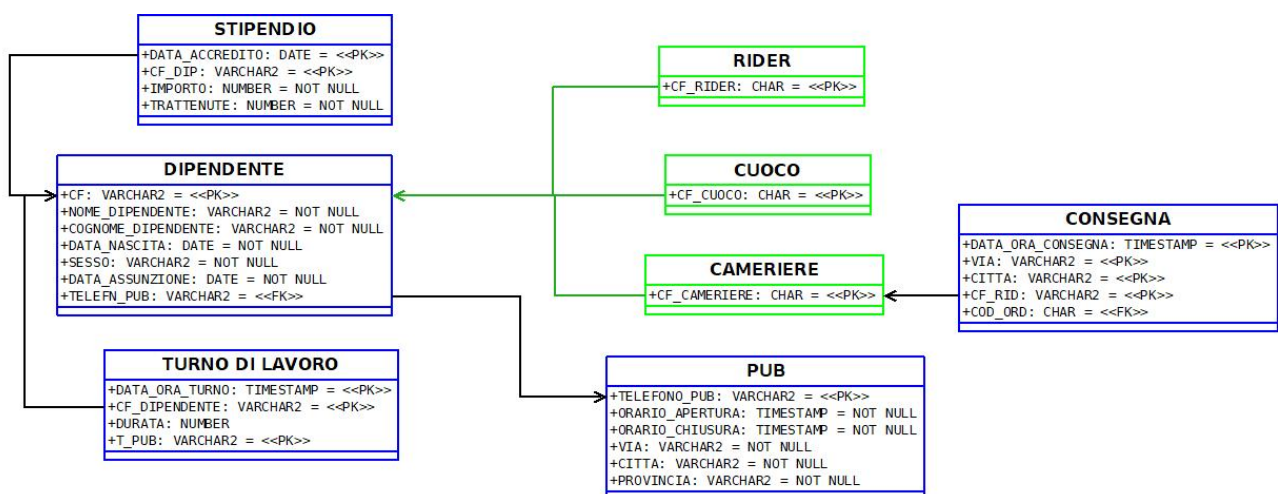
Passiamo ora al **Pub**. Se vediamo nello schema EE-R, un Pub, può possedere più **Tavoli**; ogni Pub incassa più **Conti**; va a richiedere più **Forniture**; Naturalmente ha molti **Dipendenti** che si specializzano in **Cuoco**, **Cameriere** o **Rider**; Infine può affiliare a un singolo **Parcheggio**:

- L'associazione Possiede (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub su Tavolo, poiché più tavoli appartengono ad un singolo pub;
- L'associazione Incassa (N:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub sul Conto, poiché più conti dono incassati da un singolo pub;
- L'associazione Richiede (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub sulla Fornitura, poiché più forniture vengono inviate al pub, con l'aggiunta degli attributi Data Consegna e Data Richiesta sulla fornitura poiché è utile sapere la data della richiesta e della consegna;
- L'associazione Ha_2 (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub sul Dipendente, poiché più dipendenti (Cameriere, Cuoco o Rider) vanno lavorano per un pub;
- L'associazione Affiliato_a (1:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub sul Parcheggio, poiché non tutti i pub sono affiliati ad un parcheggio:



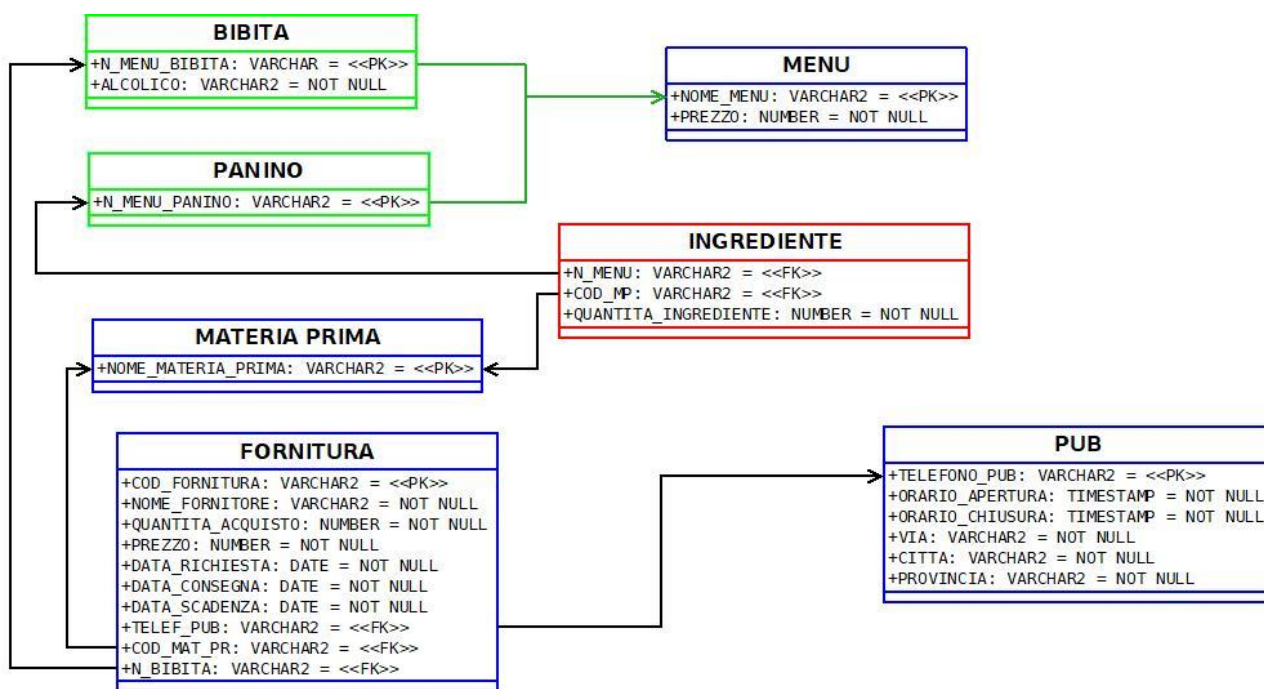
Passiamo ora al **Dipendente**. Se vediamo nello schema EE-R, un Dipendente, può registrare dei **Turni di Lavoro**; ogni Dipendente riceve i propri **Stipendi** e ogni Rider, porta più **Consegne**. Infine, ogni Dipendente lavora per un **Pub**:

- L'associazione Registra (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Dipendente sul Turno di Lavoro, poiché molti turni di lavoro sono registrati da un singolo dipendente;
- L'associazione Riceve (N:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Dipendente sullo Stipendio, poiché più stipendi vengono inviati ad un singolo dipendente;
- L'associazione Porta (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Dipendente sulla Consegna, poiché molte consegne sono portate dal singolo dipendente, ovvero il rider;
- L'associazione Ha_2 (N:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub sul Dipendente, poiché più dipendenti (Cameriere, Cuoco o Rider) lavorano per il pub:

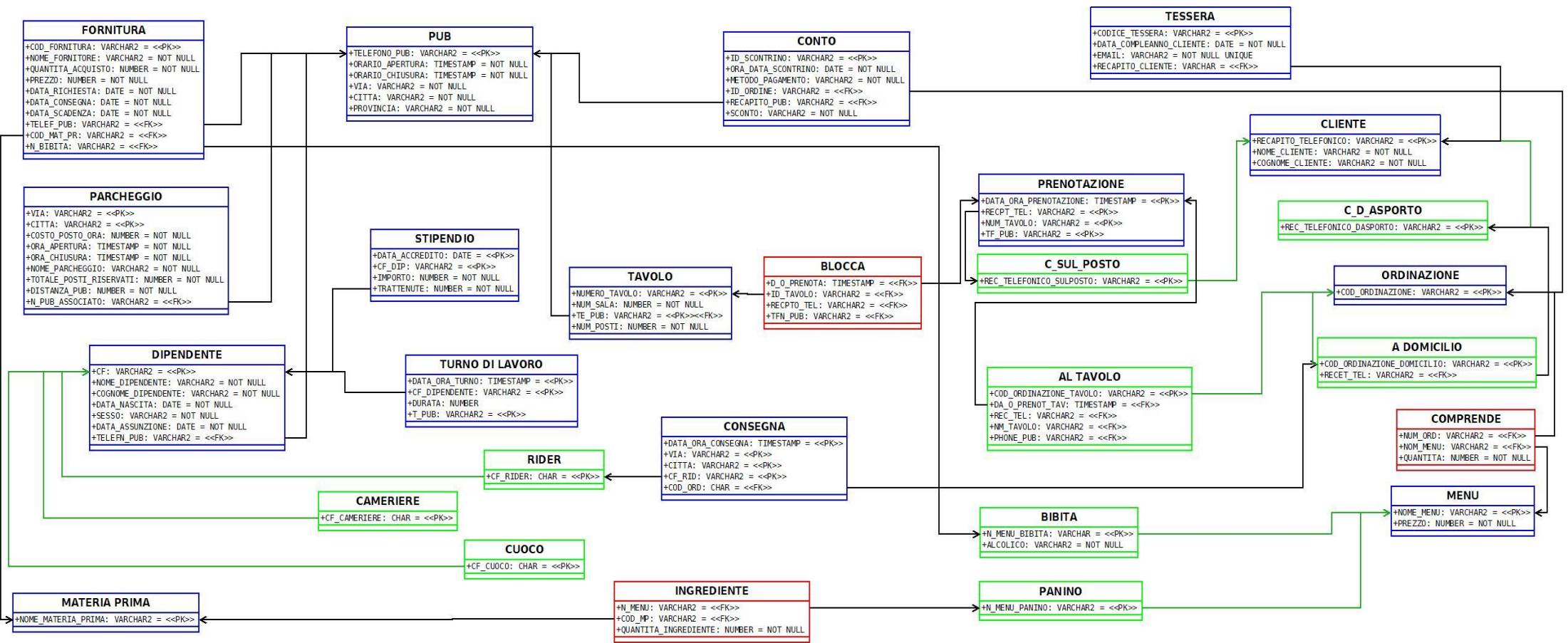


Infine, passiamo sia al **Menù** che alla **Fornitura**. Se vediamo nello schema EE-R, un Menù, ovvero la **Bibita**, può essere contenuta in più Forniture; ogni Fornitura viene richiesto da un singolo **Pub**. Anche ogni **Materia Prima** è contenuta in più Forniture. Naturalmente ogni Materia Prima va comporre come ogni ingrediente ogni **Panino** del Menù:

- L'associazione Riguarda (N:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Bibita del Menù sulla Fornitura, poiché molte forniture contengono una bibita;
- L'associazione Richiede (1:N) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Pub sulla Fornitura, poiché più forniture vengono inviate al pub, con l'aggiunta degli attributi Data Consegna e Data Richiesta sulla fornitura;
- L'associazione Contiene (N:1) è stata tradotta con l'aggiunta di un vincolo di chiave esterna, della chiave primaria di Materia Prima sulla Fornitura, poiché molte forniture contengono una materia prima;
- L'associazione Ingrediente (N:M) è stata tradotta con l'aggiunta di una nuova tabella, le sue chiavi esterne sono N_Menu, che collegherà alla chiave primaria della classe figlia N_Menu_Panino di Panino, che si collegherà di conseguenza alla classe madre Menù e Cod_MP, che collegherà alla chiave primaria Nome_Materia_Prima di Materia Prima:



Esaminata ogni parte del modello, questo è il Modello Relazionale del progetto al completo:



4.2 Elenco delle Relazioni del Modello Logico Relazionale

Vi riporto ora una serie di entità e associazioni con i loro rispettivi attributi, chiavi primarie ed esterne:

Cliente → (Recapito_Telefonico, Nome_Cliente, Cognome_Cliente);

Tessera → (Codice_Tessera, Data_Compleanno_Cliente, Email, *Recapito_Cliente*);

C_Sul_Posto → (Rec_Telefonico_SulPosto);

C_D_Asporto → (Rec_Telefonico_DAsporto);

Pub → (Telefono_Pub, Orario_Apertura, Orario_Chiusura, Via, Citta, Provincia);

Dipendente → (CF, Nome_Dipendente, Cognome_Dipendente, Data_Nascita, Data_Assunzione, Sesso, *Telefn_Pub*);

Cameriere → (CF_Cameriere);

Cuoco → (CF_Cuoco);

Rider → (CF_Rider);

Stipendio → (Data_Accredito, CF_Dip, Importo, Trattenute);

Tavolo → (Numero_Tavolo, Te_Pub, Num_Sala, Num_Posti);

Turno_di_Lavoro → (Data_Ora_Turno, CF_Dipendente, Durata);

Prenotazione → (Data_Ora_Prenotazione, Recpt_Tel, Num_Tavolo, Tf_Pub);

Blocca → (ID_Tavolo, D_O_Prenota, Recpto_Tel, Tfn_Pub);

Ordinazione → (Cod_Ordinazione);

O_Al_Tavolo → (Cod_Ordinazione_Tavolo, Da_O_Prenot_Tav, Rec_Tel, Nm_Tavolo, Phone_Pub);

O_A_Domicilio → (Cod_Ordinazione_Domicilio, Recet_Tel);

Conto → (ID_Scontrino, Ora_Data_Scontrino, Metodo_Pagamento, ID_Ordine, Recapito_Pub, Sconto);

Consegna → (Data_Ora_Consegna, Via, Citta, CF_Rid, Cod_Ord);

Menu → (Nome_Menu, Prezzo);

Comprende → (Num_Ord, Nom_Menu, Quantita);

Bibita → (N_Menu_Bibita, Alcolico);

Panino → (N_Menu_Panino);

Materia_Prima → (Nome_Materia_Prima);

Fornitura → (Cod_Fornitura, Telef_Pub, Cod_Mat_Pr, N_Bibita, Nome_Fornitore, Prezzo, Quantita_Acquisto, Data_Richiesta, Data_Consegna, Data_Scadenza);

Ingrediente → (*N_Menu*, *Cod_Mp*, *Quantita_Ingrediente*);

Parcheggio → (*Via*, *Citta*, *Costo_Posto_Ora*, *Ora_Apertura*, *Ora_Chiusura*, *Nome_Parcheggio*, *Totale_Posti_Riservati*, *Distanza_Pub*, *N_Pub_Associato*);

4.3 Verifica delle forme normali

A questo punto verifico che le relazioni rispettino le forme normali:

4.3.1 Prima forma normale

Tutte le relazioni trovate soddisfano la prima forma normale. Infatti, tutti gli attributi presenti in ciascuna relazione sono atomici: gli attributi composti presenti nello schema EE-R sono stati resi in attributi semplici (Es Indirizzo è stata scomposta in Via, Città e Provincia)

4.3.2 Seconda forma normale

Molte relazioni trovate soddisfano la seconda forma normale. Ciò è evidente per tutte quelle relazioni (in totale 14) la cui chiave è costituita da un solo attributo, mentre tutte le altre (per esempio Consegna) hanno una coppia di chiavi proprie.

4.3.3 Terza forma normale

Molte relazioni trovate soddisfano la terza forma normale. Ciò è evidente per tutte quelle relazioni (in totale 15) dove gli attributi dipendono strettamente dall'attributo chiave, mentre tutte le altre (per esempio Dipendente e Fornitura) alcuni attributi possono anche non dipendere dall'attributo chiave.

4.3.4 Forma normale di Boyce e Codd (BCNF)

Solamente 5 relazioni (Tessera, Conto, Dipendente, Ordinazione e Fornitura) soddisfano la forma normale di Boyce e Codd. Ciò è evidente per quelle relazioni nelle quali un attributo chiave è formato da due parti, come se fosse un'unione di due chiavi primarie.

5 Implementazione SQL

Adesso si passa alla traduzione in codice SQL del modello logico relazionale precedentemente mostrato, nonché di tutte le operazioni definite sulla base di dati

In questa sezione è riportata la parte di codice SQL che serve alla definizione delle tabelle. Creiamo innanzitutto l'utente Amministratore che in automatico faremo connettere al nostro database. Egli avrà tutti i privilegi:

```
CREATE USER AMMINISTRATORE IDENTIFIED BY ADMIN;  
GRANT ALL PRIVILEGES TO AMMINISTRATORE;
```

5.1 Creazione delle Tabelle

Ora vi riporto la creazione di tutte le tabelle presenti nel mio database, con vari vincoli statici

ELIMINAZIONE PRELIMINARE TABELLE

```
DROP TABLE CLIENTE;  
DROP TABLE C_SUL_POSTO;  
DROP TABLE C_D_ASORTO;  
DROP TABLE TESSERA;  
DROP TABLE PUB;  
DROP TABLE DIPENDENTE;  
DROP TABLE CAMERIERE;  
DROP TABLE CUOCO;  
DROP TABLE RIDER;  
DROP TABLE STIPENDIO;  
DROP TABLE TAVOLO;  
DROP TABLE TURNO_DI_LAVORO;  
DROP TABLE PRENOTAZIONE;  
DROP TABLE BLOCCA;  
DROP TABLE ORDINAZIONE;  
DROP TABLE O_AL_TAVOLO;  
DROP TABLE O_A_DOMICILIO;  
DROP TABLE CONTO;  
DROP TABLE CONSEGNA;  
DROP TABLE MENU;  
DROP TABLE COMPRENDE;  
DROP TABLE BIBITA;  
DROP TABLE PANINO;  
DROP TABLE MATERIA_PRIMA;  
DROP TABLE FORNITURA;  
DROP TABLE INGREDIENTE;  
DROP TABLE PARCHEGGIO;
```

CREAZIONE TABELLA CLIENTE

```
CREATE TABLE CLIENTE(  
    RECAPITO_TELEFONICO VARCHAR2(13) PRIMARY KEY,  
    NOME_CLIENTE VARCHAR2(20) NOT NULL,  
    COGNOME_CLIENTE VARCHAR2(20) NOT NULL  
);
```

CREAZIONE TABELLA C_SUL_POSTO E C_D_ASORTO

```
CREATE TABLE C_SUL_POSTO(  
    REC_TELEFONICO_SULPOSTO VARCHAR2(13) PRIMARY KEY,  
    CONSTRAINT FK_MADRE_CLIENTE1 FOREIGN KEY (REC_TELEFONICO_SULPOSTO) REFERENCES  
CLIENTE(RECAPITO_TELEFONICO)  
);
```

```
CREATE TABLE C_D_ASORTO(  
    REC_TELEFONICO_DASPORTO VARCHAR2(13) PRIMARY KEY,  
    CONSTRAINT FK_MADRE_CLIENTE2 FOREIGN KEY (REC_TELEFONICO_DASPORTO) REFERENCES  
CLIENTE(RECAPITO_TELEFONICO)  
);
```

CREAZIONE TABELLA TESSERA

```
CREATE TABLE TESSERA(  
    CODICE_TESSERA VARCHAR2(7) PRIMARY KEY,  
    DATA_COMPLEANNO_CLIENTE DATE NOT NULL,  
    EMAIL VARCHAR2(40) NOT NULL UNIQUE,  
    RECAPITO_CLIENTE VARCHAR2(13),  
    CONSTRAINT FK_USA FOREIGN KEY (RECAPITO_CLIENTE) REFERENCES  
CLIENTE(RECAPITO_TELEFONICO)  
);
```

CREAZIONE TABELLA PUB

```
CREATE TABLE PUB(  
    TELEFONO_PUB VARCHAR2(13) PRIMARY KEY,  
    ORARIO_APERTURA DATE NOT NULL,  
    ORARIO_CHIUSURA DATE NOT NULL,  
    VIA VARCHAR2(255) NOT NULL,  
    CITTA VARCHAR2(100) NOT NULL,  
    PROVINCIA VARCHAR2(2) NOT NULL  
);
```


CREAZIONE TABELLA DIPENDENTE

```
CREATE TABLE DIPENDENTE(  
    CF VARCHAR2(16) PRIMARY KEY,  
    NOME_DIPENDENTE VARCHAR2(20) NOT NULL,  
    COGNOME_DIPENDENTE VARCHAR2(20) NOT NULL,  
    DATA_NASCITA DATE NOT NULL,  
    DATA_ASSUNZIONE DATE NOT NULL,  
    SESSO VARCHAR2(2) NOT NULL,  
    TELEFN_PUB VARCHAR2(13),  
    CONSTRAINT FK_PUB FOREIGN KEY(TELEFN_PUB) REFERENCES PUB(TELEFONO_PUB),  
    CONSTRAINT CHECK_SEX CHECK(SESSO IN('M', 'F', 'NB')),  
    CONSTRAINT CHECK_CF CHECK(LENGTH(CF) =16)  
);
```

CREAZIONE TABELLA CAMERIERE, CUOCO E RIDER

```
CREATE TABLE CAMERIERE(  
    CF_CAMERIERE VARCHAR2(16) PRIMARY KEY,  
    CONSTRAINT FK_MADRE_DIPENDENTE1 FOREIGN KEY (CF_CAMERIERE) REFERENCES  
DIPENDENTE(CF),  
    CONSTRAINT CHECK_CF1 CHECK(LENGTH(CF_CAMERIERE) =16)  
);
```

```
CREATE TABLE CUOCO(  
    CF_CUOCO VARCHAR2(16) PRIMARY KEY,  
    CONSTRAINT FK_MADRE_DIPENDENTE2 FOREIGN KEY (CF_CUOCO) REFERENCES  
DIPENDENTE(CF),  
    CONSTRAINT CHECK_CF2 CHECK(LENGTH(CF_CUOCO) =16)  
);
```

```
CREATE TABLE RIDER(  
    CF_RIDER VARCHAR2(16) PRIMARY KEY,  
    CONSTRAINT FK_MADRE_DIPENDENTE3 FOREIGN KEY (CF_RIDER) REFERENCES  
DIPENDENTE(CF),  
    CONSTRAINT CHECK_CF3 CHECK(LENGTH(CF_RIDER) =16)  
);
```

CREAZIONE TABELLA STIPENDIO

```
CREATE TABLE STIPENDIO(  
    DATA_ACCREDITO DATE,  
    CF_DIP VARCHAR2(16),  
    IMPORTO NUMBER NOT NULL,  
    TRATTENUTE NUMBER NOT NULL,  
    CONSTRAINT PK_STIPENDIO PRIMARY KEY (DATA_ACCREDITO, CF_DIP),  
    CONSTRAINT FK_DIP FOREIGN KEY (CF_DIP) REFERENCES DIPENDENTE(CF),  
    CONSTRAINT CHECK_STP CHECK (IMPORTO >= 800),  
    CONSTRAINT CHECK_CF4 CHECK(LENGTH(CF_DIP) =16)  
);
```

CREAZIONE TABELLA TAVOLO

```
CREATE TABLE TAVOLO(  
    NUMERO_TAVOLO VARCHAR2(3),  
    NUM_SALA VARCHAR2(2) NOT NULL,  
    TE_PUB VARCHAR2(13),  
    NUM_POSTI NUMBER NOT NULL,  
    CONSTRAINT PK_TAVOLO PRIMARY KEY (NUMERO_TAVOLO, TE_PUB),  
    CONSTRAINT FK_PUBS FOREIGN KEY (TE_PUB) REFERENCES PUB(TELEFONO_PUB)  
);
```

CREAZIONE TABELLA TURNO_DI_LAVORO

```
CREATE TABLE TURNO_DI_LAVORO(  
    DATA_ORA_TURNO TIMESTAMP,  
    CF_DIPENDENTE VARCHAR2(16),  
    DURATA NUMBER,  
    T_PUB VARCHAR2(13),  
    CONSTRAINT PK_TURNO_DI_LAVORO PRIMARY KEY (DATA_ORA_TURNO, CF_DIPENDENTE, T_PUB),  
    CONSTRAINT FK_CFDIP FOREIGN KEY (CF_DIPENDENTE) REFERENCES DIPENDENTE(CF),  
    CONSTRAINT CHECK_DURATA CHECK (DURATA <= 8),  
    CONSTRAINT CHECK_CF5 CHECK(LENGTH(CF_DIPENDENTE) =16)  
);
```

CREAZIONE TABELLA PRENOTAZIONE

```
CREATE TABLE PRENOTAZIONE(  
    DATA_ORA_PRENOTAZIONE TIMESTAMP,  
    RECPT_TEL VARCHAR2(13),  
    NUM_TAVOLO VARCHAR2(3),  
    TF_PUB VARCHAR2(13),  
    CONSTRAINT PK_PRENOTAZIONE PRIMARY KEY (DATA_ORA_PRENOTAZIONE, RECPT_TEL,  
    NUM_TAVOLO, TF_PUB),  
    CONSTRAINT FK_EFFETTUATO FOREIGN KEY (RECPT_TEL) REFERENCES  
    C_SUL_POSTO(REC_TELEFONICO_SULPOSTO)  
);
```

CREAZIONE TABELLA BLOCCA

```
CREATE TABLE BLOCCA(  
    ID_TAVOLO VARCHAR2(3),  
    D_O_PRENOTA TIMESTAMP,  
    RECPTO_TEL VARCHAR2(13),  
    TFN_PUB VARCHAR2(13),  
    CONSTRAINT FK_BLOCCA_TAV FOREIGN KEY (ID_TAVOLO, TFN_PUB) REFERENCES  
TAVOLO(NUMERO_TAVOLO, TE_PUB),  
    CONSTRAINT FK_BLOCCA_PRE1 FOREIGN KEY  
(D_O_PRENOTA, RECPTO_TEL, ID_TAVOLO, TFN_PUB) REFERENCES  
PRENOTAZIONE(DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)  
);
```

CREAZIONE TABELLA ORDINAZIONE

```
CREATE TABLE ORDINAZIONE(  
    COD_ORDINAZIONE VARCHAR2(255) PRIMARY KEY  
);
```

CREAZIONE TABELLA O_AL_TAVOLO E O_A_DOMICILIO

```
CREATE TABLE O_AL_TAVOLO(  
    COD_ORDINAZIONE_TAVOLO VARCHAR2(255) PRIMARY KEY,  
    DA_O_PRENOT_TAV TIMESTAMP,  
    REC_TEL VARCHAR2(13),  
    NM_TAVOLO VARCHAR2(3),  
    PHONE_PUB VARCHAR2(13),  
    CONSTRAINT FK_MADRE_ORDINAZIONE1 FOREIGN KEY (COD_ORDINAZIONE_TAVOLO)  
REFERENCES ORDINAZIONE(COD_ORDINAZIONE),  
    CONSTRAINT FK_PRENOTAZ FOREIGN KEY  
(DA_O_PRENOT_TAV, REC_TEL, NM_TAVOLO, PHONE_PUB) REFERENCES  
PRENOTAZIONE(DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)  
);
```

```
CREATE TABLE O_A_DOMICILIO(  
    COD_ORDINAZIONE_DOMICILIO VARCHAR2(255) PRIMARY KEY,  
    RECET_TEL VARCHAR2(13),  
    CONSTRAINT FK_MADRE_ORDINAZIONE2 FOREIGN KEY (COD_ORDINAZIONE_DOMICILIO)  
REFERENCES ORDINAZIONE(COD_ORDINAZIONE),  
    CONSTRAINT FK_COLLEGA FOREIGN KEY (RECET_TEL) REFERENCES  
C_D_ASORTO(REC_TELEFONICO_DASPORTO)  
);
```

CREAZIONE TABELLA CONTO

```
CREATE TABLE CONTO(  
    ID_SCONTRINO VARCHAR2(15) PRIMARY KEY,  
    ORA_DATA_SCONTRINO DATE NOT NULL,  
    METODO_PAGAMENTO VARCHAR2(20) NOT NULL,  
    ID_ORDINE VARCHAR2(255),  
    RECAPITO_PUB VARCHAR2(13),  
    SCONTO VARCHAR2(2) NOT NULL,  
    CONSTRAINT FK_PAGATO FOREIGN KEY (ID_ORDINE) REFERENCES  
ORDINAZIONE(COD_ORDINAZIONE),  
    CONSTRAINT FK_INCASSATO FOREIGN KEY (RECAPITO_PUB) REFERENCES  
PUB(TELEFONO_PUB),  
    CONSTRAINT CHECK_MP CHECK(METODO_PAGAMENTO IN('CARTA', 'CONTANTI'))  
);
```

CREAZIONE TABELLA CONSEGNA

```
CREATE TABLE CONSEGNA(  
    DATA_ORA_CONSEGNA TIMESTAMP,  
    VIA VARCHAR2(40),  
    CITTA VARCHAR2(30),  
    CF_RID VARCHAR2(16),  
    COD_ORD VARCHAR2(255),  
    CONSTRAINT PK_CONSEGNA PRIMARY KEY (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID),  
    CONSTRAINT FK_RIDER FOREIGN KEY (CF_RID) REFERENCES RIDER(CF_RIDER),  
    CONSTRAINT FK_COMPOSTO_DA FOREIGN KEY (COD_ORD) REFERENCES  
O_A_DOMICILIO(COD_ORDINAZIONE_DOMICILIO),  
    CONSTRAINT CHECK_CF6 CHECK(LENGTH(CF_RID) =16)  
);
```

CREAZIONE TABELLA MENU

```
CREATE TABLE MENU(  
    NOME_MENU VARCHAR2(20) PRIMARY KEY,  
    PREZZO NUMBER NOT NULL  
);
```

CREAZIONE TABELLA COMPRENDE

```
CREATE TABLE COMPRENDE(  
    NUM_ORD VARCHAR2(255),  
    NOM_MENU VARCHAR2(20),  
    QUANTITA NUMBER(5) NOT NULL,  
    CONSTRAINT FK_COMPRENDE_ORD FOREIGN KEY (NUM_ORD) REFERENCES  
ORDINAZIONE(COD_ORDINAZIONE),  
    CONSTRAINT FK_COMPRENDE_MENU FOREIGN KEY (NOM_MENU) REFERENCES MENU(NOME_MENU)  
);
```

CREAZIONE TABELLA PANINO E BIBITA

```
CREATE TABLE BIBITA(  
    N_MENU_BIBITA VARCHAR2(20) PRIMARY KEY,  
    ALCOLICO VARCHAR2(2) NOT NULL,  
    CONSTRAINT FK_MADRE_MENU1 FOREIGN KEY (N_MENU_BIBITA) REFERENCES  
MENU(NOME_MENU),  
    CHECK(ALCOLICO IN('SI', 'NO'))  
);
```

```
CREATE TABLE PANINO(  
    N_MENU_PANINO VARCHAR2(20) PRIMARY KEY,  
    CONSTRAINT FK_MADRE_MENU2 FOREIGN KEY (N_MENU_PANINO) REFERENCES  
MENU(NOME_MENU)  
);
```

CREAZIONE TABELLA MATERIA PRIMA

```
CREATE TABLE MATERIA_PRIMA(  
    NOME_MATERIA_PRIMA VARCHAR2(255) PRIMARY KEY  
);
```

CREAZIONE TABELLA FORNITURA

```
CREATE TABLE FORNITURA(  
    COD_FORNITURA VARCHAR2(255),  
    TELEF_PUB VARCHAR2(13),  
    COD_MAT_PR VARCHAR2(255),  
    N_BIBITA VARCHAR2(20),  
    NOME_FORNITORE VARCHAR2(255) NOT NULL,  
    PREZZO NUMBER NOT NULL,  
    QUANTITA_ACQUISTO NUMBER NOT NULL,  
    DATA_RICHIESTA DATE NOT NULL,  
    DATA_CONSEGNA DATE NOT NULL,  
    DATA_SCADENZA DATE NOT NULL,  
    CONSTRAINT PK_FORNITURA PRIMARY KEY (COD_FORNITURA, TELEF_PUB),  
    CONSTRAINT FK_RICHIESTO FOREIGN KEY (TELEF_PUB) REFERENCES PUB(TELEFONO_PUB),  
    CONSTRAINT FK_CONTIENE FOREIGN KEY (COD_MAT_PR) REFERENCES  
MATERIA_PRIMA(NOME_MATERIA_PRIMA),  
    CONSTRAINT FK_RIGUARDA FOREIGN KEY (N_BIBITA) REFERENCES BIBITA(N_MENU_BIBITA)  
);
```

CREAZIONE TABELLA INGREDIENTE

```
CREATE TABLE INGREDIENTE(  
  N_MENU VARCHAR2(20),  
  COD_MP VARCHAR2(255),  
  QUANTITA_INGREDIENTE NUMBER NOT NULL,  
  CONSTRAINT FK_COMPOSTO_DA_MENU FOREIGN KEY (N_MENU) REFERENCES  
  PANINO(N_MENU_PANINO),  
  CONSTRAINT FK_COMPOSTO_DA INGR FOREIGN KEY (COD_MP) REFERENCES  
  MATERIA_PRIMA(NOME_MATERIA_PRIMA)  
);
```

CREAZIONE TABELLA PARCHEGGIO

```
CREATE TABLE PARCHEGGIO(  
  VIA VARCHAR2(255),  
  CITTA VARCHAR2(255),  
  COSTO_POSTO_ORA NUMBER NOT NULL,  
  ORA_APERTURA DATE NOT NULL,  
  ORA_CHIUSURA DATE NOT NULL,  
  NOME_PARCHEGGIO VARCHAR2(255) NOT NULL,  
  TOTALE_POSTI_RISERVATI NUMBER NOT NULL,  
  DISTANZA_PUB NUMBER NOT NULL,  
  N_PUB_ASSOCIATO VARCHAR2(13),  
  CONSTRAINT PK_PARCHEGGIO PRIMARY KEY(VIA,CITTA),  
  CONSTRAINT FK_PARCHEGGIO FOREIGN KEY (N_PUB_ASSOCIATO) REFERENCES  
  PUB(TELEFONO_PUB)  
);
```

5.2 Creazione delle Viste

CREAZIONE VISTA PER VISUALIZZARE TUTTI GLI ORDINI EFFETTUATI NEL GIORNO CORRENTE

```
CREATE OR REPLACE VIEW CONTI_ORD AS
SELECT CONTO.ID_SCONTRINO, CONTO.ORA_DATA_SCONTRINO, CONTO.RECAPITO_PUB,
COMPRENDE.NUM_ORD,NOM_MENU ,SUM(MENU.PREZZO) AS PREZZO_TOTALE,
CONTO.METODO_PAGAMENTO
FROM ((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=NUM_ORD) JOIN MENU ON NOM_MENU=NOME_MENU
WHERE TO_CHAR(ORA_DATA_SCONTRINO,'DD/MM/YYYY' )=TO_CHAR(SYSDATE,'DD/MM/YYYY' )
GROUP BY ID_SCONTRINO, ORA_DATA_SCONTRINO, RECAPITO_PUB, NUM_ORD, PREZZO,
METODO_PAGAMENTO,NOM_MENU
ORDER BY ID_SCONTRINO,RECAPITO_PUB;
```

CREAZIONE VISTA PER VEDERE IL MENÙ DEI PANINI

```
CREATE OR REPLACE VIEW MENU_PANINI AS
SELECT NOME_MENU as NOME_PANINO, PREZZO
FROM MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO
ORDER BY NOME_MENU;
```

CREAZIONE VISTA PER VEDERE IL MENÙ DELLE BIBITE

```
CREATE OR REPLACE VIEW MENU_BIBITE AS
SELECT NOME_MENU AS BIBITE, PREZZO,ALCOLICO
FROM MENU JOIN BIBITA ON NOME_MENU=N_MENU_BIBITA
ORDER BY NOME_MENU;
```

CREAZIONE VISTA PER VEDERE GLI INGREDIENTI DEI PANINI, VERSIONE 1

```
CREATE OR REPLACE VIEW INGRE_PANINI AS
SELECT COD_MP AS IRON_MAN, NULL AS HULK,NULL AS CAPITAN_AMERICA, NULL AS
SCARLET_WITCH, NULL AS SPIDER_MAN, NULL AS DOCTOR_STRANGE, NULL AS THOR
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON
N_MENU_PANINO=N_MENU
WHERE NOME_MENU ='IRON MAN'
```

UNION

```
SELECT NULL AS IRON_MAN, COD_MP AS HULK,NULL AS CAPITAN_AMERICA, NULL AS
SCARLET_WITCH, NULL AS SPIDER_MAN, NULL AS DOCTOR_STRANGE, NULL AS THOR
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON
N_MENU_PANINO=N_MENU
WHERE NOME_MENU ='HULK'
```

UNION

```
SELECT NULL AS IRON_MAN, NULL AS HULK,COD_MP AS CAPITAN_AMERICA, NULL AS
SCARLET_WITCH, NULL AS SPIDER_MAN, NULL AS DOCTOR_STRANGE, NULL AS THOR
```

```
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON  
N_MENU_PANINO=N_MENU  
WHERE NOME_MENU ='CAPITAN AMERICA'
```

```
UNION  
SELECT NULL AS IRON_MAN, NULL AS HULK, NULL AS CAPITAN_AMERICA, COD_MP AS  
SCARLET_WITCH, NULL AS SPIDER_MAN, NULL AS DOCTOR_STRANGE, NULL AS THOR  
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON  
N_MENU_PANINO=N_MENU  
WHERE NOME_MENU ='SCARLET WITCH'  
UNION
```

```
SELECT NULL AS IRON_MAN, NULL AS HULK, NULL AS CAPITAN_AMERICA, NULL AS  
SCARLET_WITCH, COD_MP AS SPIDER_MAN, NULL AS DOCTOR_STRANGE, NULL AS THOR  
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON  
N_MENU_PANINO=N_MENU  
WHERE NOME_MENU ='SPIDER MAN'
```

```
UNION  
SELECT NULL AS IRON_MAN , NULL AS HULK, NULL AS CAPITAN_AMERICA, NULL AS  
SCARLET_WITCH, NULL AS SPIDER_MAN, COD_MP AS DOCTOR_STRANGE, NULL AS THOR  
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON  
N_MENU_PANINO=N_MENU  
WHERE NOME_MENU ='DOCTOR STRANGE'
```

```
UNION  
SELECT NULL AS IRON_MAN, NULL AS HULK, NULL AS CAPITAN_AMERICA, NULL AS  
SCARLET_WITCH, NULL AS SPIDER_MAN, NULL AS DOCTOR_STRANGE, COD_MP AS THOR  
FROM (MENU JOIN PANINO ON NOME_MENU=N_MENU_PANINO) JOIN INGREDIENTE ON  
N_MENU_PANINO=N_MENU  
WHERE NOME_MENU ='THOR';
```

CREAZIONE VISTA PER VEDERE GLI INGREDIENTI DEI PANINI, VERSIONE 2

```
CREATE OR REPLACE VIEW INGREDIENTI2 AS  
SELECT N_MENU_PANINO AS NOME_PANINO, COD_MP AS INGREDIENTE  
FROM INGREDIENTE JOIN PANINO ON N_MENU=N_MENU_PANINO;
```


CREAZIONE VISTA PER IL RIEPILOGO DELLE VENDITE DI MATERIE PRIME DEL PUB

```
CREATE OR REPLACE VIEW RIEPILOGO_VENDITE_MP AS
select
NOME_MATERIA_PRIMA, RECAPITO_PUB, COUNT (NOME_MATERIA_PRIMA) as PORZIONI_VENDUTE ,
rtrim(to_char(QUANTITA_INGREDIENTE, 'FM90.999'), '.') AS KG_PER_PORZIONE,
rtrim(to_char((COUNT (NOME_MATERIA_PRIMA)*QUANTITA_INGREDIENTE), 'FM90.999'), '.')
AS SCORTE_CONSUMATE_KG
from(
SELECT *
FROM (((((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=
NUM_ORD) JOIN MENU ON NOM_MENU=NOME_MENU ) JOIN PANINO ON N_MENU_PANINO=NOME_MENU)
JOIN INGREDIENTE
ON N_MENU=N_MENU_PANINO) JOIN MATERIA_PRIMA ON NOME_MATERIA_PRIMA=COD_MP)
GROUP BY NOME_MATERIA_PRIMA, QUANTITA_INGREDIENTE, RECAPITO_PUB
ORDER BY NOME_MATERIA_PRIMA;
```

CREAZIONE VISTA PER IL RIEPILOGO DELLE FORNITURE DI MATERIE PRIME DEL PUB

```
CREATE OR REPLACE VIEW RIEPILOGO_FORNITURE_MP AS
SELECT NOME_MATERIA_PRIMA, TELEF_PUB,
rtrim(to_char(QUANTITA_ACQUISTO, 'FM900.99'), '.') AS QUANTITA_ACQUISTO_KG,
rtrim(to_char(PREZZO, 'FM90.99'), '.') AS
PREZZO, NOME_FORNITORE, DATA_RICHIESTA, DATA_CONSEGNA, DATA_SCADENZA
FROM(
SELECT *
FROM (PUB JOIN FORNITURA ON TELEF_PUB=TELEFONO_PUB) JOIN MATERIA_PRIMA ON
COD_MAT_PR=NOME_MATERIA_PRIMA
)
ORDER BY NOME_MATERIA_PRIMA, TELEF_PUB;
```

CREAZIONE VISTA PER IL RIEPILOGO DELLE VENDITE DI BIBITE DEL PUB

```
CREATE OR REPLACE VIEW RIEPILOGO_VENDITE_BIBITE AS
select
N_MENU_BIBITA, QUANTITA, RECAPITO_PUB, COD_ORDINAZIONE
FROM (((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=
NUM_ORD) JOIN MENU ON NOM_MENU=NOME_MENU ) JOIN BIBITA ON N_MENU_BIBITA=NOME_MENU
GROUP BY N_MENU_BIBITA, RECAPITO_PUB, QUANTITA, COD_ORDINAZIONE;
```

CREAZIONE VISTA PER IL RIEPILOGO DELLE FORNITURE DI BIBITE DEL PUB

```
CREATE OR REPLACE VIEW RIEPILOGO_FORNITURE_BIBITE AS
SELECT N_BIBITA,TELEF_PUB,
rtrim(to_char(QUANTITA_ACQUISTO, 'FM900.99'), '.') AS QUANTITA_ACQUISTO,
rtrim(to_char(PREZZO, 'FM90.99'), '.') AS
PREZZO,NOME_FORNITORE,DATA_RICHIESTA,DATA_CONSEGNA,DATA_SCADENZA
FROM(
SELECT *
FROM (PUB JOIN FORNITURA ON TELEF_PUB=TELEFONO_PUB)JOIN BIBITA ON
N_MENU_BIBITA=N_BIBITA
)
ORDER BY N_BIBITA,TELEF_PUB;
```

5.3 Creazione degli Utenti

Di seguito sono riportati gli utenti che accedono alla base di dati con i rispettivi privilegi.

UTENTE CLIENTE

```
CREATE USER CLIENTE IDENTIFIED BY ACQUIRENTE;
GRANT CONNECT, CREATE SESSION TO CLIENTE;

GRANT SELECT ON MENU_PANINI TO CLIENTE;
GRANT SELECT ON MENU_BIBITE TO CLIENTE;
GRANT SELECT ON INGRE_PANINI TO CLIENTE;
GRANT SELECT ON INGREDIENTI2 TO CLIENTE;
GRANT EXECUTE ON PARK TO CLIENTE;
```

UTENTE DIPENDENTE

```
CREATE USER DIPENDENTE IDENTIFIED BY PERSONALE;
GRANT CONNECT, CREATE SESSION TO DIPENDENTE;

GRANT SELECT ON CONTI_ORD TO DIPENDENTE;
GRANT EXECUTE ON DELETE_INFO_CLIENTE TO DIPENDENTE;
GRANT EXECUTE ON SCONTO_TES TO DIPENDENTE;
```

UTENTE RESPONSABILE_PUB

```
CREATE USER RESPONSABILE_PUB IDENTIFIED BY RESPONSABILE;
GRANT CONNECT, CREATE SESSION TO RESPONSABILE_PUB;

GRANT EXECUTE ON AUM_STP TO RESPONSABILE_PUB;
GRANT EXECUTE ON BONUS_MESE TO RESPONSABILE_PUB;
```

UTENTE GESTORE_FORNITURE

```
CREATE USER GESTORE_FORNITURE IDENTIFIED BY FORNITORE_PUB;
GRANT CONNECT, CREATE SESSION TO GESTORE_FORNITURE;

GRANT SELECT ON RIEPILOGO_VENDITE_MP TO GESTORE_FORNITURE;
GRANT SELECT ON RIEPILOGO_FORNITURE_MP TO GESTORE_FORNITURE;
GRANT SELECT ON RIEPILOGO_VENDITE_BIBITE TO GESTORE_FORNITURE;
GRANT SELECT ON RIEPILOGO_FORNITURE_BIBITE TO GESTORE_FORNITURE;
GRANT EXECUTE ON SCORTA TO GESTORE_FORNITURE;
```

5.4 Creazione dei Trigger

I seguenti Trigger che sono la traduzione dei vincoli di integrità dinamici. Essi effettuano controlli prima di eseguire operazioni di Inserimento (BEFORE INSERT):

TRIGGER PER IL CONTROLLO DEL LIMITE MASSIMO DI DIPENDENTI

Questo trigger si attiva quando all'inserimento di un dipendente del pub si supera il limite massimo di dipendenti per pub (12). Quindi la sua funzione è quella di controllare se quell'entità rispetta il limite di 12 tuple per pub. Se non lo rispetta viene lanciato un avvertimento e non si concretizzerà l'insert:

```
CREATE OR REPLACE TRIGGER MAX_DEP
BEFORE INSERT ON DIPENDENTE
FOR EACH ROW
DECLARE
CONT_DIP NUMBER(3);
MAX_DIP EXCEPTION;
BEGIN
SELECT COUNT(CF) INTO CONT_DIP
FROM DIPENDENTE
WHERE TELEFN_PUB=:NEW.TELEFN_PUB;
IF CONT_DIP=12 THEN
RAISE MAX_DIP;
END IF;
EXCEPTION
WHEN MAX_DIP THEN
RAISE_APPLICATION_ERROR(-20001, 'Limite di 12 dipendenti');
END MAX_DEP;
```

TRIGGER PER IL CONTROLLO DELLA DATA DI SCADENZA DELLA FORNITURA

Questo trigger si attiva quando si inserisce una fornitura. Si controlla la sua data di scadenza e se questa è inferiore a 1 settimana dalla data di richiesta, viene lanciato l'errore che blocca l'inserimento:

```
CREATE OR REPLACE TRIGGER NO_SCADUTO
BEFORE INSERT ON FORNITURA
FOR EACH ROW
DECLARE
SCADUTO EXCEPTION;
BEGIN
IF (:NEW.DATA_SCADENZA<:NEW.DATA_RICHIESTA+7) THEN
RAISE SCADUTO;
END IF;
EXCEPTION
WHEN SCADUTO THEN
RAISE_APPLICATION_ERROR(-20002, 'Fornitura prossima alla scadenza');
END NO_SCADUTO;
```

TRIGGER PER IL CONTROLLO DEL BLOCCO DEI TAVOLI DELLA PRENOTAZIONE

Questo trigger si attiva quando durante la fase di prenotazione dei tavoli. Se il numero dei tavoli prenotati è maggiore di 5 il trigger segnala con un messaggio di errore tale condizione:

```
CREATE OR REPLACE TRIGGER BLOCK_TAV
BEFORE INSERT ON PRENOTAZIONE
FOR EACH ROW
DECLARE
COUNT_TAV NUMBER(2);
MAX_TAV EXCEPTION;
BEGIN
SELECT COUNT(*) INTO COUNT_TAV
FROM PRENOTAZIONE
WHERE RECPT_TEL=:NEW.RECPT_TEL AND DATA_ORA_PRENOTAZIONE=:NEW.DATA_ORA_PRENOTAZIONE
AND TF_PUB=:NEW.TF_PUB;
IF COUNT_TAV>=5 THEN
RAISE MAX_TAV;
END IF;
EXCEPTION
WHEN MAX_TAV THEN
RAISE_APPLICATION_ERROR(-20003, 'Non puoi bloccare più di 5 tavoli in questo
giorno');
END BLOCK_TAV;
```

TRIGGER PER IL CONTROLLO DELLO STIPENDIO DEL NEO-DIPENDENTE

Questo trigger non dà accesso ai neo-dipendenti ad uno stipendio che viene dato ai dipendenti esperti perché i neo-dipendenti a differenza degli esperti sono privi di esperienza lavorativa, quindi non possono guadagnare più di 1600 euro:

```
CREATE OR REPLACE TRIGGER DIP_ESPERTO
BEFORE INSERT ON STIPENDIO
FOR EACH ROW
DECLARE
DATA_AST DATE;
STP_HIGH EXCEPTION;
BEGIN
SELECT DATA_ASSUNZIONE INTO DATA_AST
FROM DIPENDENTE
WHERE :NEW.CF_DIP=CF;
IF DATA_AST>SYSDATE - 365 AND :NEW.IMPORTO>1600 THEN
RAISE STP_HIGH;
END IF;
EXCEPTION
WHEN STP_HIGH THEN
RAISE_APPLICATION_ERROR(-20004, 'Il dipendente ha lavorato per un periodo troppo
breve per questo stipendio');
END DIP_ESPERTO;
```

TRIGGER PER IL CONTROLLO DEI GIORNI LAVORATIVI DEL DIPENDENTE

Questo trigger tutela i dipendenti controllando se il loro turno sono meno di 7 per ogni settimana, garantendo così almeno un giorno di riposo al dipendente:

```
CREATE OR REPLACE TRIGGER GIORNO_FERIALE
BEFORE INSERT OR UPDATE ON TURNO_DI_LAVORO
FOR EACH ROW
DECLARE
TURNI_SETTIMANALI NUMBER(1);
FERIALE EXCEPTION;
BEGIN
SELECT COUNT (*) INTO TURNI_SETTIMANALI
FROM TURNO_DI_LAVORO
WHERE :NEW.CF_DIPENDENTE=CF_DIPENDENTE AND
TO_CHAR(:NEW.DATA_ORA_TURNO, 'IYYYIW')=TO_CHAR(DATA_ORA_TURNO, 'IYYYIW');
IF TURNI_SETTIMANALI>=6 THEN
RAISE FERIALE;
END IF;
EXCEPTION
WHEN FERIALE THEN RAISE_APPLICATION_ERROR(-20005, 'Il dipendente deve riposare
almeno un giorno a settimana!');
END GIORNO_FERIALE;
```

TRIGGER PER IL CONTROLLO DELLE ORE LAVORATIVE DEL DIPENDENTE

Questo trigger si attiva quando, durante il calcolo delle ore complessive mensili, queste risultano essere superiori a 190. Quindi questo trigger tutela il dipendente dallo sfruttamento lavorativo, allarmando che tale dipendente abbia lavorato troppo in quel mese:

```
CREATE OR REPLACE TRIGGER MASSIMO_ORE
BEFORE INSERT OR UPDATE ON TURNO_DI_LAVORO
FOR EACH ROW
DECLARE
ORE_LAVORO NUMBER(8);
MAX_ORE EXCEPTION;
BEGIN
SELECT SUM(DURATA) INTO ORE_LAVORO
FROM TURNO_DI_LAVORO
WHERE :NEW.CF_DIPENDENTE=CF_DIPENDENTE AND TO_CHAR(:NEW.DATA_ORA_TURNO, 'MM-
YYYY')=TO_CHAR(DATA_ORA_TURNO, 'MM-YYYY');
ORE_LAVORO:=ORE_LAVORO+:NEW.DURATA;
IF ORE_LAVORO>=190 THEN
RAISE MAX_ORE;
END IF;
EXCEPTION
WHEN MAX_ORE THEN RAISE_APPLICATION_ERROR(-20006, 'Il dipendente ha lavorato
troppo per questo mese');
END MASSIMO_ORE;
```

TRIGGER PER IL CONTROLLO DEI TURNI LAVORATIVI

Questo trigger si attiva se nello stesso giorno vengono identificati più turni di lavoro da parte dello stesso dipendente. Quindi la funzione di questo trigger è assicurarsi che i dipendenti che hanno già prestato servizio una volta non possano prestarlo altre volte in quel giorno:

```
CREATE OR REPLACE TRIGGER TURNO_DOPPIO
BEFORE INSERT OR UPDATE ON TURNO_DI_LAVORO
FOR EACH ROW
DECLARE
NUM_TURNI NUMBER(1);
MAX_TURNI EXCEPTION;
BEGIN
SELECT COUNT (*) INTO NUM_TURNI
FROM TURNO_DI_LAVORO
WHERE :NEW.CF_DIPENDENTE=CF_DIPENDENTE AND TO_CHAR(:NEW.DATA_ORA_TURNO, 'DD-MM-
YYYY')=TO_CHAR(DATA_ORA_TURNO, 'DD-MM-YYYY');
IF NUM_TURNI>=1 THEN
RAISE MAX_TURNI;
END IF;
EXCEPTION
WHEN MAX_TURNI THEN RAISE_APPLICATION_ERROR(-20007, 'Il dipendente ha già lavorato
oggi');
END TURNO_DOPPIO;
```

TRIGGER PER IL CONTROLLO DELLA PRENOTAZIONE DEL CLIENTE

Questo trigger si attiva quando durante la prenotazione da parte di un cliente, quest'ultimo si prenota solo per il giorno corrente o al più per i 2 giorni successivi:

```
CREATE OR REPLACE TRIGGER LIMIT_DAY
BEFORE INSERT OR UPDATE ON PRENOTAZIONE
FOR EACH ROW
DECLARE
LIMDAY EXCEPTION;
BEGIN
IF (:NEW.DATA_ORA_PRENOTAZIONE>SYSDATE+2) THEN
RAISE LIMDAY;
END IF;
EXCEPTION
WHEN LIMDAY THEN
RAISE_APPLICATION_ERROR(-20008, 'Limite giorno di prenotazione superato');
END LIMIT_DAY;
```

TRIGGER PER IL CONTROLLO DEL LAVORO DEL DIPENDENTE OLTRE L'ORARIO DI CHIUSURA

Questo trigger si attiva quando il dipendente continua a lavorare fino a notte inoltrata. Quindi se il lavoratore starebbe ancora lavorando oltre le 03:00 di notte, emana un messaggio di avviso:

```
CREATE OR REPLACE TRIGGER ORA_CHIUSURA
BEFORE INSERT OR UPDATE ON TURNO_DI_LAVORO
FOR EACH ROW
DECLARE
LIMITE TIMESTAMP;
LIMITEMAX TIMESTAMP;
LIMITEMAX1 TIMESTAMP;

LIM_ORE EXCEPTION;
OK EXCEPTION;
BEGIN
LIMITE:=:NEW.DATA_ORA_TURNO + (:NEW.DURATA/24) ;
LIMITEMAX:= TO_TIMESTAMP('23:59', 'HH24:MI');
LIMITEMAX1:= TO_TIMESTAMP('10:00', 'HH24:MI');
IF TO_CHAR(LIMITE, 'HH24:MI')<TO_CHAR(LIMITEMAX, 'HH24:MI') AND
TO_CHAR(LIMITE, 'HH24:MI')>TO_CHAR(LIMITEMAX1, 'HH24:MI')
THEN
RAISE OK;
ELSE
LIMITEMAX:= TO_TIMESTAMP('03:00', 'HH24:MI');
END IF;
IF TO_CHAR(LIMITE, 'HH24:MI')>TO_CHAR(LIMITEMAX, 'HH24:MI') THEN
RAISE LIM_ORE;
END IF;
```

```

EXCEPTION
WHEN OK THEN NULL;
WHEN LIM_ORE THEN RAISE_APPLICATION_ERROR(-20009, 'Il dipendente lavorerebbe oltre
l orario di chiusura');
END ORA_CHIUSURA;

```

5.5 Procedure PL/SQL

PROCEDURA CHE AUMENTA LO STIPENDIO DEL DIPENDENTE SE QUESTO è MINORE DI 1500 EURO E IN UN MESE HA TOTALIZZATO Più DI 160 ORE LAVORATIVE

Tale procedura, ha come parametro di input il codice fiscale del dipendente. La procedura fa un controllo per sapere se lo stipendio è maggiore di 1500 euro, se sì allora scatterà il primo messaggio d'eccezione per dire che lo stipendio è già alto, altrimenti fa il secondo controllo dove scopre se le ore lavorative mensili sono minori delle ore stabilite (in questo caso 160), se sì allora scatterà il secondo messaggio d'eccezione dove segnala appunto che le ore minime non sono state raggiunte. Se si superano i due controlli allora farà un aggiornamento nella tabella Stipendio aumentando (+100) l'importo:

```

CREATE OR REPLACE PROCEDURE AUM_STP(CF_DIPE IN VARCHAR2) IS
STIPENDIO_ATT NUMBER;
ORE_DUR NUMBER;
STIPENDIO_ALTO EXCEPTION;
ORE_MINIME EXCEPTION;
BEGIN
SELECT MAX(IMPORTO), SUM(DURATA) INTO STIPENDIO_ATT, ORE_DUR
FROM (STIPENDIO JOIN DIPENDENTE ON CF_DIP=CF) JOIN TURNO_DI_LAVORO ON
CF=CF_DIPENDENTE
WHERE CF_DIP=CF_DIPE
GROUP BY CF_DIP;
IF STIPENDIO_ATT>1500 THEN
RAISE STIPENDIO_ALTO;
RETURN;
ELSIF ORE_DUR<160 THEN
RAISE ORE_MINIME;
RETURN;
ELSE
UPDATE STIPENDIO SET IMPORTO = IMPORTO +100 WHERE CF_DIP=CF_DIPE;
UPDATE TURNO_DI_LAVORO SET DURATA = NULL WHERE CF_DIPENDENTE=CF_DIPE;
END IF;
EXCEPTION
WHEN STIPENDIO_ALTO THEN
RAISE_APPLICATION_ERROR(-20011, 'STIPENDIO TROPPO ALTO');
WHEN ORE_MINIME THEN
RAISE_APPLICATION_ERROR(-20012, 'ORE MINIME NON RAGGIUNTE');
END;
/
--EXECUTE AUM_STP('CF DEL DIPENDENTE');

```


*PROCEDURA CHE APPLICA LO SCONTO (SCRIVENDO SI) AL CLIENTE UNA VOLTA
PAGATO IL CONTO SOLO SE IL CLIENTE È TESSERATO*

Tale procedura, ha come parametri di input il codice d'ordinazione del cliente e il suo numero di telefono. La procedura controlla che il cliente sia tesserato; se lo è allora passa al calcolo del conto applicando uno sconto del 10%.

```
CREATE OR REPLACE PROCEDURE SCONTO_TES(ORDINATION IN VARCHAR2, TELEFONO IN
VARCHAR2) IS
TESSERA_PRESENTE VARCHAR2(20);
TOTALE_CONTO NUMBER;
BEGIN
SELECT CODICE_TESSERA INTO TESSERA_PRESENTE
FROM (((((TESSERA JOIN CLIENTE ON RECAPITO_CLIENTE=RECAPITO_TELEFONICO) JOIN
C_SUL_POSTO ON RECAPITO_TELEFONICO=REC_TELEFONICO_SULPOSTO) JOIN PRENOTAZIONE ON
REC_TELEFONICO_SULPOSTO=RECPT_TEL) JOIN O_AL_TAVOLO ON
DATA_ORA_PRENOTAZIONE=DA_O_PRENOT_TAV) JOIN ORDINAZIONE ON
COD_ORDINAZIONE_TAVOLO=COD_ORDINAZIONE) JOIN CONTO ON COD_ORDINAZIONE=ID_ORDINE
WHERE ID_ORDINE=ORDINATION AND RECAPITO_TELEFONICO=TELEFONO;
IF TESSERA_PRESENTE IS NOT NULL THEN
UPDATE CONTO SET SCONTO = 'SI' WHERE ID_ORDINE=ORDINATION;
SELECT SUM(PREZZO*QUANTITA) INTO TOTALE_CONTO
FROM ((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=NUM_ORD) JOIN MENU ON NOM_MENU=NOME_MENU
WHERE ID_ORDINE=ORDINATION
GROUP BY ID_ORDINE;
TOTALE_CONTO:=TOTALE_CONTO-(TOTALE_CONTO*10/100);
DBMS_OUTPUT.put_line('Viene applicato lo sconto e il prezzo totale è: '
||TOTALE_CONTO);
ELSE
UPDATE CONTO SET SCONTO = 'NO' WHERE ID_ORDINE=ORDINATION;
DBMS_OUTPUT.put_line('Non viene applicato lo sconto');
END IF;
END;
/
--EXECUTE SCONTO_TES('CODICE ORDINAZIONE DEL CLIENTE', 'NUM DI TELEFONO CLIENTE');
```

PROCEDURA CHE ELIMINA LE INFORMAZIONI DEL CLIENTE(A DOMICILIO) SOLO SE QUEST'ULTIMO NON HA EFFETTUATO ORDINAZIONI DA PIÙ DI UN ANNO

Tale procedura, ha come parametri il codice d'ordinazione del cliente e il suo numero di telefono. La procedura, selezionando la massima ora e data dello scontrino proveniente da Conto, controlla se essa è inferiore a un anno. Se sì allora farà un'eliminazione nella tabella Consegna, nella tabella Ordinazione (A Domicilio), nella tabella Cliente (D'Asporto) e nella tabella Conto, con un messaggio che segnala l'eliminazione del cliente:

```
CREATE OR REPLACE PROCEDURE DELETE_INFO_CLIENTE (ORDINATION IN VARCHAR2, PHONE IN
VARCHAR2)
IS
DATA_SCONTRINO TIMESTAMP;
BEGIN
SELECT MAX(ORA_DATA_SCONTRINO) INTO DATA_SCONTRINO
FROM ((CONTO join ordinazione on id_ordine=COD_ORDINAZIONE) join O_A_DOMICILIO on
COD_ORDINAZIONE=COD_ORDINAZIONE_DOMICILIO) JOIN CONSEGNA ON
COD_ORDINAZIONE_DOMICILIO=COD_ORD
where COD_ORD=ORDINATION AND RECET_TEL=PHONE;
IF DATA_SCONTRINO<SYSDATE-365 THEN
DELETE FROM CONSEGNA WHERE COD_ORD=ORDINATION;
DELETE FROM O_A_DOMICILIO WHERE RECET_TEL=PHONE;
DELETE FROM C_D_ASporto WHERE REC_TELEFONICO_DASPORTO=PHONE;
DELETE FROM CONTO WHERE ID_ORDINE=ORDINATION;
DBMS_OUTPUT.PUT_LINE('Eliminazione informazioni cliente confermata');
ELSE
DBMS_OUTPUT.PUT_LINE('Cliente ha diritto di rimanere nel sistema');
END IF;
END;
/
--EXECUTE DELETE_INFO_CLIENTE('CODICE ORDINAZIONE A DOMICILIO DEL CLIENTE
D'ASPORTO', 'TELEFONO CLIENTE D'ASPORTO');
```

*PROCEDURA CHE PERMETTE AL CLIENTE DI SAPERE SE IL PUB È DOTATO DI UN
PARCHEGGIO AFFILIATO*

Tale procedura, ha come parametro il numero telefonico del pub. Controlla se il numero è associato ad un Pub. Dunque, se i dati di input sono corretti, verifica se il pub ha un'affiliazione con un parcheggio: se il controllo va a buon fine restituisce un messaggio che indica al cliente la posizione dal parcheggio e la distanza dal pub.

```
CREATE OR REPLACE PROCEDURE PARK(TEL__PUB IN VARCHAR2)
IS
DIST NUMBER(4);
TEMP NUMBER(1);
CIT VARCHAR2(255);
V VARCHAR2(255);
CP VARCHAR2(255);
VP VARCHAR2(255);
ERRORE EXCEPTION;
BEGIN

SELECT COUNT(*) INTO TEMP FROM PUB WHERE TELEFONO_PUB=TEL__PUB;

IF TEMP=0 THEN
RAISE ERRORE;
END IF;

DIST:=0;
SELECT CITTA , VIA INTO CP, VP
FROM PUB
WHERE TELEFONO_PUB=TEL__PUB;
SELECT COUNT(*) INTO TEMP FROM PARCHEGGIO JOIN PUB ON N_PUB_ASSOCIATO=TELEFONO_PUB
WHERE TELEFONO_PUB=TEL__PUB;

IF TEMP=0 THEN
DBMS_OUTPUT.PUT_LINE( 'IL PUB DI ' ||CP|| ', IN ' ||VP|| ' NON HA ANCORA UN
PARCHEGGIO AFFILIATO.' );
ELSE
SELECT DISTANZA_PUB, CITTA, VIA INTO DIST, CIT, V
FROM PARCHEGGIO
WHERE N_PUB_ASSOCIATO=TEL__PUB;
DBMS_OUTPUT.PUT_LINE( 'IL PARCHEGGIO PIU VICINO AL PUB DI ' ||CP|| ' IN ' ||VP|| '
DISTA ' ||DIST|| ' METRI, SI TROVA A ' ||CIT|| ', ' ||V);
END IF;

EXCEPTION
WHEN ERRORE THEN DBMS_OUTPUT.PUT_LINE( 'IL PUB SELEZIONATO NON ESISTE.' );
END;
/
--EXECUTE PARK('NUMERO DI TELEFONO DEL PUB');
```

PROCEDURA CHE PERMETTE AL GESTORE DELLE FORNITURE DI SAPERE LE SCORTE ATTUALI DEL PUB

Questa procedura controlla la scorta di un pub relativa ad un prodotto selezionato.

Che sia una materia prima o una bibita il concetto è lo stesso: si ricerca prima la quantità consumata del prodotto e poi la quantità delle forniture e si ottiene la scorta facendo acquistati-consumati:

```
CREATE OR REPLACE PROCEDURE SCORTA(TEL__PUB IN VARCHAR2, MAT IN VARCHAR2)
IS
CONSUMATO NUMBER(9,4);
ACQUISTO NUMBER(4);
SCORTE NUMBER(9,4);
TEMP NUMBER(1);
ERRORE EXCEPTION;
BEGIN
--CONTROLLLO SE IL PUB ESISTE
SELECT COUNT(*) INTO TEMP FROM PUB WHERE TELEFONO_PUB=TEL__PUB;
IF TEMP=0 THEN
RAISE ERRORE;
END IF;
--CONTROLLLO SE SI VUOLE UNA BIBITA O UNA MATERIA PRIMA
SELECT COUNT(*) INTO TEMP FROM MATERIA_PRIMA WHERE NOME_MATERIA_PRIMA=MAT ;
IF TEMP=0 THEN
--QUERY PER BIBITE CONSUMATE
select
SUM (QUANTITA) INTO CONSUMATO
FROM (((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=
NUM_ORD)JOIN MENU ON NOM_MENU=NOME_MENU )JOIN BIBITA ON N_MENU_BIBITA=NOME_MENU
WHERE RECAPITO_PUB=TEL__PUB AND N_MENU_BIBITA=MAT
GROUP BY N_MENU_BIBITA;
--QUERY PER BIBITE ACQUISTATE
SELECT
SUM(QUANTITA_ACQUISTO) INTO ACQUISTO
FROM PUB JOIN FORNITURA ON TELEF_PUB=TELEFONO_PUB
WHERE TELEFONO_PUB=TEL__PUB AND N_BIBITA=MAT;
--CALCOLO SCORTE
SCORTE := ACQUISTO-CONSUMATO;
DBMS_OUTPUT.PUT_LINE( 'IL PUB SELEZIONATO POSSIEDE ' ||SCORTE|| ' SCORTE DI:
' ||MAT);
ELSE
--QUERY PER MATERIE PRIME CONSUMATE
select
rtrim(to_char((COUNT (NOME_MATERIA_PRIMA)*QUANTITA_INGREDIENTE), 'FM90.999'), '.')
AS SCORTE_CONSUMATE_KG
INTO CONSUMATO
from(
SELECT *
```

```

FROM (((((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=
NUM_ORD) JOIN MENU ON NOM_MENU=NOME_MENU ) JOIN PANINO ON N_MENU_PANINO=NOME_MENU)
JOIN INGREDIENTE
ON N_MENU=N_MENU_PANINO) JOIN MATERIA_PRIMA ON NOME_MATERIA_PRIMA=COD_MP
WHERE RECAPITO_PUB=TEL__PUB AND NOME_MATERIA_PRIMA=MAT )
GROUP BY NOME_MATERIA_PRIMA,QUANTITA_INGREDIENTE
ORDER BY NOME_MATERIA_PRIMA;
--QUERY PER MATERIE PRIME ACQUISTATE
SELECT
rtrim(to_char(SUM(QUANTITA_ACQUISTO), 'FM900.99'), '.') AS QUANTITA_ACQUISTO_KG
INTO ACQUISTO
FROM(
SELECT *
FROM (PUB JOIN FORNITURA ON TELEF_PUB=TELEFONO_PUB) JOIN MATERIA_PRIMA ON
COD_MAT_PR=NOME_MATERIA_PRIMA
WHERE TELEFONO_PUB=TEL__PUB AND NOME_MATERIA_PRIMA=MAT);
--CALCOLO SCORTE
SCORTE := ACQUISTO-CONSUMATO;
DBMS_OUTPUT.PUT_LINE( 'IL PUB SELEZIONATO POSSIEDE ' ||SCORTE||' KG DI SCORTE DI:
' ||MAT);
END IF;
EXCEPTION
WHEN ERRORE THEN DBMS_OUTPUT.PUT_LINE( 'ERRORE NEI DATI DI INPUT. ');
END;
/
--EXECUTE SCORTA('NUMERO DI TELEFONO DEL PUB', 'MATERIA PRIMA CORRENTE')

```

PROCEDURA CHE PERMETTE AI DIPENDENTI DI UN PUB DI AVERE UN BONUS SULLO STIPENDIO SE SI RAGGIUNGE UN OBIETTIVO MINIMO DI VENDITE(AL NETTO DEGLI SCONTI)

```
CREATE OR REPLACE PROCEDURE BONUS_MESE( TELEFONO_PUBB IN VARCHAR2, MESE IN
VARCHAR2, OBIETTIVO IN NUMBER,BONUS NUMBER) IS
TOTALE_MENS NUMBER ;
BEGIN
SELECT SUM(TOTALE_CONTO) INTO TOTALE_MENS FROM(
SELECT COD_ORDINAZIONE, SUM(PREZZO*QUANTITA) AS TOTALE_CONTO
FROM ((CONTO JOIN ORDINAZIONE ON ID_ORDINE=COD_ORDINAZIONE) JOIN COMPRENDE ON
COD_ORDINAZIONE=NUM_ORD) JOIN MENU ON NOM_MENU=NOME_MENU
WHERE RECAPITO_PUB=TELEFONO_PUBB AND TO_CHAR(ORA_DATA_SCONTRINO, 'MM/YYYY')=MESE
GROUP BY ID_ORDINE);
IF TOTALE_MENS>= OBIETTIVO THEN
DBMS_OUTPUT.put_line('ABBIAMO RAGGIUNTO L OBIETTIVO MENSILE INCASSANDO '
||TOTALE_MENS||' AL NETTO DEGLI SCONTI APPLICATI. COMPLIMENTI! ALZO LO STIPENDIO DI
'||BONUS||' EURO');
UPDATE (SELECT IMPORTO FROM STIPENDIO JOIN DIPENDENTE ON CF=CF_DIP WHERE
TELEFN_PUB=TELEFONO_PUBB AND TO_CHAR (DATA_ACCREDITO, 'MM/YYYY')=MESE)SET
IMPORTO=IMPORTO+BONUS;
ELSE
TOTALE_MENS:=OBIETTIVO-TOTALE_MENS;
DBMS_OUTPUT.put_line('NON ABBIAMO RAGGIUNTO L OBIETTIVO MENSILE PER
'||TOTALE_MENS||' PECCATO!');
END IF;
END;
/
--EXEC BONUS_MESE('TELEFONO PUB', 'MM/YYYY', 'OBIETTIVO', 'BONUS');
```

5.6 Popolamento fittizio del DB

Infine, vi riporto l'inserimento dei dati fittizi all'interno del DB:

INSERIMENTO PUB

--Creo 3 pub diversi

```
INSERT INTO PUB (TELEFONO_PUB, ORARIO_APERTURA, ORARIO_CHIUSURA, VIA, CITTA,
PROVINCIA) VALUES ('0815745838', TO_DATE('12:30', 'HH24:MI'), TO_DATE('02:10',
'HH24:MI'), 'VIA SARDI', 'ERCOLANO', 'NA');
INSERT INTO PUB (TELEFONO_PUB, ORARIO_APERTURA, ORARIO_CHIUSURA, VIA, CITTA,
PROVINCIA) VALUES ('0816317485', TO_DATE('12:30', 'HH24:MI'), TO_DATE('02:10',
'HH24:MI'), 'VIA SIVIGLIA', 'SPOLETO', 'PG');
INSERT INTO PUB (TELEFONO_PUB, ORARIO_APERTURA, ORARIO_CHIUSURA, VIA, CITTA,
PROVINCIA) VALUES ('0814103527', TO_DATE('12:30', 'HH24:MI'), TO_DATE('02:10',
'HH24:MI'), 'VIA CATALANA', 'ALGHERO', 'SS');
```

INSERIMENTO MENU

--CREO IL MENU, CHE SARÀ UGUALE IN TUTTI I PUB

```
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('IRON MAN', 15);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('HULK', 10);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('DOCTOR STRANGE', 12);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('CAPITAN AMERICA', 14);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('THOR', 13);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('SPIDER MAN', 11);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('SCARLET WITCH', 8);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('COCA COLA', 3);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('PERONI', 3);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('FANTA', 3);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('SPRITE', 3);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('ACQUA FRIZZANTE', 1);
INSERT INTO MENU (NOME_MENU, PREZZO) VALUES ('ACQUA NATURALE', 1);
```

INSERIMENTO BIBITA

--SPECIALIZZAZIONE BIBITA

```
INSERT INTO BIBITA (N_MENU_BIBITA, ALCOLICO) VALUES ('COCA COLA', 'NO');
INSERT INTO BIBITA (N_MENU_BIBITA, ALCOLICO) VALUES ('PERONI', 'SI');
INSERT INTO BIBITA (N_MENU_BIBITA, ALCOLICO) VALUES ('FANTA', 'NO');
INSERT INTO BIBITA (N_MENU_BIBITA, ALCOLICO) VALUES ('SPRITE', 'NO');
INSERT INTO BIBITA (N_MENU_BIBITA, ALCOLICO) VALUES ('ACQUA FRIZZANTE', 'NO');
INSERT INTO BIBITA (N_MENU_BIBITA, ALCOLICO) VALUES ('ACQUA NATURALE', 'NO');
```

INSERIMENTO PANINO

--SPECIALIZZAZIONE PANINO

```
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('IRON MAN');
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('HULK');
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('CAPITAN AMERICA');
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('THOR');
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('DOCTOR STRANGE');
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('SPIDER MAN');
INSERT INTO PANINO (N_MENU_PANINO) VALUES ('SCARLET WITCH');
```

INSERIMENTO MATERIA PRIMA

--MATERIA PRIMA PANINO

```
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('PANINO');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('HAMBURGER');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('LATTUGA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('COTOLETTA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('PORCHETTA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('SALSICCIA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('BECON');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('UOVO');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('SOTTILETTA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('PROVOLA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('POMODORO');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('PIADINA');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('PATATINE');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('CETRIOLO');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('HAMBURGER - VEGETARIANO');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('FRIARIELLI');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('PEPERONI');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('FUNGHI CHAMPIGNON');
INSERT INTO MATERIA_PRIMA (NOME_MATERIA_PRIMA) VALUES ('CHEDDAR');
```


INSERIMENTO INGREDIENTE

--I PANINI DEI PUB SONO COMPOSTI DAI SEGUENTI INGREDIENTI

```
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('IRON MAN',
'PANINO', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('IRON MAN',
'POMODORO', 0.04);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('IRON MAN',
'HAMBURGER', 0.22);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('IRON MAN',
'PATATINE', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('IRON MAN',
'LATTUGA', 0.08);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('HULK',
'PANINO', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('HULK',
'HAMBURGER - VEGETARIANO', 0.20);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('HULK',
'LATTUGA', 0.08);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('HULK',
'CETRIOLO', 0.03);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('HULK',
'PEPERONI', 0.1);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('CAPITAN
AMERICA', 'PANINO', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('CAPITAN
AMERICA', 'SALSICCIA', 0.22);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('CAPITAN
AMERICA', 'PROVOLA', 0.06);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('CAPITAN
AMERICA', 'FRIARIELLI', 0.09);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('THOR',
'PANINO', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('THOR',
'PORCHETTA', 0.2);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('THOR',
'PATATINE', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('DOCTOR
STRANGE', 'PANINO', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('DOCTOR
STRANGE', 'PORCHETTA', 0.2);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('DOCTOR
STRANGE', 'SOTTILETTA', 0.025);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('DOCTOR
STRANGE', 'FUNGHI CHAMPIGNON', 0.12);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SPIDER
MAN', 'PANINO', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SPIDER
MAN', 'COTOLETTA', 0.22);
```

```

INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SPIDER
MAN', 'POMODORO', 0.04);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SPIDER
MAN', 'PATATINE', 0.15);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SPIDER
MAN', 'UOVO', 0.05);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SCARLET
WITCH', 'PIADINA', 0.1);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SCARLET
WITCH', 'BECON', 0.1);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SCARLET
WITCH', 'LATTUGA', 0.1);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SCARLET
WITCH', 'POMODORO', 0.05);
INSERT INTO INGREDIENTE (N_MENU, COD_MP, QUANTITA_INGREDIENTE) VALUES ('SCARLET
WITCH', 'CHEDDAR', 0.05);

```

INSERIMENTO PARCHEGGIO

--QUESTI SONO I PARCHEGGI CHE HANNO UNA PARTNER COL PUB

```

INSERT INTO PARCHEGGIO (VIA, CITTA, COSTO_POSTO_ORA, ORA_APERTURA, ORA_CHIUSURA,
NOME_PARCHEGGIO, TOTALE_POSTI_RISERVATI, DISTANZA_PUB, N_PUB_ASSOCIATO) VALUES
('VIA LOS SANTOS', 'ERCOLANO', 1, TO_DATE('06:00', 'HH24:MI'), TO_DATE('03:00',
'HH24:MI'), 'PARKING ERCOLANO', 30, 50, '0815745838');
INSERT INTO PARCHEGGIO (VIA, CITTA, COSTO_POSTO_ORA, ORA_APERTURA, ORA_CHIUSURA,
NOME_PARCHEGGIO, TOTALE_POSTI_RISERVATI, DISTANZA_PUB, N_PUB_ASSOCIATO) VALUES
('VIA SIVIGLIA', 'SPOLETO', 1, TO_DATE('07:00', 'HH24:MI'), TO_DATE('02:00',
'HH24:MI'), 'PARKING SPOLETO', 20, 20, '0816317485');

```

INSERIMENTO CLIENTE

--INSERISCO I CLIENTI DEL PRIMO PUB

```

INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3758523303', 'MIRIAM', 'AURIA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3985464623', 'ARMANDO', 'SANTO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3425974638', 'FRANCESCO', 'AMATO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3284576231', 'GIUSEPPE', 'CASSANDRO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3592742975', 'GIOVANNI', 'ONOFRIO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3412052324', 'ARIANA', 'NOVIELLO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0813428854', 'JESSICA', 'ROGHI');

```

```

INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3357523176', 'ROBERTO', 'RAVENNA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3327419820', 'RAIMONDO', 'ZUDDAS');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0812109721', 'LORENZO', 'MAZZA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0814732014', 'ENRICO', 'MADONNA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0811465976', 'MADDALENA', 'SANTIAGO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3745892010', 'BENIAMINO', 'NARDONE');
--I CLIENTI DEL SECONDO PUB
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3758523302', 'BRUNO', 'VELLO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3985464622', 'ALESSANDRO', 'BOVE');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3425974632', 'MARIO', 'AMATO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3284576232', 'GIUSEPPE', 'COZZOLINO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3592742972', 'GIOVANNI', 'VINCENZINI');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3412052322', 'ARIANA', 'GRANDE');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0813428852', 'TONY', 'STARK');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3357523172', 'BRITNEY', 'RAVENNA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3327419822', 'RAIMONDO', 'PIZO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0812109722', 'ASH', 'KETCHUM');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0814732012', 'ERRICO', 'MADONA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0811465972', 'VERONICA', 'IMPERATO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3745892012', 'MATILDA', 'SUSSONE');
--I CLIENTI DEL TERZO PUB
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3758523399', 'LUIGI', 'PIENO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3985464699', 'GESUALDO', 'LO MONACO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3425974699', 'ANDREA', 'GALEAZZI');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3284576299', 'PEPPE', 'LINO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3592742999', 'GIOVANNA', 'PIACENZINI');

```

```

INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3412052399', 'MILO', 'PICCOLO');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0813428899', 'TOBEY', 'MCGUIRE');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3357523199', 'NICO', 'SCHIRA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3327419899', 'MICHELA', 'PIZONNA');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0812109799', 'ANDREW', 'GARFIELD');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0814732099', 'TOMMASO', 'HOLLAND');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('0811465999', 'MICHELE', 'SCOTTI');
INSERT INTO CLIENTE (RECAPITO_TELEFONICO, NOME_CLIENTE, COGNOME_CLIENTE) VALUES
('3745892099', 'GERRY', 'MENTONE');

```

INSERIMENTO C SUL POSTO

--SPECIALIZZAZIONE C SUL POSTO PRIMO PUB

```

INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3758523303');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3985464623');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3425974638');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3284576231');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3592742975');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3412052324');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3357523176');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3327419820');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3745892010');

```

--SPECIALIZZAZIONE C SUL POSTO SECONDO PUB

```

INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3758523302');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3985464622');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3425974632');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3284576232');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3592742972');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3412052322');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3357523172');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3327419822');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3745892012');

```

--SPECIALIZZAZIONE C SUL POSTO TERZO PUB

```

INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3758523399');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3985464699');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3425974699');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3284576299');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3592742999');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3412052399');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3357523199');
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3327419899');

```

```
INSERT INTO C_SUL_POSTO (REC_TELEFONICO_SULPOSTO) VALUES ('3745892099');
```

INSERIMENTO C D'ASPORTO

--SPECIALIZZAZIONE C ASPORTO PRIMO PUB

```
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0813428854');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0812109721');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0814732014');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0811465976');
```

--SPECIALIZZAZIONE C ASPORTO SECONDO PUB

```
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0813428852');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0812109722');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0814732012');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0811465972');
```

--SPECIALIZZAZIONE C ASPORTO TERZO PUB

```
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0813428899');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0812109799');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0814732099');
INSERT INTO C_D_ASORTO (REC_TELEFONICO_DASPORTO) VALUES ('0811465999');
```

INSERIMENTO TESSERA

--TESSERE PRIMO PUB

```
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1001', TO_DATE('11/12/1993',
'DD/MM/YYYY'), 'MIRIAM.AURIA@GMAIL.COM', '3758523303');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1002', TO_DATE('11/01/1992',
'DD/MM/YYYY'), 'ARMANDO.SANTO@GMAIL.COM', '3985464623');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1003', TO_DATE('21/11/1991',
'DD/MM/YYYY'), 'FRANCESCO.AMATO@GMAIL.COM', '3425974638');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1004', TO_DATE('21/12/2001',
'DD/MM/YYYY'), 'GIUSEPPE.CASSANDRO@GMAIL.COM', '3284576231');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1005', TO_DATE('01/11/1994',
'DD/MM/YYYY'), 'GIOVANNI.ONOFRIO@GMAIL.COM', '3592742975');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1006', TO_DATE('11/11/1992',
'DD/MM/YYYY'), 'ARIANA.NOVIELLO@GMAIL.COM', '3412052324');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1007', TO_DATE('02/02/1991',
'DD/MM/YYYY'), 'JESSICA.ROGHI@GMAIL.COM', '0813428854');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1009', TO_DATE('14/02/1983',
'DD/MM/YYYY'), 'ROBERTO.RAVENNA@GMAIL.COM', '3357523176');
```

```

INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A1010', TO_DATE('11/02/1981',
'DD/MM/YYYY'), 'RAIMONDO.ZUDDAS@GMAIL.COM', '3327419820');
--TESSERE SECONDO PUB
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2001', TO_DATE('17/12/1992',
'DD/MM/YYYY'), 'BRUN.VELLO@GMAIL.COM', '3758523302');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2002', TO_DATE('14/03/1994',
'DD/MM/YYYY'), 'ALE.BOVE@GMAIL.COM', '3985464622');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2003', TO_DATE('25/05/1998',
'DD/MM/YYYY'), 'MARIO.AMATO@GMAIL.COM', '3425974632');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2004', TO_DATE('22/11/2000',
'DD/MM/YYYY'), 'GIUSEPPE.COZZ@GMAIL.COM', '3284576232');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2005', TO_DATE('06/06/1996',
'DD/MM/YYYY'), 'GIOVANNI.VINC@GMAIL.COM', '3592742972');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2006', TO_DATE('13/01/1999',
'DD/MM/YYYY'), 'ARIANA.BIG@GMAIL.COM', '3412052322');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2007', TO_DATE('02/02/1990',
'DD/MM/YYYY'), 'IRON.MAN@GMAIL.COM', '0813428852');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2009', TO_DATE('30/05/1989',
'DD/MM/YYYY'), 'ITSBRITNEY.BEACH@GMAIL.COM', '3357523172');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A2010', TO_DATE('12/09/1985',
'DD/MM/YYYY'), 'ASH.KK@GMAIL.COM', '3327419822');
--TESSERE TERZO PUB
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3001', TO_DATE('13/12/1991',
'DD/MM/YYYY'), 'LULLI.PIENO@GMAIL.COM', '3758523399');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3002', TO_DATE('12/03/1992',
'DD/MM/YYYY'), 'GESU.LOM@GMAIL.COM', '3985464699');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3003', TO_DATE('26/05/1993',
'DD/MM/YYYY'), 'ANDREA.GALE@GMAIL.COM', '3425974699');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3004', TO_DATE('27/11/2002',
'DD/MM/YYYY'), 'PEPPE.LIN@GMAIL.COM', '3284576299');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3005', TO_DATE('05/06/1992',
'DD/MM/YYYY'), 'GIOVANNA.PIA@GMAIL.COM', '3592742999');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3006', TO_DATE('1/01/1993',
'DD/MM/YYYY'), 'MILO.BAUDOG@GMAIL.COM', '3412052399');

```



```

INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3007', TO_DATE('04/02/1993',
'DD/MM/YYYY'), 'IMTHEBESTSPIDERMEN@GMAIL.COM', '0813428899');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3009', TO_DATE('23/05/1985',
'DD/MM/YYYY'), 'FANTANICO@GMAIL.COM', '3357523199');
INSERT INTO TESSERA (CODICE_TESSERA, DATA_COMPLEANNO_CLIENTE, EMAIL,
RECAPITO_CLIENTE) VALUES ('A3010', TO_DATE('13/09/1984',
'DD/MM/YYYY'), 'MICKELA@GMAIL.COM', '3327419899');

```

INSERIMENTO DIPENDENTE

```

--PER OGNI PUB INSERISCO 7 DIPENDENTI (3 camerieri, 2 cuochi, 2 rider)
--DIPENDENTI PRIMO PUB
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1CAM0000000001', 'ROBERTA', 'GILARDI',
TO_DATE('14/06/2020', 'DD/MM/YYYY'), TO_DATE('16/06/1997', 'DD/MM/YYYY'), 'F',
'0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1CAM0000000002', 'ALESSANDRO',
'TREGLIA', TO_DATE('1/05/2021', 'DD/MM/YYYY'), TO_DATE('04/04/2001', 'DD/MM/YYYY'),
'M', '0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1CAM0000000003', 'MARIA', 'ZAGABRIA',
TO_DATE('14/08/2022', 'DD/MM/YYYY'), TO_DATE('03/05/1990', 'DD/MM/YYYY'), 'F',
'0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1CAM0000000004', 'GERMANO', 'SONDI',
TO_DATE('1/01/2021', 'DD/MM/YYYY'), TO_DATE('20/10/1998', 'DD/MM/YYYY'), 'M',
'0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1CU0000000005', 'TERESA', 'GARCIA',
TO_DATE('4/05/2018', 'DD/MM/YYYY'), TO_DATE('14/07/1999', 'DD/MM/YYYY'), 'F',
'0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1CU0000000006', 'GIACOMO', 'REVILLA',
TO_DATE('13/05/2018', 'DD/MM/YYYY'), TO_DATE('10/05/1990', 'DD/MM/YYYY'), 'M',
'0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE,
COGNOME_DIPENDENTE, DATA_ASSUNZIONE, DATA_NASCITA, SESSO, TELEFN_PUB) VALUES
('CF1RID0000000007', 'VALERIA', 'LUNDINI', TO_DATE('17/10/2018', 'DD/MM/YYYY'),
TO_DATE('14/05/1996', 'DD/MM/YYYY'), 'F', '0815745838');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF1RID0000000008', 'VIKTOR', 'UMBRELLA',
TO_DATE('18/09/2022', 'DD/MM/YYYY'), TO_DATE('14/05/1996', 'DD/MM/YYYY'), 'NB',
'0815745838');
--DIPENDENTI SECONDO PUB

```

```

INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2CAM0000000001', 'VANIA', 'UMBRELLINI',
TO_DATE('12/05/2020', 'DD/MM/YYYY'), TO_DATE('16/06/1987', 'DD/MM/YYYY'), 'NB',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2CAM0000000002', 'ROSANNA', 'EHLIS',
TO_DATE('11/03/2018', 'DD/MM/YYYY'), TO_DATE('04/04/2000', 'DD/MM/YYYY'), 'F',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2CAM0000000003', 'MARIANNA', 'ZIZZINI',
TO_DATE('18/06/2020', 'DD/MM/YYYY'), TO_DATE('03/05/1998', 'DD/MM/YYYY'), 'F',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2CAM0000000004', 'ROCCO', 'GERMANI',
TO_DATE('11/06/2021', 'DD/MM/YYYY'), TO_DATE('20/10/1998', 'DD/MM/YYYY'), 'M',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2CU0000000005', 'ELVIRA', 'MOSCONI',
TO_DATE('14/05/2021', 'DD/MM/YYYY'), TO_DATE('14/07/1993', 'DD/MM/YYYY'), 'F',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2CU0000000006', 'JACK', 'REVONE',
TO_DATE('4/04/2020', 'DD/MM/YYYY'), TO_DATE('10/05/1980', 'DD/MM/YYYY'), 'M',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2RID0000000007', 'EMANUELA', 'FANELLI',
TO_DATE('15/05/2018', 'DD/MM/YYYY'), TO_DATE('14/05/1999', 'DD/MM/YYYY'), 'F',
'0816317485');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF2RID0000000008', 'BEN', 'MILIONE',
TO_DATE('14/06/2021', 'DD/MM/YYYY'), TO_DATE('14/05/1990', 'DD/MM/YYYY'), 'M',
'0816317485');
--DIPENDENTI TERZO PUB
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3CAM0000000001', 'STEFANO', 'PROPHET',
TO_DATE('13/05/2020', 'DD/MM/YYYY'), TO_DATE('16/06/1987', 'DD/MM/YYYY'), 'M',
'0814103527');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3CAM0000000002', 'BIAGIO', 'COZZOLINI',
TO_DATE('2/02/2017', 'DD/MM/YYYY'), TO_DATE('04/04/2000', 'DD/MM/YYYY'), 'NB',
'0814103527');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3CAM0000000003', 'CARLO', 'PICCINO',
TO_DATE('13/04/2021', 'DD/MM/YYYY'), TO_DATE('03/05/1998', 'DD/MM/YYYY'), 'M',
'0814103527');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3CAM0000000004', 'ALVIN', 'FRANCIOSA',
TO_DATE('12/09/2022', 'DD/MM/YYYY'), TO_DATE('20/10/1998', 'DD/MM/YYYY'), 'M',
'0814103527');

```



```

INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3CU00000000005', 'MASSIMO',
'FRANCIONNA', TO_DATE('16/05/2020', 'DD/MM/YYYY'), TO_DATE('14/07/1993',
'DD/MM/YYYY'), 'M', '0814103527');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3CU00000000006', 'ANNA', 'ROSSINI',
TO_DATE('12/03/2019', 'DD/MM/YYYY'), TO_DATE('10/05/1980', 'DD/MM/YYYY'), 'F',
'0814103527');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3RID00000000007', 'TORE',
'FORMUSACCHIO', TO_DATE('13/05/2018', 'DD/MM/YYYY'), TO_DATE('14/05/1999',
'DD/MM/YYYY'), 'M', '0814103527');
INSERT INTO DIPENDENTE (CF, NOME_DIPENDENTE, COGNOME_DIPENDENTE, DATA_ASSUNZIONE,
DATA_NASCITA, SESSO, TELEFN_PUB) VALUES ('CF3RID00000000008', 'SABRINA', 'MAFRESCA',
TO_DATE('1/06/2021', 'DD/MM/YYYY'), TO_DATE('14/05/1990', 'DD/MM/YYYY'), 'F',
'0814103527');

```

INSERIMENTO CAMERIERE

```

--CAMERIERI PRIMO PUB
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF1CAM00000000001');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF1CAM00000000002');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF1CAM00000000003');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF1CAM00000000004');
--CAMERIERI SECONDO PUB
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF2CAM00000000001');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF2CAM00000000002');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF2CAM00000000003');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF2CAM00000000004');
--CAMERIERI TERZO PUB
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF3CAM00000000001');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF3CAM00000000002');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF3CAM00000000003');
INSERT INTO CAMERIERE (CF_CAMERIERE) VALUES ('CF3CAM00000000004');

```

INSERIMENTO CUOCHO

```

--CUOCHI PRIMO PUB
INSERT INTO CUOCO (CF_CUOCO) VALUES ('CF1CU00000000005');
INSERT INTO CUOCO (CF_CUOCO) VALUES ('CF1CU00000000006');
--CUOCHI SECONDO PUB
INSERT INTO CUOCO (CF_CUOCO) VALUES ('CF2CU00000000005');
INSERT INTO CUOCO (CF_CUOCO) VALUES ('CF2CU00000000006');
--CUOCHI TERZO PUB
INSERT INTO CUOCO (CF_CUOCO) VALUES ('CF3CU00000000005');
INSERT INTO CUOCO (CF_CUOCO) VALUES ('CF3CU00000000006');

```

INSERIMENTO RIDER

--RIDER PRIMO PUB

```
INSERT INTO RIDER (CF_RIDER) VALUES ('CF1RID0000000007');  
INSERT INTO RIDER (CF_RIDER) VALUES ('CF1RID0000000008');
```

--RIDER SECONDO PUB

```
INSERT INTO RIDER (CF_RIDER) VALUES ('CF2RID0000000007');  
INSERT INTO RIDER (CF_RIDER) VALUES ('CF2RID0000000008');
```

--RIDER TERZO PUB

```
INSERT INTO RIDER (CF_RIDER) VALUES ('CF3RID0000000007');  
INSERT INTO RIDER (CF_RIDER) VALUES ('CF3RID0000000008');
```

INSERIMENTO STIPENDIO

--STIPENDI DIPENDENTI PRIMO PUB

```
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1CAM0000000001', 1400, 140);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1CAM0000000003', 1500, 150);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1CAM0000000002', 900, 90);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1CAM0000000004', 1200, 120);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1CU0000000005', 1900, 190);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1CU0000000006', 2000, 200);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1RID0000000007', 1800, 180);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF1RID0000000008', 1400, 180);
```

--STIPENDI SECONDO PUB

```
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2CAM0000000001', 1300, 140);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2CAM0000000003', 1400, 150);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2CAM0000000002', 950, 90);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2CAM0000000004', 1400, 120);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2CU0000000005', 1400, 190);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2CU0000000006', 2000, 200);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2RID0000000007', 1700, 180);  
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES  
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF2RID0000000008', 1500, 180);
```

--STIPENDI TERZO PUB

```

INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3CAM0000000001', 1500, 140);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3CAM0000000003', 1600, 150);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3CAM0000000002', 1550, 90);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3CAM0000000004', 1600, 120);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3CU0000000005', 1900, 190);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3CU0000000006', 1900, 200);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3RID0000000007', 1400, 180);
INSERT INTO STIPENDIO (DATA_ACCREDITO, CF_DIP, IMPORTO, TRATTENUTE) VALUES
(TO_DATE('25/06/2022', 'DD/MM/YYYY'), 'CF3RID0000000008', 1100, 180);

```

INSERIMENTO TAVOLO

--TAVOLI PRIMO PUB

```

INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('1', '1',
'0815745838', 2);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('2', '1',
'0815745838', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('3', '1',
'0815745838', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('4', '1',
'0815745838', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('5', '2',
'0815745838', 4);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('6', '2',
'0815745838', 4);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('7', '2',
'0815745838', 5);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('8', '3',
'0815745838', 6);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('9', '3',
'0815745838', 6);

```

--TAVOLI SECONDO PUB

```

INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('1', '1',
'0816317485', 2);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('2', '1',
'0816317485', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('3', '1',
'0816317485', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('4', '1',
'0816317485', 3);

```

```

INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('5', '2',
'0816317485', 4);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('6', '2',
'0816317485', 4);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('7', '2',
'0816317485', 5);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('8', '3',
'0816317485', 6);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('9', '3',
'0816317485', 6);
--TAVOLI TERZO PUB
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('1', '1',
'0814103527', 2);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('2', '1',
'0814103527', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('3', '1',
'0814103527', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('4', '1',
'0814103527', 3);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('5', '2',
'0814103527', 4);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('6', '2',
'0814103527', 4);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('7', '2',
'0814103527', 5);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('8', '3',
'0814103527', 6);
INSERT INTO TAVOLO (NUMERO_TAVOLO, NUM_SALA, TE_PUB, NUM_POSTI) VALUES ('9', '3',
'0814103527', 6);

```

INSERIMENTO TURNO DI LAVORO

```

--TURNI DI LAVORO PRIMO PUB
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF1CAM0000000001', 6,
'0815745838');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF1CAM0000000002', 8,
'0815745838');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF1CU0000000005', 6,
'0815745838');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF1CAM0000000003', 8,
'0815745838');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF1CAM0000000004', 8,
'0815745838');

```

```

INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF1CU00000000006', 8,
'0815745838');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF1RID0000000007', 5,
'0815745838');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF1RID0000000008', 8,
'0815745838');
--TURNI DI LAVORO SECONDO PUB
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF2CAM0000000001', 6,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF2CAM0000000002', 8,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF2CU0000000005', 6,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF2CAM0000000003', 8,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF2CAM0000000004', 8,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF2CU0000000006', 8,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF2RID0000000007', 5,
'0816317485');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF2RID0000000008', 8,
'0816317485');
--TURNI DI LAVORO TERZO PUB
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF3CAM0000000001', 7,
'0814103527');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:30', 'DD/MM/YYYY HH24:MI'), 'CF3CAM0000000002', 4,
'0814103527');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 12:00', 'DD/MM/YYYY HH24:MI'), 'CF3CU0000000005', 7,
'0814103527');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:40', 'DD/MM/YYYY HH24:MI'), 'CF3CAM0000000003', 4,
'0814103527');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:40', 'DD/MM/YYYY HH24:MI'), 'CF3CAM0000000004', 8,
'0814103527');

```

```

INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:40', 'DD/MM/YYYY HH24:MI'), 'CF3CU0000000006', 8,
'0814103527');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF3RID0000000007', 5,
'0814103527');
INSERT INTO TURNO_DI_LAVORO (DATA_ORA_TURNO, CF_DIPENDENTE, DURATA, T_PUB) VALUES
(TO_TIMESTAMP('10/06/22 18:00', 'DD/MM/YYYY HH24:MI'), 'CF3RID0000000008', 8,
'0814103527');

```

INSERIMENTO PRENOTAZIONE

-- PRENOTAZIONI DEL PRIMO PUB

```

INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 12:14', 'DD/MM/YYYY HH24:MI'), '3758523303',
'4', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 13:30', 'DD/MM/YYYY HH24:MI'), '3985464623',
'4', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:20', 'DD/MM/YYYY HH24:MI'), '3425974638',
'2', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:14', 'DD/MM/YYYY HH24:MI'), '3284576231',
'1', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 18:20', 'DD/MM/YYYY HH24:MI'), '3592742975',
'2', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:30', 'DD/MM/YYYY HH24:MI'), '3412052324',
'5', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 14:14', 'DD/MM/YYYY HH24:MI'), '3357523176',
'1', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:45', 'DD/MM/YYYY HH24:MI'), '3327419820',
'3', '0815745838');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), '3745892010',
'4', '0815745838');

```

-- PRENOTAZIONI DEL SECONDO PUB

```

INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 12:14', 'DD/MM/YYYY HH24:MI'), '3758523302',
'6', '0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 13:30', 'DD/MM/YYYY HH24:MI'), '3985464622',
'6', '0816317485');

```



```

INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:20', 'DD/MM/YYYY HH24:MI'), '3425974632',
'4','0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:14', 'DD/MM/YYYY HH24:MI'), '3284576232',
'3','0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 18:20', 'DD/MM/YYYY HH24:MI'), '3592742972',
'5','0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:30', 'DD/MM/YYYY HH24:MI'), '3412052322',
'7','0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 14:14', 'DD/MM/YYYY HH24:MI'), '3357523172',
'3','0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:45', 'DD/MM/YYYY HH24:MI'), '3327419822',
'5','0816317485');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), '3745892012',
'6','0816317485');
-- PRENOTAZIONI DEL TERZO PUB
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 13:14', 'DD/MM/YYYY HH24:MI'), '3758523399',
'6','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 14:30', 'DD/MM/YYYY HH24:MI'), '3985464699',
'6','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 21:20', 'DD/MM/YYYY HH24:MI'), '3425974699',
'4','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:14', 'DD/MM/YYYY HH24:MI'), '3284576299',
'3','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 19:20', 'DD/MM/YYYY HH24:MI'), '3592742999',
'5','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), '3412052399',
'7','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 15:14', 'DD/MM/YYYY HH24:MI'), '3357523199',
'3','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 20:45', 'DD/MM/YYYY HH24:MI'), '3327419899',
'5','0814103527');
INSERT INTO PRENOTAZIONE (DATA_ORA_PRENOTAZIONE, RECPT_TEL, NUM_TAVOLO, TF_PUB)
VALUES (TO_TIMESTAMP('15/06/2022 21:30', 'DD/MM/YYYY HH24:MI'), '3745892099',
'6','0814103527');

```

INSERIMENTO BLOCCA (TRA PRENOTAZIONE E TAVOLO)

--tabella di transizione tra prenotazione e tavolo primo pub

```
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('4',
TO_TIMESTAMP('15/06/2022 12:14', 'DD/MM/YYYY HH24:MI'), '3758523303', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('4',
TO_TIMESTAMP('15/06/2022 13:30', 'DD/MM/YYYY HH24:MI'), '3985464623', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('2',
TO_TIMESTAMP('15/06/2022 20:20', 'DD/MM/YYYY HH24:MI'), '3425974638', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('1',
TO_TIMESTAMP('15/06/2022 19:14', 'DD/MM/YYYY HH24:MI'), '3284576231', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('2',
TO_TIMESTAMP('15/06/2022 18:20', 'DD/MM/YYYY HH24:MI'), '3592742975', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('5',
TO_TIMESTAMP('15/06/2022 19:30', 'DD/MM/YYYY HH24:MI'), '3412052324', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('1',
TO_TIMESTAMP('15/06/2022 14:14', 'DD/MM/YYYY HH24:MI'), '3357523176', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('3',
TO_TIMESTAMP('15/06/2022 19:45', 'DD/MM/YYYY HH24:MI'), '3327419820', '0815745838');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('4',
TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), '3745892010', '0815745838');
```

--tabella di transizione tra prenotazione e tavolo secondo pub

```
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('6',
TO_TIMESTAMP('15/06/2022 12:14', 'DD/MM/YYYY HH24:MI'), '3758523302', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('6',
TO_TIMESTAMP('15/06/2022 13:30', 'DD/MM/YYYY HH24:MI'), '3985464622', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('4',
TO_TIMESTAMP('15/06/2022 20:20', 'DD/MM/YYYY HH24:MI'), '3425974632', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('3',
TO_TIMESTAMP('15/06/2022 19:14', 'DD/MM/YYYY HH24:MI'), '3284576232', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('5',
TO_TIMESTAMP('15/06/2022 18:20', 'DD/MM/YYYY HH24:MI'), '3592742972', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('7',
TO_TIMESTAMP('15/06/2022 19:30', 'DD/MM/YYYY HH24:MI'), '3412052322', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('3',
TO_TIMESTAMP('15/06/2022 14:14', 'DD/MM/YYYY HH24:MI'), '3357523172', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('5',
TO_TIMESTAMP('15/06/2022 19:45', 'DD/MM/YYYY HH24:MI'), '3327419822', '0816317485');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('6',
TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), '3745892012', '0816317485');
```

--tabella di transizione tra prenotazione e tavolo terzo pub

```
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('6',
TO_TIMESTAMP('15/06/2022 13:14', 'DD/MM/YYYY HH24:MI'), '3758523399', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('6',
TO_TIMESTAMP('15/06/2022 14:30', 'DD/MM/YYYY HH24:MI'), '3985464699', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('4',
TO_TIMESTAMP('15/06/2022 21:20', 'DD/MM/YYYY HH24:MI'), '3425974699', '0814103527');
```



```

INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('3',
TO_TIMESTAMP('15/06/2022 20:14', 'DD/MM/YYYY HH24:MI'), '3284576299', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('5',
TO_TIMESTAMP('15/06/2022 19:20', 'DD/MM/YYYY HH24:MI'), '3592742999', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('7',
TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), '3412052399', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('3',
TO_TIMESTAMP('15/06/2022 15:14', 'DD/MM/YYYY HH24:MI'), '3357523199', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('5',
TO_TIMESTAMP('15/06/2022 20:45', 'DD/MM/YYYY HH24:MI'), '3327419899', '0814103527');
INSERT INTO BLOCCA (ID_TAVOLO, D_O_PRENOTA, RECPTO_TEL, TFN_PUB) VALUES ('6',
TO_TIMESTAMP('15/06/2022 21:30', 'DD/MM/YYYY HH24:MI'), '3745892099', '0814103527');

```

INSERIMENTO ORDINAZIONE

```

--ordinazioni al tavolo primo pub
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1001');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1002');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1003');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1004');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1005');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1006');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1007');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1008');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT1009');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD1001');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD1002');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD1003');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD1004');
--ordinazioni al tavolo secondo pub
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2001');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2002');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2003');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2004');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2005');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2006');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2007');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2008');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT2009');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD2001');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD2002');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD2003');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD2004');
--ordinazioni al tavolo terzo pub
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3001');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3002');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3003');

```

```

INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3004');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3005');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3006');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3007');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3008');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OT3009');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD3001');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD3002');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD3003');
INSERT INTO ORDINAZIONE (COD_ORDINAZIONE) VALUES ('OD3004');

```

INSERIMENTO ORDINAZIONE AL TAVOLO

--figli ordinazione (al tavolo) primo pub

```

INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1001', TO_TIMESTAMP('15/06/2022 12:14',
'DD/MM/YYYY HH24:MI'), '3758523303', '4', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1002', TO_TIMESTAMP('15/06/2022 13:30',
'DD/MM/YYYY HH24:MI'), '3985464623', '4', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1003', TO_TIMESTAMP('15/06/2022 20:20',
'DD/MM/YYYY HH24:MI'), '3425974638', '2', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1004', TO_TIMESTAMP('15/06/2022 19:14',
'DD/MM/YYYY HH24:MI'), '3284576231', '1', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1005', TO_TIMESTAMP('15/06/2022 18:20',
'DD/MM/YYYY HH24:MI'), '3592742975', '2', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1006', TO_TIMESTAMP('15/06/2022 19:30',
'DD/MM/YYYY HH24:MI'), '3412052324', '5', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1007', TO_TIMESTAMP('15/06/2022 14:14',
'DD/MM/YYYY HH24:MI'), '3357523176', '1', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1008', TO_TIMESTAMP('15/06/2022 19:45',
'DD/MM/YYYY HH24:MI'), '3327419820', '3', '0815745838');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT1009', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), '3745892010', '4', '0815745838');

```

--figli ordinazione (al tavolo) secondo pub

```

INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2001', TO_TIMESTAMP('15/06/2022 12:14',
'DD/MM/YYYY HH24:MI'), '3758523302', '6', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2002', TO_TIMESTAMP('15/06/2022 13:30',
'DD/MM/YYYY HH24:MI'), '3985464622', '6', '0816317485');

```

```

INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2003', TO_TIMESTAMP('15/06/2022 20:20',
'DD/MM/YYYY HH24:MI'), '3425974632', '4', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2004', TO_TIMESTAMP('15/06/2022 19:14',
'DD/MM/YYYY HH24:MI'), '3284576232', '3', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2005', TO_TIMESTAMP('15/06/2022 18:20',
'DD/MM/YYYY HH24:MI'), '3592742972', '5', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2006', TO_TIMESTAMP('15/06/2022 19:30',
'DD/MM/YYYY HH24:MI'), '3412052322', '7', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2007', TO_TIMESTAMP('15/06/2022 14:14',
'DD/MM/YYYY HH24:MI'), '3357523172', '3', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2008', TO_TIMESTAMP('15/06/2022 19:45',
'DD/MM/YYYY HH24:MI'), '3327419822', '5', '0816317485');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT2009', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), '3745892012', '6', '0816317485');
--figli ordinazione (al tavolo) terzo pub
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3001', TO_TIMESTAMP('15/06/2022 13:14',
'DD/MM/YYYY HH24:MI'), '3758523399', '6', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3002', TO_TIMESTAMP('15/06/2022 14:30',
'DD/MM/YYYY HH24:MI'), '3985464699', '6', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3003', TO_TIMESTAMP('15/06/2022 21:20',
'DD/MM/YYYY HH24:MI'), '3425974699', '4', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3004', TO_TIMESTAMP('15/06/2022 20:14',
'DD/MM/YYYY HH24:MI'), '3284576299', '3', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3005', TO_TIMESTAMP('15/06/2022 19:20',
'DD/MM/YYYY HH24:MI'), '3592742999', '5', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3006', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), '3412052399', '7', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3007', TO_TIMESTAMP('15/06/2022 15:14',
'DD/MM/YYYY HH24:MI'), '3357523199', '3', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3008', TO_TIMESTAMP('15/06/2022 20:45',
'DD/MM/YYYY HH24:MI'), '3327419899', '5', '0814103527');
INSERT INTO O_AL_TAVOLO (COD_ORDINAZIONE_TAVOLO, DA_O_PRENOT_TAV, REC_TEL,
NM_TAVOLO, PHONE_PUB) VALUES ('OT3009', TO_TIMESTAMP('15/06/2022 21:30',
'DD/MM/YYYY HH24:MI'), '3745892099', '6', '0814103527');

```

INSERIMENTO ORDINAZIONE A DOMICILIO

--figli ordinazione (a domicilio) primo pub

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD1001',  
'0812109721');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD1002',  
'0814732014');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD1003',  
'0811465976');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD1004',  
'0813428854');
```

--figli ordinazione (a domicilio) secondo pub

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD2001',  
'0812109722');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD2002',  
'0814732012');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD2003',  
'0811465972');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD2004',  
'0813428852');
```

--figli ordinazione (a domicilio) terzo pub

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD3001',  
'0812109799');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD3002',  
'0814732099');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD3003',  
'0811465999');
```

```
INSERT INTO O_A_DOMICILIO (COD_ORDINAZIONE_DOMICILIO, RECET_TEL) VALUES ('OD3004',  
'0813428899');
```

INSERIMENTO CONTO

--questi sono i conti relativi agli ordini del primo pub

```
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,  
RECAPITO_PUB, SCONTO) VALUES ('SP1001', TO_TIMESTAMP('15/06/2022 13:14',  
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT1001', '0815745838', 'SI');
```

```
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,  
RECAPITO_PUB, SCONTO) VALUES ('SP1002', TO_TIMESTAMP('15/06/2022 14:30',  
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT1002', '0815745838', 'SI');
```

```
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,  
RECAPITO_PUB, SCONTO) VALUES ('SP1003', TO_TIMESTAMP('15/06/2022 21:20',  
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT1003', '0815745838', 'SI');
```

```
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,  
RECAPITO_PUB, SCONTO) VALUES ('SP1004', TO_TIMESTAMP('15/06/2022 20:14',  
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT1004', '0815745838', 'SI');
```

```

INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1005', TO_TIMESTAMP('15/06/2022 19:20',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT1005', '0815745838', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1006', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT1006', '0815745838', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1007', TO_TIMESTAMP('15/06/2022 15:14',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT1007', '0815745838', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1008', TO_TIMESTAMP('15/06/2022 20:45',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT1008', '0815745838', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1009', TO_TIMESTAMP('15/06/2022 21:30',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT1009', '0815745838', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1010', TO_TIMESTAMP('15/06/2022 17:45',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OD1001', '0815745838', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1011', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OD1002', '0815745838', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1012', TO_TIMESTAMP('15/06/2022 22:00',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OD1003', '0815745838', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP1013', TO_TIMESTAMP('15/06/2022 21:00',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OD1004', '0815745838', 'SI');
--questi sono i conti relativi agli ordini del secondo pub
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2001', TO_TIMESTAMP('15/06/2022 13:14',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT2001', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2002', TO_TIMESTAMP('15/06/2022 14:30',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT2002', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2003', TO_TIMESTAMP('15/06/2022 21:20',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT2003', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2004', TO_TIMESTAMP('15/06/2022 20:14',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT2004', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2005', TO_TIMESTAMP('15/06/2022 19:20',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT2005', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2006', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT2006', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2007', TO_TIMESTAMP('15/06/2022 15:14',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT2007', '0816317485', 'SI');

```



```

INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2008', TO_TIMESTAMP('15/06/2022 20:45',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT2008', '0816317485', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2009', TO_TIMESTAMP('15/06/2022 21:30',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT2009', '0816317485', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2010', TO_TIMESTAMP('15/06/2022 17:45',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OD2001', '0816317485', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2011', TO_TIMESTAMP('15/06/2022 20:30',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OD2002', '0816317485', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2012', TO_TIMESTAMP('15/06/2022 22:00',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OD2003', '0816317485', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP2013', TO_TIMESTAMP('15/06/2022 21:00',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OD2004', '0816317485', 'SI');
--questi sono i conti relativi agli ordini del terzo pub
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3001', TO_TIMESTAMP('15/06/2022 14:14',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT3001', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3002', TO_TIMESTAMP('15/06/2022 15:30',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT3002', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3003', TO_TIMESTAMP('15/06/2022 22:20',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT3003', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3004', TO_TIMESTAMP('15/06/2022 21:14',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT3004', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3005', TO_TIMESTAMP('15/06/2022 20:20',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT3005', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3006', TO_TIMESTAMP('15/06/2022 21:30',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT3006', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3007', TO_TIMESTAMP('15/06/2022 16:14',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT3007', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3008', TO_TIMESTAMP('15/06/2022 21:45',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OT3008', '0814103527', 'SI');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3009', TO_TIMESTAMP('15/06/2022 22:30',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OT3009', '0814103527', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3010', TO_TIMESTAMP('15/06/2022 18:45',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OD3001', '0814103527', 'NO');

```

```

INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3011', TO_TIMESTAMP('15/06/2022 21:30',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OD3002', '0814103527', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3012', TO_TIMESTAMP('15/06/2022 23:00',
'DD/MM/YYYY HH24:MI'), 'CARTA', 'OD3003', '0814103527', 'NO');
INSERT INTO CONTO (ID_SCONTRINO, ORA_DATA_SCONTRINO, METODO_PAGAMENTO, ID_ORDINE,
RECAPITO_PUB, SCONTO) VALUES ('SP3013', TO_TIMESTAMP('15/06/2022 22:00',
'DD/MM/YYYY HH24:MI'), 'CONTANTI', 'OD3004', '0814103527', 'SI');

```

INSERIMENTO CONSEGNA

--CONSEGNE PRIMO PUB

```

INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 18:30', 'DD/MM/YYYY HH24:MI'), 'VIALE LOS SANTOS',
'ERCOLANO', 'CF1RID0000000007', 'OD1001');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), 'VIA PUGLIANO',
'ERCOLANO', 'CF1RID0000000007', 'OD1002');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 22:14', 'DD/MM/YYYY HH24:MI'), 'CORSO RESINA',
'ERCOLANO', 'CF1RID0000000008', 'OD1003');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 21:00', 'DD/MM/YYYY HH24:MI'), 'VIA MIGLIO DORO',
'ERCOLANO', 'CF1RID0000000008', 'OD1004');

```

--CONSEGNE SECONDO PUB

```

INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 18:30', 'DD/MM/YYYY HH24:MI'), 'VIALE GRABBINI 23',
'SPOLETO', 'CF2RID0000000007', 'OD2001');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 20:30', 'DD/MM/YYYY HH24:MI'), 'VIA GIULIANO 32',
'SPOLETO', 'CF2RID0000000007', 'OD2002');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 22:14', 'DD/MM/YYYY HH24:MI'), 'CORSO GARIBALDONE 65',
'SPOLETO', 'CF2RID0000000008', 'OD2003');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 21:00', 'DD/MM/YYYY HH24:MI'), 'VIA MIGLIO ARGENTATO 24',
'SPOLETO', 'CF2RID0000000008', 'OD2004');

```

--CONSEGNE TERZO PUB

```

INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 19:30', 'DD/MM/YYYY HH24:MI'), 'VIALE VITINO 23',
'ALGHERO', 'CF3RID0000000007', 'OD3001');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 21:30', 'DD/MM/YYYY HH24:MI'), 'VIA CESARE 42',
'ALGHERO', 'CF3RID0000000007', 'OD3002');
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 23:14', 'DD/MM/YYYY HH24:MI'), 'CORSO ERCOLE 11',
'ALGHERO', 'CF3RID0000000008', 'OD3003');

```

```
INSERT INTO CONSEGNA (DATA_ORA_CONSEGNA, VIA, CITTA, CF_RID, COD_ORD) VALUES
(TO_TIMESTAMP('15/06/2022 22:00', 'DD/MM/YYYY HH24:MI'), 'VIA MIGLIO BRONZEO 31',
'ALGHERO', 'CF3RID0000000008', 'OD3004');
```

INSERIMENTO COMPRENDE (TRA ORDINAZIONE E MENU)

--SPECIFICO DA COSA E' COMPOSTO UN ORDINE (PRIMO PUB)

```
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1001', 'THOR', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1001', 'HULK', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1001', 'FANTA', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1002', 'SCARLET
WITCH', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1002', 'SPIDER MAN',
1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1002', 'COCA COLA',
3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1003', 'CAPITAN
AMERICA', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1003', 'SPIDER MAN',
1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1003', 'SCARLET
WITCH', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1003', 'COCA COLA',
2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1003', 'FANTA', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1004', 'THOR', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1004', 'ACQUA
NATURALE', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1004', 'SPRITE', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1005', 'HULK', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1005', 'ACQUA
NATURALE', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1006', 'IRON MAN',
5);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1006', 'FANTA', 5);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1007', 'SPIDER MAN',
2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1007', 'FANTA', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1008', 'IRON MAN',
3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1008', 'ACQUA
FRIZZANTE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1009', 'SCARLET
WITCH', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT1009', 'SPRITE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1001', 'SCARLET
WITCH', 7);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1001', 'FANTA', 6);
```



```

INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1002', 'CAPITAN
AMERICA', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1002', 'DOCTOR
STRANGE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1002', 'COCA COLA',
6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1003', 'SPIDER MAN',
4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1003', 'PERONI', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1004', 'SCARLET
WITCH', 6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD1004', 'COCA COLA',
6);
--SPECIFICO DA COSA E' COMPOSTO UN ORDINE (SECONDO PUB)
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2001', 'IRON MAN',
2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2001', 'HULK', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2001', 'ACQUA
NATURALE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2002', 'SCARLET
WITCH', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2002', 'DOCTOR
STRANGE', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2002', 'COCA COLA',
3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2003', 'CAPITAN
AMERICA', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2003', 'SPIDER MAN',
1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2003', 'DOCTOR
STRANGE', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2003', 'COCA COLA',
1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2003', 'PERONI', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2004', 'THOR', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2004', 'ACQUA
NATURALE', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2004', 'SPRITE', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2005', 'HULK', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2005', 'ACQUA
NATURALE', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2006', 'IRON MAN',
5);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2006', 'FANTA', 5);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2007', 'SPIDER MAN',
2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2007', 'COCA COLA',
3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2008', 'IRON MAN',
3);

```

```

INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2008', 'ACQUA
FRIZZANTE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2009', 'HULK', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT2009', 'SPRITE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2001', 'SCARLET
WITCH', 7);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2001', 'FANTA', 6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2002', 'CAPITAN
AMERICA', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2002', 'DOCTOR
STRANGE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2002', 'COCA COLA',
6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2003', 'HULK', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2003', 'PERONI', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2004', 'THOR', 6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD2004', 'COCA COLA',
6);
--SPECIFICO DA COSA E' COMPOSTO UN ORDINE (TERZO PUB)
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3001', 'THOR', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3001', 'HULK', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3001', 'FANTA', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3002', 'CAPITAN
AMERICA', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3002', 'HULK', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3002', 'PERONI', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3003', 'CAPITAN
AMERICA', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3003', 'DOCTOR
STRANGE', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3003', 'IRON MAN',
1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3003', 'COCA COLA',
2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3003', 'SPRITE', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3004', 'HULK', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3004', 'ACQUA
NATURALE', 1);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3004', 'FANTA', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3005', 'HULK', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3005', 'ACQUA
NATURALE', 2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3006', 'IRON MAN',
5);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3006', 'FANTA', 5);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3007', 'SPIDER MAN',
2);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3007', 'FANTA', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3008', 'IRON MAN',
3);

```

```

INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3008', 'ACQUA
FRIZZANTE', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3009', 'HULK', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OT3009', 'COCA COLA',
3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3001', 'SCARLET
WITCH', 7);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3001', 'COCA COLA',
6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3002', 'CAPITAN
AMERICA', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3002', 'THOR', 3);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3002', 'COCA COLA',
6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3003', 'THOR', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3003', 'PERONI', 4);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3004', 'IRON MAN',
6);
INSERT INTO COMPRENDE (NUM_ORD, NOM_MENU, QUANTITA) VALUES ('OD3004', 'PERONI', 6);

```

INSERIMENTO FORNITURA

```

--QUESTE SONO LE FORNITURE DELLE MATERIE PRIME E DELLE BIBITE (PRIMO PUB)
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1001', '0815745838', 'PANINO', NULL,
'NAPOLIHORECA', 1.2, 220, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1002', '0815745838', 'HAMBURGER', NULL,
'NAPOLIHORECA', 1.2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1003', '0815745838', 'LATTUGA', NULL,
'NAPOLIHORECA', 1.2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1004', '0815745838', 'COTOLETTA', NULL,
'NAPOLIHORECA', 1, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1005', '0815745838', 'PORCHETTA', NULL,
'NAPOLIHORECA', 1.2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));

```

```

INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1006', '0815745838', 'SALSICCIA', NULL,
'NAPOLIHORECA', 1.2, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1007', '0815745838', 'BECON', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1008', '0815745838', 'UOVO', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1009', '0815745838', 'SOTTILETTA', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1010', '0815745838', 'PROVOLA', NULL,
'NAPOLIHORECA', 1.1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1011', '0815745838', 'POMODORO', NULL,
'NAPOLIHORECA', 1, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1013', '0815745838', 'PIADINA', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1015', '0815745838', 'PATATINE', NULL,
'NAPOLIHORECA', 1.2, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1016', '0815745838', 'CETRIOLO', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F1017', '0815745838', 'HAMBURGER -
VEGETARIANO', NULL, 'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));

```

```

INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1018', '0815745838', 'FRIARIELLI', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1019', '0815745838', 'PEPERONI', NULL,
'NAPOLIHORECA', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1020', '0815745838', 'FUNGHI CHAMPIGNON',
NULL, 'NAPOLIHORECA', 1, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1021', '0815745838', 'CHEDDAR', NULL,
'NAPOLIHORECA', 1, 130, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1023', '0815745838', NULL, 'COCA COLA',
'CIFA SRL', 2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1024', '0815745838', NULL, 'PERONI', 'CIFA
SRL', 2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1025', '0815745838', NULL, 'FANTA', 'CIFA
SRL', 2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1026', '0815745838', NULL, 'SPRITE', 'CIFA
SRL', 2, 300, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1027', '0815745838', NULL, 'ACQUA
FRIZZANTE', 'CIFA SRL', 0.6, 300, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F1028', '0815745838', NULL, 'ACQUA NATURALE',
'CIFA SRL', 0.6, 300, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));

```



```

--QUESTE SONO LE FORNITURE DELLE MATERIE PRIME E DELLE BIBITE (SECONDO PUB)
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2001', '0816317485', 'PANINO', NULL,
'FRATELLI BUCCERO', 1.2, 150, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'),TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2002', '0816317485', 'HAMBURGER', NULL,
'FRATELLI BUCCERO', 1.9, 130, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2003', '0816317485', 'LATTUGA', NULL,
'FRATELLI BUCCERO', 0.5, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2004', '0816317485', 'COTOLETTA', NULL,
'FRATELLI BUCCERO', 1, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2005', '0816317485', 'PORCHETTA', NULL,
'FRATELLI BUCCERO', 1, 90, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2006', '0816317485', 'SALSICCIA', NULL,
'FRATELLI BUCCERO', 1, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2007', '0816317485', 'BECON', NULL,
'FRATELLI BUCCERO', 1, 111, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2008', '0816317485', 'UOVO', NULL, 'FRATELLI
BUCCERO', 1, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2009', '0816317485', 'SOTTILETTA', NULL,
'FRATELLI BUCCERO', 1, 130, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2010', '0816317485', 'PROVOLA', NULL,

```

```

'FRATELLI BUCCERO', 0.9, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2011', '0816317485', 'POMODORO', NULL,
'FRATELLI BUCCERO', 1, 105, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2013', '0816317485', 'PIADINA', NULL,
'FRATELLI BUCCERO', 2, 105, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2015', '0816317485', 'PATATINE', NULL,
'FRATELLI BUCCERO', 1, 104, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2016', '0816317485', 'CETRIOLO', NULL,
'FRATELLI BUCCERO', 3, 102, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2017', '0816317485', 'HAMBURGER -
VEGETARIANO', NULL, 'FRATELLI BUCCERO', 1, 110, TO_DATE('01/06/2022',
'DD/MM/YYYY'), TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022',
'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2018', '0816317485', 'FRIARIELLI', NULL,
'FRATELLI BUCCERO', 1, 103, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2019', '0816317485', 'PEPERONI', NULL,
'FRATELLI BUCCERO', 1, 140, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2020', '0816317485', 'FUNGHI CHAMPIGNON',
NULL, 'FRATELLI BUCCERO', 0.2, 90, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F2021', '0816317485', 'CHEDDAR', NULL,
'FRATELLI BUCCERO', 1, 90, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,

```

```

DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2023', '0816317485', NULL, 'COCA COLA',
'PARATESI SRL', 1, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2024', '0816317485', NULL, 'PERONI',
'PARATESI SRL', 1, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2025', '0816317485', NULL, 'FANTA',
'PARATESI SRL', 1, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2026', '0816317485', NULL, 'SPRITE',
'PARATESI SRL', 1, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2027', '0816317485', NULL, 'ACQUA
FRIZZANTE', 'PARATESI SRL', 0.2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F2028', '0816317485', NULL, 'ACQUA NATURALE',
'PARATESI SRL', 0.2, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
--QUESTE SONO LE FORNITURE DELLE MATERIE PRIME E DELLE BIBITE (TERZO PUB)
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3001', '0814103527', 'PANINO', NULL,
'PARTESA SRL', 1.5, 103, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3002', '0814103527', 'HAMBURGER', NULL,
'PARTESA SRL', 2, 130, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3003', '0814103527', 'LATTUGA', NULL,
'PARTESA SRL', 0.4, 140, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3004', '0814103527', 'COTOLETTA', NULL,
'PARTESA SRL', 1.5, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));

```



```

INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3005', '0814103527', 'PORCHETTA', NULL,
'PARTESA SRL', 1.3, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3006', '0814103527', 'SALSICCIA', NULL,
'PARTESA SRL', 1.8, 130, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3007', '0814103527', 'BECON', NULL, 'PARTESA
SRL', 0.9, 140, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3008', '0814103527', 'UOVO', NULL, 'PARTESA
SRL', 1, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3009', '0814103527', 'SOTTILETTA', NULL,
'PARTESA SRL', 0.1, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3010', '0814103527', 'PROVOLA', NULL,
'PARTESA SRL', 0.2, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3011', '0814103527', 'POMODORO', NULL,
'PARTESA SRL', 1, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3013', '0814103527', 'PIADINA', NULL,
'PARTESA SRL', 1.4, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3015', '0814103527', 'PATATINE', NULL,
'PARTESA SRL', 1.2, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA,DATA_SCADENZA) VALUES ('F3016', '0814103527', 'CETRIOLO', NULL,
'PARTESA SRL', 0.3, 110, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));

```

```

INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3017', '0814103527', 'HAMBURGER -
VEGETARIANO', NULL, 'PARTESA SRL', 1, 100, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3018', '0814103527', 'FRIARIELLI', NULL,
'PARTESA SRL', 1, 140, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3019', '0814103527', 'PEPERONI', NULL,
'PARTESA SRL', 1, 120, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3020', '0814103527', 'FUNGHI CHAMPIGNON',
NULL, 'PARTESA SRL', 1, 90, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3021', '0814103527', 'CHEDDAR', NULL,
'PARTESA SRL', 0.1, 80, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2022', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3023', '0814103527', NULL, 'COCA COLA',
'GIOIASPA', 1, 200, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3024', '0814103527', NULL, 'PERONI',
'GIOIASPA', 1, 300, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3025', '0814103527', NULL, 'FANTA',
'GIOIASPA', 1, 300, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3026', '0814103527', NULL, 'SPRITE',
'GIOIASPA', 1, 300, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));
INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3027', '0814103527', NULL, 'ACQUA
FRIZZANTE', 'GIOIASPA', 0.3, 400, TO_DATE('01/06/2022', 'DD/MM/YYYY'),
TO_DATE('05/06/2022', 'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));

```

```

INSERT INTO FORNITURA (COD_FORNITURA, TELEF_PUB, COD_MAT_PR, N_BIBITA,
NOME_FORNITORE, PREZZO, QUANTITA_ACQUISTO, DATA_RICHIESTA,
DATA_CONSEGNA, DATA_SCADENZA) VALUES ('F3028', '0814103527', NULL, 'ACQUA NATURALE',
'GIOIASPA', 0.3, 430, TO_DATE('01/06/2022', 'DD/MM/YYYY'), TO_DATE('05/06/2022',
'DD/MM/YYYY'), TO_DATE('10/06/2023', 'DD/MM/YYYY'));

```

5.7 Scheduler

Riporto, inoltre, uno scheduler. Esso permette di cancellare, ogni settimana, le prenotazioni effettuate più di un anno fa:

```

BEGIN
DBMS_SCHEDULER.CREATE_JOB(
    JOB_NAME => 'CANCELLA_PRENOTAZIONI',
    JOB_TYPE => 'PLSQL_BLOCK',
    JOB_ACTION => 'BEGIN
                        DELETE FROM PRENOTAZIONE
                        WHERE DATA_ORA_PRENOTAZIONE<SYSDATE-365;
                    END;',
    START_DATE => SYSDATE,
    REPEAT_INTERVAL => 'FREQ = WEEKLY',
    ENABLED => TRUE,
    COMMENTS => 'Cancellazione delle prenotazioni effettuate da più di un anno');
END;

/* CANCELLAZIONE DEL JOB

BEGIN
DBMS_SCHEDULER.DROP_JOB('CANCELLA_PRENOTAZIONI');
END;

*/

```