



UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

DIPARTIMENTO DI SCIENZE E TECNOLOGIE
CORSO DI BASI DI DATI E LAB. DI BASI DI DATI

Fabbrica di microcontrollori



| Studenti | Matricola |
|--------------------|----------------------|
| Galiero Nicola | 0124002671 |
| Capocotta Giuseppe | 0124002676 |
| Data di consegna | 18/06/2023 |
| Anno accademico | 2022 - 2023 |
| Categoria | Attività commerciale |

Indice

| | | |
|----------|---|----------|
| 1 | Progettazione | 3 |
| 1.1 | Analisi dei requisiti | 3 |
| 1.2 | Glossario | 3 |
| 1.3 | Diagramma EE/R | 4 |
| 1.4 | Elenco delle operazioni di base | 4 |
| 1.5 | Elenco delle operazioni degli utenti | 5 |
| 1.6 | Use case e descrittore di operazione per le operazioni degli utenti | 5 |
| 1.7 | Elenco dei vincoli d'integrità statici | 7 |
| 1.8 | Elenco dei vincoli di integrità dinamici | 8 |
| 1.9 | Diagramma relazionale | 8 |
| 1.10 | Verifica di normalità dello schema | 8 |
| 2 | Implementazione | 9 |
| 2.1 | Creazione utenti | 9 |
| 2.2 | Data Definition Language | 9 |
| 2.2.1 | Cliente | 9 |
| 2.2.2 | Fattura | 9 |
| 2.2.3 | Spedizione | 9 |
| 2.2.4 | Ordine | 10 |
| 2.2.5 | Componente | 10 |
| 2.2.6 | Progetto | 10 |
| 2.2.7 | ComponentiProgetto | 10 |
| 2.2.8 | Contiene | 10 |
| 2.2.9 | Fornitore | 11 |
| 2.2.10 | Fornitura | 11 |
| 2.2.11 | Reparto | 11 |
| 2.2.12 | Progettazione | 11 |
| 2.2.13 | Produzione | 12 |
| 2.2.14 | Collaudo | 12 |
| 2.2.15 | Magazzino | 12 |
| 2.2.16 | Immagazzinamento | 12 |
| 2.2.17 | Dipendente | 12 |
| 2.2.18 | Turno | 13 |
| 2.3 | Data Manipulation Language | 14 |
| 2.3.1 | Popolamento tabella Cliente | 14 |
| 2.3.2 | Popolamento tabella fattura | 14 |
| 2.3.3 | Popolamento tabella spedizione | 15 |
| 2.3.4 | Popolamento tabella ordine | 16 |
| 2.3.5 | Popolamento tabella componente | 17 |
| 2.3.6 | Popolamento tabella progetto | 17 |
| 2.3.7 | Popolamento tabella componentiProgetto | 18 |
| 2.3.8 | Popolamento tabella contiene | 19 |
| 2.3.9 | Popolamento tabella fornitore | 19 |
| 2.3.10 | Popolamento tabella fornitura | 20 |
| 2.3.11 | Popolamento tabella reparto | 21 |
| 2.3.12 | Popolamento tabella progettazione | 22 |
| 2.3.13 | Popolamento tabella collaudo | 22 |
| 2.3.14 | Popolamento tabella magazzino | 23 |
| 2.3.15 | Popolamento tabella immagazzinamento | 23 |
| 2.3.16 | Popolamento tabella dipendente | 24 |
| 2.3.17 | Popolamento tabella turno | 26 |
| 2.4 | Procedure e funzioni | 29 |
| 2.4.1 | Cliente ordina un progetto | 29 |
| 2.4.2 | Cliente cancella un progetto dall'ordine | 30 |
| 2.4.3 | Fornitore inserisce una nuova fornitura | 31 |
| 2.4.4 | Stima di consegna di un progetto | 32 |
| 2.5 | Viste | 33 |

| | | |
|-------|--|----|
| 2.5.1 | Vista ordine cliente | 33 |
| 2.5.2 | Vista fattura dettagliata | 33 |
| 2.5.3 | Vista componenti progetto | 33 |
| 2.5.4 | Vista dipendente reparto | 33 |
| 2.5.5 | VISTA CALCOLO PRESENZE | 33 |
| 2.5.6 | VISTA PER CALCOLARE LO SPAZIO IN MAGAZZINO | 34 |
| 2.6 | Data Control Language | 35 |
| 2.7 | Scheduler | 35 |

Premessa

Noi sottoscritti Galiero Nicola e Capocotta Giuseppe abbiamo scelto un tema a noi stretto in quanto avendo frequentato insieme le scuole superiori, portando all'esame di maturità dei progetti legati al settore dei microcontrollori.

1 Progettazione

1.1 Analisi dei requisiti

L'azienda produttrice di componenti elettronici necessita di un sistema di gestione dei dati efficiente e affidabile per supportare le sue attività quotidiane. In particolare, l'azienda deve essere in grado di gestire i dati relativi alla produzione dei suoi prodotti, ai fornitori, ai clienti e ai dipendenti.

Per quanto riguarda la produzione, l'azienda deve essere in grado di monitorare lo stato dei suoi processi produttivi, il livello di inventario dei componenti (magazzino), la disponibilità della forza lavoro (turni) e la qualità dei prodotti finiti (collaudo). Inoltre, l'azienda deve essere in grado di programmare la produzione dei suoi prodotti in modo efficiente, considerando i tempi di consegna richiesti dai clienti e le scorte di magazzino disponibili.

Per quanto riguarda i fornitori, l'azienda deve essere in grado di gestire i dati relativi ai fornitori con cui collabora, inclusi i loro dati di contatto e i loro prezzi. Inoltre, l'azienda deve essere in grado di tenere traccia degli ordini effettuati ai fornitori e dei tempi di consegna previsti.

Per quanto riguarda i clienti, l'azienda deve essere in grado di gestire i dati relativi ai clienti, inclusi i loro dati di contatto e i loro ordini. Inoltre, i corrieri spediscono gli ordini richiesti dai clienti.

1.2 Glossario

Il glossario ha lo scopo fondamentale di chiarire il gergo tecnico usato nella descrizione dei requisiti e di evidenziare eventuali sinonimi.

| Termine | Definizione | Sinonimo |
|---------------------------------|---|------------|
| Gestione dei dati | Organizzazione e manipolazione dei dati | Rendimento |
| Processi produttivi | Attività e fasi di lavorazione | Lavoro |
| Inventari di componenti | Quantità di componenti elettronici | Catalogo |
| Forza lavoro | Personale impiegato nell'azienda | — |
| Programmazione della produzione | Pianificazione delle attività produttive | — |
| Fornitori | Imprese che forniscono componenti | — |
| Scorte di magazzino | Rappresentano i componenti elettronici | — |
| Stima tempo | Stima dei giorni che un progetto è in progettazione | — |

1.3 Diagramma EE/R

Non sono indicate le cardinalità massime e minime delle associazioni. Sono indicate solo le molteplicità.

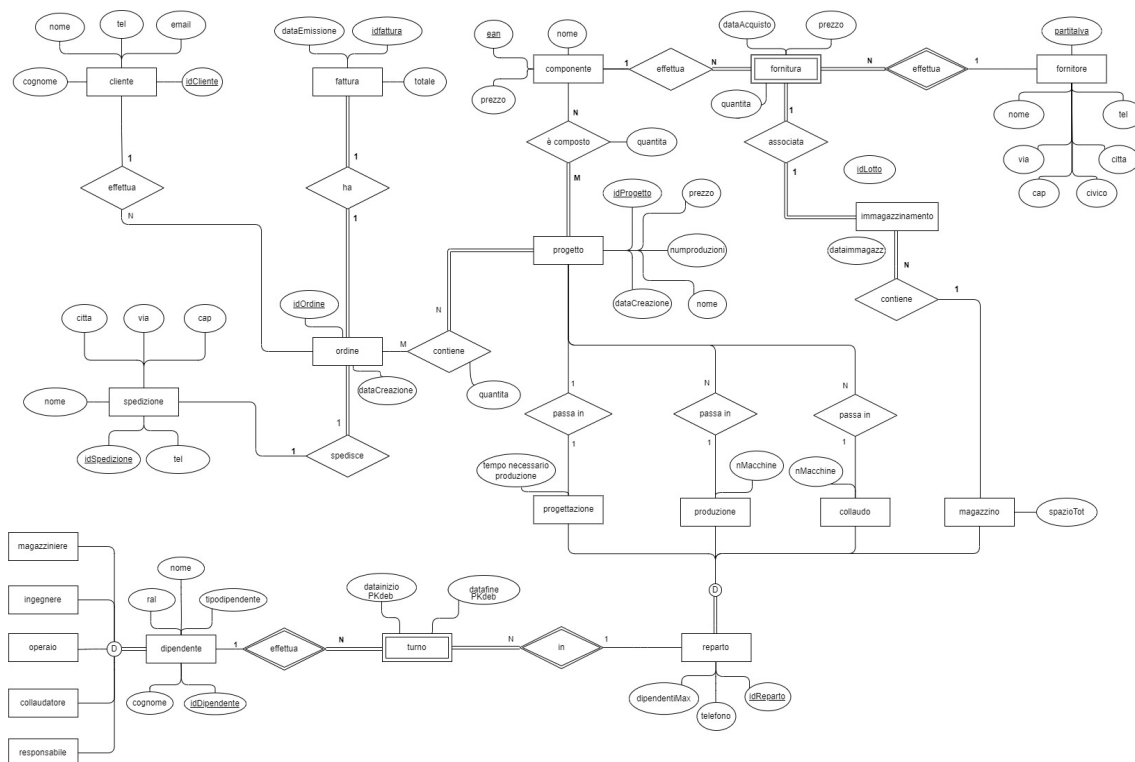


Figura 1: Diagramma relazionale

1.4 Elenco delle operazioni di base

Le operazioni di base in un database si riferiscono alle azioni fondamentali che possono essere eseguite per manipolare i dati all'interno di un database, come l'inserimento di un nuovo dipendente. Questi dati vengono gestiti dall'amministratore che detiene tutti i permessi. Queste operazioni sono visibili nella parte di popolamento delle tabelle(vedi paragrafo 2.3 Data Manipulation Language).

1.5 Elenco delle operazioni degli utenti

Gli utenti (vedi paragrafo 2.6) effettuano le seguenti operazioni con i relativi permessi:

| 1 | UTENTE | TIPO | PERMESSI |
|----|--------------|----------------|------------------------------------|
| 2 | | | |
| 3 | db_fabbrica | amministratore | ALL |
| 4 | | | |
| 5 | responsabile | comune | UPDATE ON progettazione |
| 6 | | | produzione |
| 7 | | | EXECUTE responsabile_aumenta_stato |
| 8 | | | |
| 9 | cliente | comune | SELECT ON cliente |
| 10 | | | progetto |
| 11 | | | INSERT ON cliente |
| 12 | | | ordine |
| 13 | | | UPDATE ON ordine |
| 14 | | | |
| 15 | | | EXECUTE ordina_progetto |
| 16 | | | EXECUTE cancella_progetto_ordine |
| 17 | | | |
| 18 | fornitore | comune | SELECT ON fornitore |
| 19 | | | SELECT ON componente |
| 20 | | | INSERT on forniture |
| 21 | | | |
| 22 | | | EXECUTE inserisci_fornitura |

1.6 Use case e descrittore di operazione per le operazioni degli utenti

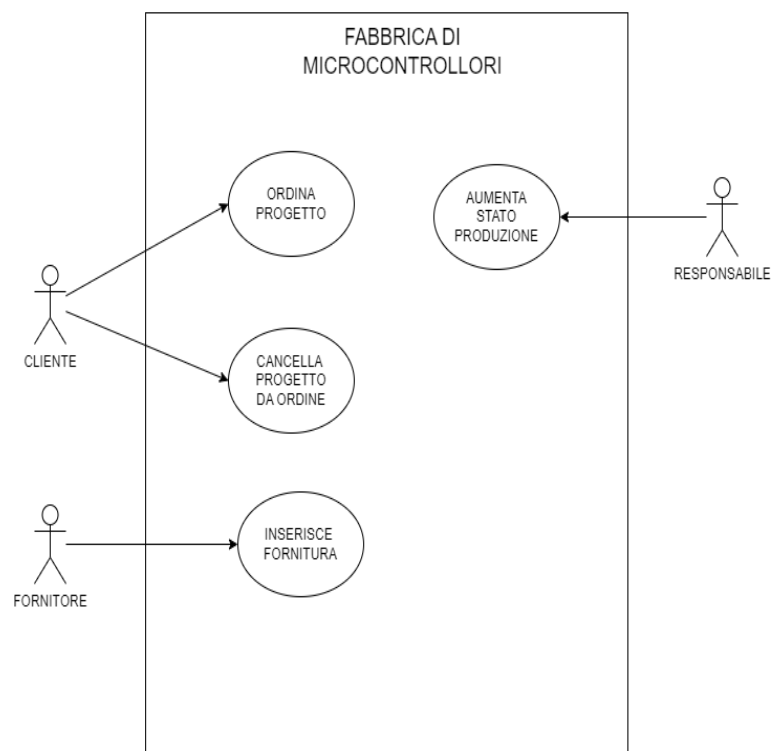


Diagramma Use Case

Seguono le schede descrittore operazione per le operazioni degli utenti

| OPERAZIONI | ORDINA PROGETTO |
|-------------------|--|
| SCOPO | Inserire un nuovo ordine da parte di un cliente |
| ARGOMENTI | Nome cliente Cognome cliente Email cliente Telefono cliente Via cliente Città cliente Cap cliente Nome progetto |
| RISULTATO | Ordine aggiunto alla tabella ordini |
| ERRORI | Il progetto non esiste |
| USA | Cliente Progetto Ordine |
| MODIFICA | Cliente Ordine |
| PRIMA | Se il cliente non esiste viene creato |
| POI | Cliente aggiunto al database |
| OPERAZIONI | CANCELLA PROGETTO DALL'ORDINE |
| SCOPO | Cancellare un progetto da un ordine se quest'ultimo ne contiene più progetti |
| ARGOMENTI | Nome cliente ID ordine ID progetto |
| RISULTATO | Ordine cancellato |
| ERRORI | L'ordine non esiste |
| USA | Progetto Ordine |
| MODIFICA | Ordine |
| PRIMA | Ordine con due o più progetti |
| POI | Progetto dall'ordine cancellato |

| OPERAZIONI | INSERIRE FORNITURA |
|-------------------|---|
| SCOPO | Inserire una nuova fornitura |
| ARGOMENTI | Partita iva del fornitore EAN componente Quantità Data dell'acquisto |
| RISULTATO | Fornitura aggiunta |
| ERRORI | Il fornitore non esiste Il componente non esiste |
| USA | Componente Fornitore Fornitura |
| MODIFICA | Fornitura |
| PRIMA | - - - |
| POI | Fornitura inserita |
| OPERAZIONI | AUMENTO DI STATO NEI REPARTI |
| SCOPO | Responsabile aumenta lo Stato prototipo e lo Stato di produzione |
| ARGOMENTI | ID Progetto |
| RISULTATO | Stato produzione e progettazione variati |
| ERRORI | Gli stati non possono essere cambiati |
| USA | Progettazione Produzione |
| MODIFICA | Progettazione Produzione |
| PRIMA | I precedenti stati della produzione e progettazione erano invariati |
| POI | Variazione dello stato di produzione |

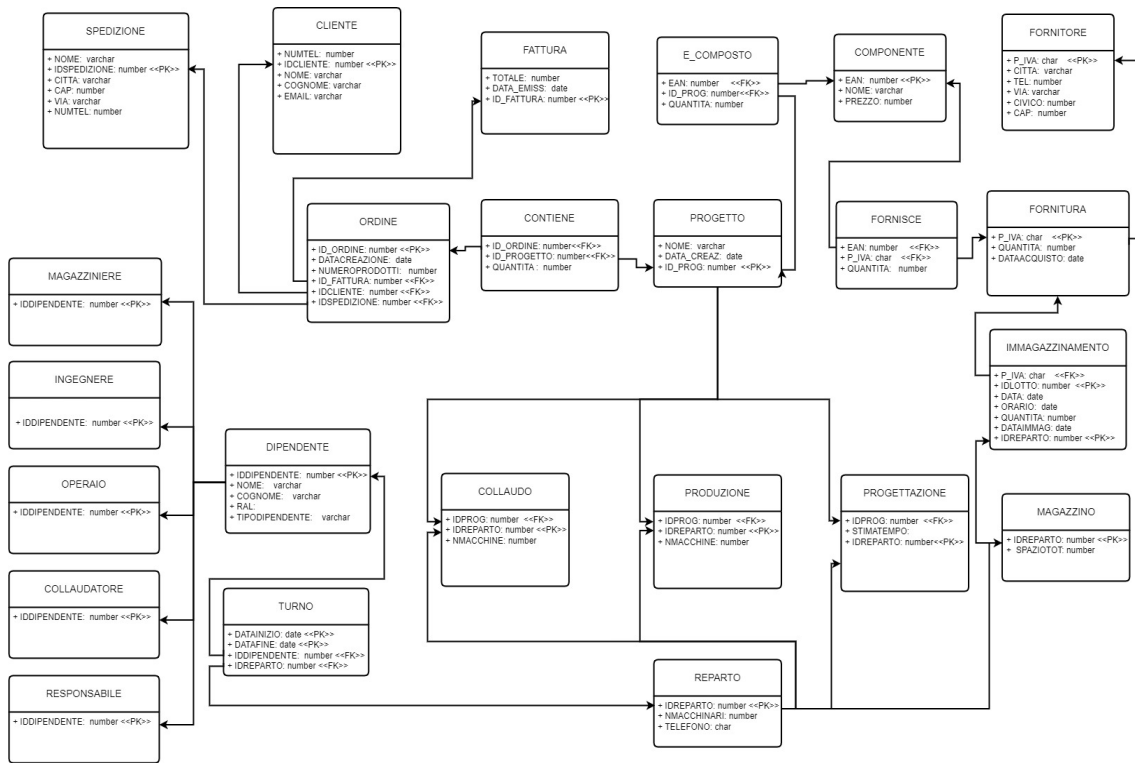
1.7 Elenco dei vincoli d'integrità statici

- Non ci devono mai essere meno di X scorte in magazzino per ogni componente, altrimenti non si può fare nessun ordine.
- In un reparto ci devono essere almeno 8 dipendenti per turno.
- Ogni microcontrollore deve avere un singolo identificare
- I dati di contatto dei fornitori e dei clienti deve essere obbligatorio
- La durata di un progetto nei reparti dura tra 5 giorni e 10 giorni

1.8 Elenco dei vincoli di integrità dinamici

- Un ordine da parte di un cliente può essere cancellato al più dopo 5 ore.
- Un progetto deve essere al più 15 giorni nei reparti di progettazione e produzione.
- I dipendenti specializzati non possono lavorare più di 8 ore al giorno.
- Un fornitore non deve impiegare più di 2 giorni nel consegnare la merce.
- Il prezzo di un progetto è determinato dal prezzo dei singoli componenti elettronici coinvolti più una percentuale del 200
- In magazzino si verifica lo spazio residuo tramite la differenza tra i componenti già immagazzinati e lo spazio totale.

1.9 Diagramma relazionale



1.10 Verifica di normalità dello schema

Il diagramma relazionale soddisfa la prima forma normale (1FN) in quanto non ci sono né attributi multivalore né attributi strutturati.

Il diagramma relazionale soddisfa la seconda forma normale (2FN).

Tutte le tabelle rispettano la terza forma normale (3NF), poiché non presentano dipendenze transitive o attributi non chiave che dipendono da altri attributi non chiave.

2 Implementazione

2.1 Creazione utenti

Nel nostro database ci sono quattro utenti, di cui uno solo ha tutti i permessi generali. In tabella l'elenco e i relativi permessi degli utenti e le operazioni che svolgono.

```
1 --Creazione utenti
2 CREATE USER proprietario_db_fabbrica IDENTIFIED BY admin;
3 CREATE USER responsabile IDENTIFIED BY responsabile;
4 CREATE USER cliente IDENTIFIED BY cliente;
5 CREATE USER fornitore IDENTIFIED BY fornitore;
6
7 --Do tutti i privilegi al proprietario del database
8 GRANT ALL PRIVILEGES TO proprietario_db_fabbrica;
```

2.2 Data Definition Language

Il DDL riflette lo schema relazionale: le tabelle sono create mediante altrettante istruzioni di CREATE TABLE che includono tutti i vincoli di integrità esprimibili nel modello relazionale.

2.2.1 Cliente

```
1 CREATE TABLE CLIENTE (
2     IDCLIENTE NUMBER,
3     NOME VARCHAR2(30) NOT NULL,
4     COGNOME VARCHAR2(30) NOT NULL,
5     EMAIL VARCHAR2(30) NOT NULL,
6     TELEFONO NUMBER NOT NULL,
7     VIA VARCHAR2(30) NOT NULL,
8     CITTA VARCHAR2(30) NOT NULL,
9     CAP NUMBER NOT NULL
10    PRIMARY KEY (IDCLIENTE),
11    CONSTRAINT UK_EMAIL UNIQUE (EMAIL),
12    CONSTRAINT UK_TELEFONO UNIQUE (TELEFONO)
13 );
```

2.2.2 Fattura

```
1 CREATE TABLE FATTURA (
2     IDFATTURA NUMBER,
3     DATAEMISSIONE DATE,
4     TOTALE NUMBER NOT NULL,
5     PRIMARY KEY (IDFATTURA)
6 );
```

2.2.3 Spedizione

```
1 CREATE TABLE SPEDIZIONE (
2     IDSPEDIZIONE NUMBER PRIMARY KEY,
3     TEL NUMBER NOT NULL,
4     NOME VARCHAR2(30) NOT NULL,
5     VIA VARCHAR2(30) NOT NULL,
6     CITTA VARCHAR2(30) NOT NULL,
7     CAP NUMBER NOT NULL
8 );
```

2.2.4 Ordine

```
1      CREATE TABLE ORDINE (
2  CREATE TABLE ORDINE (
3      IDORDINE NUMBER NOT NULL,
4      IDFATTURA NUMBER NOT NULL,
5      IDCLIENTE NUMBER NOT NULL,
6      IDSPEDIZIONE NUMBER NOT NULL,
7      NUMEROPRODOTTI NUMBER NOT NULL,
8      DATACREAZIONE DATE NOT NULL,
9      PRIMARY KEY (IDORDINE),
10     CONSTRAINT FK_ORDINE_FATTURA FOREIGN KEY (IDFATTURA) REFERENCES
        FATTURA(IDFATTURA),
11     CONSTRAINT FK_ORDINE_CLIENTE FOREIGN KEY (IDCLIENTE) REFERENCES
        CLIENTE(IDCLIENTE),
12     CONSTRAINT FK_ORDINE_SPEDIZIONE FOREIGN KEY (IDSPEDIZIONE)
        REFERENCES SPEDIZIONE(IDSPEDIZIONE)
13 );
```

2.2.5 Componente

```
1  CREATE TABLE COMPONENTE (
2      EAN NUMBER,
3      NOME VARCHAR2(30) NOT NULL,
4      PREZZO NUMBER NOT NULL,
5      PRIMARY KEY (EAN)
6  );
```

2.2.6 Progetto

```
1  CREATE TABLE PROGETTO (
2      IDPROG NUMBER,
3      NOME VARCHAR2(30) NOT NULL,
4      DATAINIZIO DATE,
5      PRIMARY KEY (IDPROG),
6      UNIQUE (NOME)
7  );
```

2.2.7 ComponentiProgetto

```
1  CREATE TABLE COMPONENTIPROGETTO ( /*      l'associazione "
        composto" */
2      IDPROG NUMBER,
3      EAN NUMBER,
4      QUANTITA NUMBER(2,0),
5      FOREIGN KEY (IDPROG) REFERENCES PROGETTO (IDPROG),
6      FOREIGN KEY (EAN) REFERENCES COMPONENTE (EAN)
7  );
```

2.2.8 Contiene

```
1  CREATE TABLE CONTIENE (
2      IDPROG NUMBER,
3      IDORDINE NUMBER,
```

```

4     QUANTITA NUMBER,
5     FOREIGN KEY (IDPROG) REFERENCES PROGETTO(IDPROG),
6     FOREIGN KEY (IDORDINE) REFERENCES ORDINE(IDORDINE)
7 );

```

2.2.9 Fornitore

```

1 CREATE TABLE FORNITORE (
2     PARTITAIVA CHAR(11),
3     VIA VARCHAR2(30),
4     CIVICO NUMBER,
5     CAP NUMBER,
6     TEL NUMBER,
7     PRIMARY KEY (PARTITAIVA),
8     UNIQUE (TEL)
9 );

```

2.2.10 Fornitura

```

1 REATE TABLE FORNITURA (
2     PARTITAIVAFORNITORE CHAR(11) PRIMARY KEY,
3     QUANTITA NUMBER(2,0) NOT NULL,
4     EAN NUMBER NOT NULL,
5     CONSTRAINT FK_FORNITURA_ESTERNA1 FOREIGN KEY (PARTITAIVAFORNITORE
6     )
7     REFERENCES FORNITORE (PARTITAIVA),
8     CONSTRAINT FK_FORNITURA_ESTERNA2 FOREIGN KEY (EAN) REFERENCES
9     COMPONENTE (EAN)
10 );

```

2.2.11 Reparto

```

1 CREATE TABLE REPARTO (
2     IDREPARTO NUMBER PRIMARY KEY,
3     TELEFONO CHAR(10) UNIQUE,
4     DIPENDENTIMAX NUMBER NOT NULL
5 );

```

2.2.12 Progettazione

```

1 CREATE TABLE PROGETTAZIONE (
2     IDREPARTO NUMBER PRIMARY KEY,
3     IDPROG NUMBER,
4     CONSTRAINT FK_PROGETTAZIONE_PROGETTO FOREIGN KEY (IDPROG)
5     REFERENCES PROGETTO(IDPROG),
6     FOREIGN KEY (IDREPARTO) REFERENCES REPARTO(IDREPARTO),
7     UNIQUE (IDPROG)
8 );

```

2.2.13 Produzione

```
1 CREATE TABLE PRODUZIONE (  
2     IDREPARTO NUMBER PRIMARY KEY,  
3     IDPROG NUMBER,  
4     NMACCHINE NUMBER NOT NULL,  
5     CONSTRAINT FK_PRODUZIONE_PROGETTO FOREIGN KEY (IDPROG) REFERENCES  
6         PROGETTO(IDPROG),  
7     FOREIGN KEY (IDREPARTO) REFERENCES REPARTO(IDREPARTO),  
8     UNIQUE (IDPROG)  
9 );
```

2.2.14 Collaudo

```
1 CREATE TABLE COLLAUDO (  
2     IDREPARTO NUMBER PRIMARY KEY,  
3     NMACCHINE NUMBER NOT NULL,  
4     IDPROG NUMBER,  
5     CONSTRAINT UK_IDPROG UNIQUE (IDPROG),  
6     FOREIGN KEY (IDPROG) REFERENCES PROGETTO(IDPROG),  
7     FOREIGN KEY (IDREPARTO) REFERENCES REPARTO(IDREPARTO)  
8 );
```

2.2.15 Magazzino

```
1 CREATE TABLE MAGAZZINO (  
2     IDREPARTO NUMBER PRIMARY KEY,  
3     SPAZIOTOT NUMBER NOT NULL,  
4     FOREIGN KEY (IDREPARTO) REFERENCES REPARTO(IDREPARTO)  
5 );
```

2.2.16 Immagazzinamento

```
1 CREATE TABLE IMMAGAZZINAMENTO (  
2     IDLOTTO number NOT NULL,  
3     ORARIO DATE NOT NULL,  
4     DATAIMM DATE NOT NULL,  
5     QUANTITA NUMBER NOT NULL,  
6     PARTITAIVA CHAR(11) NOT NULL,  
7     IDREPARTO NUMBER NOT NULL,  
8     PRIMARY KEY (IDLOTTO),  
9     CONSTRAINT FK_IMMAGAZZINAMENTO01 FOREIGN KEY (PARTITAIVA)  
10         REFERENCES FORNITORE (PARTITAIVA),  
11     CONSTRAINT FK_IMMAGAZZINAMENTO03 FOREIGN KEY (IDREPARTO)  
12         REFERENCES MAGAZZINO (IDREPARTO)  
13 );
```

2.2.17 Dipendente

```
1 CREATE TABLE DIPENDENTE (  
2     IDDIPENDENTE NUMBER PRIMARY KEY,  
3     NOME VARCHAR2(30) NOT NULL,  
4     COGNOME VARCHAR2(30) NOT NULL,  
5     RAL NUMBER NOT NULL,
```

```
6     TIPODIPENDENTE VARCHAR(20) NOT NULL
7 );
```

2.2.18 Turno

```
1 CREATE TABLE TURNO (
2     IDDIPENDENTE NUMBER NOT NULL,
3     IDREPARTO NUMBER NOT NULL,
4     ORAINIZIO DATE NOT NULL,
5     ORAFINE DATE,
6     CONSTRAINT FK_TURNO_DIPENDENTE FOREIGN KEY (IDDIPENDENTE)
7         REFERENCES DIPENDENTE (IDDIPENDENTE),
8     CONSTRAINT FK_TURNO_REPARTO FOREIGN KEY (IDREPARTO)
9         REFERENCES REPARTO (IDREPARTO)
10 );
```

2.3 Data Manipulation Language

2.3.1 Popolamento tabella Cliente

```
1  INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
2  VALUES(1, 'Mario', 'Rossi', 'mario@email.com', 123456789);
3  INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
4  VALUES(2, 'Laura', 'Bianchi', 'laura@email.com', 987654321);
5  INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
6  VALUES(3, 'Luigi', 'Verdi', 'luigi@email.com', 555555555);
7  INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
8  VALUES(4, 'Giovanna', 'Ferrari', 'giovanna@email.com', 111111111);
9  INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
10 VALUES(5, 'Roberto', 'Russo', 'roberto@email.com', 999989999);
11 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
12 VALUES(6, 'Chiara', 'Romano', 'chiara@email.com', 777777777);
13 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
14 VALUES (7, 'Alessandro', 'Gallo', 'alessandro@email.com',
15          444444444);
16 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
17 VALUES(8, 'Sara', 'Rizzo', 'sara@email.com', 222222222);
18 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
19 VALUES(9, 'Marco', 'Lombardi', 'marco@email.com', 888888888);
20 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
21 VALUES(10, 'Francesca', 'Moretti', 'francesca@email.com',
22          666666666);
23 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
24 VALUES(11, 'Antonio', 'Conti', 'antonio@email.com', 333332333);
25 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
26 VALUES(12, 'Valeria', 'Marchetti', 'valeria@email.com', 555555855);
27 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
28 VALUES(13, 'Giovanni', 'Santoro', 'giovanni@email.com', 777677777);
29 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
30 VALUES(14, 'Elena', 'Gatti', 'elena@email.com', 999999959);
31 INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO)
32 VALUES(15, 'Fabio', 'Pellegrini', 'fabio@email.com', 111711111);
```

2.3.2 Popolamento tabella fattura

```
1  /*Popolamento tabella fattura*/
2  INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
3  VALUES(1, TO_DATE('2023-06-01', 'YYYY-MM-DD'), 100.50);
4  INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
5  VALUES(2, TO_DATE('2023-06-02', 'YYYY-MM-DD'), 75.20);
6  INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
7  VALUES(3, TO_DATE('2023-06-03', 'YYYY-MM-DD'), 150.00);
8  INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
9  VALUES(4, TO_DATE('2023-06-04', 'YYYY-MM-DD'), 50.80);
10 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
11 VALUES(5, TO_DATE('2023-06-05', 'YYYY-MM-DD'), 90.10);
12 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
13 VALUES(6, TO_DATE('2023-06-06', 'YYYY-MM-DD'), 120.75);
14 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
15 VALUES(7, TO_DATE('2023-06-07', 'YYYY-MM-DD'), 80.30);
16 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
17 VALUES (8, TO_DATE('2023-06-08', 'YYYY-MM-DD'), 65.50);
18 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
19 VALUES (9, TO_DATE('2023-06-09', 'YYYY-MM-DD'), 95.40);
```

```

20 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
21 VALUES (10, TO_DATE('2023-06-10', 'YYYY-MM-DD'), 110.00);
22 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
23 VALUES (11, TO_DATE('2023-06-11', 'YYYY-MM-DD'), 70.25);
24 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
25 VALUES (12, TO_DATE('2023-06-12', 'YYYY-MM-DD'), 130.80);
26 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
27 VALUES (13, TO_DATE('2023-06-13', 'YYYY-MM-DD'), 85.90);
28 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
29 VALUES (14, TO_DATE('2023-06-14', 'YYYY-MM-DD'), 45.60);
30 INSERT INTO FATTURA (IDFATTURA, DATAEMISSIONE, TOTALE)
31 VALUES (15, TO_DATE('2023-06-15', 'YYYY-MM-DD'), 105.15);

```

2.3.3 Popolamento tabella spedizione

```

1
2 /*Popolamento tabella spedizione*/
3 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
4   CITTA, CAP)
5 VALUES(1, 123456789, 'Mario', 'Rossi', 'Via Roma 1', 'Milano',
6   20121);
7 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
8   CITTA, CAP)
9 VALUES(2, 987654321, 'Laura', 'Bianchi', 'Via Verdi 5', 'Roma',
10  00185);
11 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
12  CITTA, CAP)
13 VALUES(3, 555555555, 'Luigi', 'Verdi', 'Via Garibaldi 10', 'Napoli',
14  80100);
15 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
16  CITTA, CAP)
17 VALUES(4, 111111111, 'Giovanna', 'Ferrari', 'Via Mazzini 3', '
18  Firenze', 50123);
19 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
20  CITTA, CAP)
21 VALUES(5, 999999999, 'Roberto', 'Russo', 'Via Libert 8', 'Palermo
22  ', 90100);
23 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
24  CITTA, CAP)
25 VALUES(6, 777777777, 'Chiara', 'Romano', 'Via Manzoni 12', 'Torino'
26  , 10100);
27 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
28  CITTA, CAP)
29 VALUES(7, 444444444, 'Alessandro', 'Gallo', 'Via Dante 7', 'Bologna
30  ', 40100);
31 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
32  CITTA, CAP)
33 VALUES(8, 222222222, 'Sara', 'Rizzo', 'Via Cavour 15', 'Genova',
34  16100);
35 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
36  CITTA, CAP)
37 VALUES(9, 888888888, 'Marco', 'Lombardi', 'Via Milano 4', 'Bari',
38  70100);
39 INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
40  CITTA, CAP)
41 VALUES(10, 666666666, 'Francesca', 'Moretti', 'Via Carducci 6', '
42  Verona', 37100);

```



```

23     INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
        CITTA, CAP)
24 VALUES (11, 333333333, 'Antonio', 'Conti', 'Via Garibaldi 20', '
        Trieste', 34100);
25     INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
        CITTA, CAP)
26 VALUES (12, 555555555, 'Valeria', 'Marchetti', 'Via Mazzini 3', '
        Padova', 35100);
27     INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
        CITTA, CAP)
28 VALUES (13, 777777777, 'Giovanni', 'Santoro', 'Via Leopardi 9', '
        Perugia', 06100);
29     INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
        CITTA, CAP)
30 VALUES (14, 999999999, 'Elena', 'Gatti', 'Via Manzoni 18', 'Cagliari
        ', 09100);
31     INSERT INTO SPEDIZIONE (IDSPEDIZIONE, TEL, NOME, COGNOME, VIA,
        CITTA, CAP)
32 VALUES (15, 111111111, 'Fabio', 'Pellegrini', 'Via Garibaldi 5', '
        Catania', 95100);

```

2.3.4 Popolamento tabella ordine

```

1
2  /*Popolamento tabella ordine*/
3     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
4 VALUES (1, 1, 1, 1, 3, TO_DATE('2023-06-01', 'YYYY-MM-DD'));
5     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
6 VALUES (2, 2, 2, 2, 2, TO_DATE('2023-06-02', 'YYYY-MM-DD'));
7     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
8 VALUES (3, 3, 3, 3, 1, TO_DATE('2023-06-03', 'YYYY-MM-DD'));
9     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
10 VALUES (4, 4, 4, 4, 4, TO_DATE('2023-06-04', 'YYYY-MM-DD'));
11     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
12 VALUES (5, 5, 5, 5, 2, TO_DATE('2023-06-05', 'YYYY-MM-DD'));
13     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
14 VALUES (6, 6, 6, 6, 3, TO_DATE('2023-06-06', 'YYYY-MM-DD'));
15     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
16 VALUES (7, 7, 7, 7, 2, TO_DATE('2023-06-07', 'YYYY-MM-DD'));
17     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
18 VALUES (8, 8, 8, 8, 1, TO_DATE('2023-06-08', 'YYYY-MM-DD'));
19     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
20 VALUES (9, 9, 9, 9, 4, TO_DATE('2023-06-09', 'YYYY-MM-DD'));
21     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
22 VALUES (10, 10, 10, 10, 3, TO_DATE('2023-06-10', 'YYYY-MM-DD'));
23     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)

```

```

24 VALUES (11, 11, 11, 11, 2, TO_DATE('2023-06-11', 'YYYY-MM-DD'));
25     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
26 VALUES (12, 12, 12, 12, 1, TO_DATE('2023-06-12', 'YYYY-MM-DD'));
27     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
28 VALUES (13, 13, 13, 13, 4, TO_DATE('2023-06-13', 'YYYY-MM-DD'));
29     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
30 VALUES (14, 14, 14, 14, 2, TO_DATE('2023-06-14', 'YYYY-MM-DD'));
31     INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,
        NUMEROPRODOTTI, DATACREAZIONE)
32 VALUES (15, 15, 15, 15, 3, TO_DATE('2023-06-15', 'YYYY-MM-DD'));

```

2.3.5 Popolamento tabella componente

```

1
2 /*Popolamento tabella componente*/
3     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
4 VALUES (1001, 'Componente 1', 10.99);
5     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
6 VALUES (1002, 'Componente 2', 12.99);
7     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
8 VALUES (1003, 'Componente 3', 8.99);
9     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
10 VALUES (1004, 'Componente 4', 14.99);
11     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
12 VALUES (1005, 'Componente 5', 9.99);
13     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
14 VALUES (1006, 'Componente 6', 11.99);
15     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
16 VALUES (1007, 'Componente 7', 13.99);
17     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
18 VALUES (1008, 'Componente 8', 7.99);
19     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
20 VALUES (1009, 'Componente 9', 16.99);
21     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
22 VALUES (1010, 'Componente 10', 10.49);
23     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
24 VALUES (1011, 'Componente 11', 9.49);
25     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
26 VALUES (1012, 'Componente 12', 15.99);
27     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
28 VALUES (1013, 'Componente 13', 11.49);
29     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
30 VALUES (1014, 'Componente 14', 8.49);
31     INSERT INTO COMPONENTE (EAN, NOME, PREZZO)
32 VALUES (1015, 'Componente 15', 13.49);

```

2.3.6 Popolamento tabella progetto

```

1
2 /*Popolamento tabella progetto*/
3 INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
4 VALUES (1, 'Progetto 1', TO_DATE('2023-01-01', 'YYYY-MM-DD'));
5     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)

```

```

6 VALUES (2, 'Progetto 2', TO_DATE('2023-02-01', 'YYYY-MM-DD'));
7     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
8 VALUES (3, 'Progetto 3', TO_DATE('2023-03-01', 'YYYY-MM-DD'));
9     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
10 VALUES (4, 'Progetto 4', TO_DATE('2023-04-01', 'YYYY-MM-DD'));
11     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
12 VALUES (5, 'Progetto 5', TO_DATE('2023-05-01', 'YYYY-MM-DD'));
13     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
14 VALUES (6, 'Progetto 6', TO_DATE('2023-06-01', 'YYYY-MM-DD'));
15     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
16 VALUES (7, 'Progetto 7', TO_DATE('2023-07-01', 'YYYY-MM-DD'));
17     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
18 VALUES (8, 'Progetto 8', TO_DATE('2023-08-01', 'YYYY-MM-DD'));
19     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
20 VALUES (9, 'Progetto 9', TO_DATE('2023-09-01', 'YYYY-MM-DD'));
21     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
22 VALUES (10, 'Progetto 10', TO_DATE('2023-10-01', 'YYYY-MM-DD'));
23     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
24 VALUES (11, 'Progetto 11', TO_DATE('2023-11-01', 'YYYY-MM-DD'));
25     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
26 VALUES (12, 'Progetto 12', TO_DATE('2023-12-01', 'YYYY-MM-DD'));
27     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
28 VALUES (13, 'Progetto 13', TO_DATE('2024-01-01', 'YYYY-MM-DD'));
29     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
30 VALUES (14, 'Progetto 14', TO_DATE('2024-02-01', 'YYYY-MM-DD'));
31     INSERT INTO PROGETTO (IDPROG, NOME, DATAINIZIO)
32 VALUES (15, 'Progetto 15', TO_DATE('2024-03-01', 'YYYY-MM-DD'));

```

2.3.7 Popolamento tabella componentiProgetto

```

1
2 /*Popolamento tabella componentiprogetto*/
3     INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
4         (1, 1001, 2);
5     INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
6         (1, 1002, 1);
7     INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
8         (1, 1003, 3);
9     INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
10        (2, 1001, 4);
11    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
12        (2, 1004, 2);
13    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
14        (3, 1002, 2);
15    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
16        (3, 1005, 1);
17    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
18        (4, 1003, 3);
19    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
20        (5, 1001, 2);
21    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
22        (5, 1004, 1);
23    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
24        (6, 1002, 3);
25    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
26        (7, 1005, 2);
27    INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
28        (8, 1001, 1);

```

```

16  INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
    (9, 1001, 3);
17  INSERT INTO COMPONENTIPROGETTO (IDPROG, EAN, QUANTITA) VALUES
    (10, 1003, 2);

```

2.3.8 Popolamento tabella contiene

```

1
2  /*Popolamento tabella contiene*/
3  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (1, 1,
    2);
4  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (1, 2,
    1);
5  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (2, 3,
    3);
6  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (2, 4, 4)
    ;
7  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (3, 5, 2)
    ;
8  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (3, 6, 1)
    ;
9  INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (4, 7, 3)
    ;
10 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (4, 8, 2)
    ;
11 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (5, 9, 1)
    ;
12 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (5, 10,
    2);
13 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (6, 11,
    3);
14 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (7, 12,
    2);
15 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (8, 13,
    1);
16 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (9, 14,
    4);
17 INSERT INTO CONTIENE (IDPROG, IDORDINE, QUANTITA) VALUES (10, 15,
    3);

```

2.3.9 Popolamento tabella fornitore

```

1
2  /*Popolamento tabella fornitore*/
3  INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
4  VALUES ('12345678901', 'Via Roma', 10, 'Milano', 20100, '0123456789
    ');
5  INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
6  VALUES ('23456789012', 'Via Verdi', 20, 'Roma', 00100, '0234567890
    ');
7  INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
8  VALUES ('34567890123', 'Via Italia', 30, 'Napoli', 80100, '
    0345678901');
9  INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
10 VALUES ('45678901234', 'Via Firenze', 40, 'Torino', 10100, '
    0456789012');

```

```

11     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
12     VALUES ('56789012345', 'Via Venezia', 50, 'Firenze', 50100, '
13         0567890123');
14     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
15     VALUES ('67890123456', 'Via Bologna', 60, 'Palermo', 90100, '
16         0678901234');
17     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
18     VALUES ('78901234567', 'Via Genova', 70, 'Genova', 16100, '
19         0789012345');
20     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
21     VALUES ('89012345678', 'Via Milano', 80, 'Bologna', 40100, '
22         0890123456');
23     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
24     VALUES ('90123456789', 'Via Napoli', 90, 'Verona', 37100, '
25         0901234567');
26     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
27     VALUES ('01234567890', 'Via Torino', 100, 'Venezia', 30100, '
28         0012345678');
29     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
30     VALUES ('11223344556', 'Via Palermo', 110, 'Cagliari', 90100, '
31         1122334455');
32     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
33     VALUES ('22334455667', 'Via Genova', 120, 'Napoli', 80100, '
34         2233445566');
35     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
36     VALUES ('33445566778', 'Via Firenze', 130, 'Milano', 20100, '
37         3344556677');
38     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
39     VALUES ('44556677889', 'Via Roma', 140, 'Roma', 00100, '4455667788'
40         );
41     INSERT INTO FORNITORE (PARTITAIVA, VIA, CIVICO, CITTA, CAP, TEL)
42     VALUES ('55667788990', 'Via Verdi', 150, 'Torino', 10100, '
43         5566778899');

```

2.3.10 Popolamento tabella fornitura

```

1
2     /*Popolamento tabella fornitura*/
3     INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
4         , DATAACQUISTO)
5     VALUES ('12345678901', 1001, 5, TO_DATE('2023-06-01', 'YYYY-MM-DD')
6         );
7     INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
8         , DATAACQUISTO)
9     VALUES ('23456789012', 1002, 3, TO_DATE('2023-06-02', 'YYYY-MM-DD')
10        );
11    INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
12        , DATAACQUISTO)
13    VALUES ('34567890123', 1003, 7, TO_DATE('2023-06-03', 'YYYY-MM-DD')
14        );
15    INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
16        , DATAACQUISTO)
17    VALUES ('45678901234', 1004, 4, TO_DATE('2023-06-04', 'YYYY-MM-DD')
18        );
19    INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
20        , DATAACQUISTO)
21    VALUES ('56789012345', 1005, 10, TO_DATE('2023-06-05', 'YYYY-MM-DD')
22        );

```

```

13 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
14 VALUES ('67890123456', 1006, 6, TO_DATE('2023-06-06', 'YYYY-MM-DD')
    );
15 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
16 VALUES ('78901234567', 1007, 2, TO_DATE('2023-06-07', 'YYYY-MM-DD')
    );
17 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
18 VALUES ('89012345678', 1008, 8, TO_DATE('2023-06-08', 'YYYY-MM-DD')
    );
19 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
20 VALUES ('90123456789', 1009, 9, TO_DATE('2023-06-09', 'YYYY-MM-DD')
    );
21 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
22 VALUES ('01234567890', 1010, 7, TO_DATE('2023-06-10', 'YYYY-MM-DD')
    );
23 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
24 VALUES ('11223344556', 1011, 3, TO_DATE('2023-06-11', 'YYYY-MM-DD')
    );
25 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
26 VALUES ('22334455667', 1012, 6, TO_DATE('2023-06-12', 'YYYY-MM-DD')
    );
27 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
28 VALUES ('33445566778', 1013, 4, TO_DATE('2023-06-13', 'YYYY-MM-DD')
    );
29 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
30 VALUES ('44556677889', 1014, 10, TO_DATE('2023-06-14', 'YYYY-MM-DD'
    ));
31 INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE, QUANTITA
    , DATAACQUISTO)
32 VALUES ('55667788990', 1015, 8, TO_DATE('2023-06-15', 'YYYY-MM-DD')
    );

```

2.3.11 Popolamento tabella reparto

```

1
2 /*Popolamento tabella reparto*/
3 INSERT INTO REPARTO (IDREPARTO, TELEFONO, DIPENDENTIMAX) VALUES
4 (1, '1234567890', 5);
5 INSERT INTO REPARTO (IDREPARTO, TELEFONO, DIPENDENTIMAX) VALUES
6 (2, '9876543210', 5);
7 INSERT INTO REPARTO (IDREPARTO, TELEFONO, DIPENDENTIMAX) VALUES
8 (3, '1111111111', 5);
9 INSERT INTO REPARTO (IDREPARTO, TELEFONO, DIPENDENTIMAX) VALUES
10 (4, '2222222222', 5);
11 INSERT INTO REPARTO (IDREPARTO, TELEFONO, DIPENDENTIMAX) VALUES
12 (5, '3333333333', 5);
13 INSERT INTO REPARTO (IDREPARTO, TELEFONO, DIPENDENTIMAX) VALUES
14 (6, '4444444444', 5);

```

```

15 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
16 (7, '5555555555', 5);
17 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
18 (8, '6666666666', 5);
19 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
20 (9, '7777777777', 5);
21 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
22 (10, '8888888888', 5);
23 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
24 (11, '9999999999', 5);
25 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
26 (12, '1010101010', 5);
27 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
28 (13, '1212121212', 5);
29 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
30 (14, '1313131313', 5);
31 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
32 (15, '1414141414', 5);
33 INSERT INTO REPARTO (IDREPARTO , TELEFONO , DIPENDENTIMAX) VALUES
34 (16, '1515151515', 5);

```

2.3.12 Popolamento tabella progettazione

```

1
2 /*Popolamento tabella progettazione*/
3 INSERT INTO PROGETTAZIONE (IDREPARTO , STIMATEMPO , IDPROG) VALUES
4 (1, 35, 1);
5 INSERT INTO PROGETTAZIONE (IDREPARTO , STIMATEMPO , IDPROG) VALUES
6 (2, 22, 2);
7 INSERT INTO PROGETTAZIONE (IDREPARTO , STIMATEMPO , IDPROG) VALUES
8 (3, 15, 3);
9 INSERT INTO PROGETTAZIONE (IDREPARTO , STIMATEMPO , IDPROG) VALUES
10 (4, 26, 4);

```

2.3.13 Popolamento tabella collaudo

```

1 /*Popolamento tabella collaudo*/
2 INSERT INTO COLLAUDO (IDREPARTO , NMACCHINE) VALUES
3 (9, 12);
4 INSERT INTO COLLAUDO (IDREPARTO , NMACCHINE) VALUES
5 (10, 32);
6 INSERT INTO COLLAUDO (IDREPARTO , NMACCHINE) VALUES
7 (11, 22);
8 INSERT INTO COLLAUDO (IDREPARTO , NMACCHINE) VALUES
9 (12, 15);

```

```

1 /*Popolamento tabella produzione*/
2 INSERT INTO produzione (IDREPARTO , NMACCHINE) VALUES
3 (5, 12);
4 INSERT INTO produzione (IDREPARTO , NMACCHINE) VALUES
5 (6, 16);
6 INSERT INTO produzione (IDREPARTO , NMACCHINE) VALUES
7 (7, 17);
8 INSERT INTO produzione (IDREPARTO , NMACCHINE) VALUES
9 (8, 21);

```

2.3.14 Popolamento tabella magazzino

```
1
2 /*Popolamento tabella magazzino*/
3 INSERT INTO MAGAZZINO(IDREPARTO, SPAZIOTOT) VALUES
4 (13,50);
5 INSERT INTO MAGAZZINO(IDREPARTO, SPAZIOTOT) VALUES
6 (14,50);
7 INSERT INTO MAGAZZINO(IDREPARTO, SPAZIOTOT) VALUES
8 (15,50);
9 INSERT INTO MAGAZZINO(IDREPARTO, SPAZIOTOT) VALUES
10 (16,50);
```

2.3.15 Popolamento tabella immagazzinamento

```
1
2 /*Popolamento tabella immagazzinamento*/
3 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
4 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
5 VALUES (1, 10, TO_DATE('2023-06-13 08:30:00', 'YYYY-MM-DD HH24:MI
6 :SS'), TO_DATE('2023-06-13', 'YYYY-MM-DD'), 50, '12345678901',
7 1001, 16);
8 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
9 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
10 VALUES (2, 15, TO_DATE('2023-06-14 09:15:00', 'YYYY-MM-DD HH24:MI
11 :SS'), TO_DATE('2023-06-14', 'YYYY-MM-DD'), 30, '23456789012',
12 1002, 16);
13 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
14 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
15 VALUES (3, 8, TO_DATE('2023-06-15 10:45:00', 'YYYY-MM-DD HH24:MI:
16 SS'), TO_DATE('2023-06-15', 'YYYY-MM-DD'), 20, '45678901234',
17 1002, 13);
18 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
19 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
20 VALUES (4, 12, TO_DATE('2023-06-16 14:20:00', 'YYYY-MM-DD HH24:MI
21 :SS'), TO_DATE('2023-06-16', 'YYYY-MM-DD'), 40, '01234567890',
22 1004, 14);
23 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
24 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
25 VALUES (5, 5, TO_DATE('2023-06-17 11:10:00', 'YYYY-MM-DD HH24:MI:
26 SS'), TO_DATE('2023-06-17', 'YYYY-MM-DD'), 10, '89012345678',
27 1005, 15);
28 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
29 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
30 VALUES (6, 20, TO_DATE('2023-06-18 13:45:00', 'YYYY-MM-DD HH24:MI
31 :SS'), TO_DATE('2023-06-18', 'YYYY-MM-DD'), 60, '23456789012',
32 1007, 16);
33 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
34 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
35 VALUES (7, 7, TO_DATE('2023-06-19 09:55:00', 'YYYY-MM-DD HH24:MI:
36 SS'), TO_DATE('2023-06-19', 'YYYY-MM-DD'), 15, '34567890123',
37 1008, 14);
38 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
39 DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
40 VALUES (8, 14, TO_DATE('2023-06-20 12:30:00', 'YYYY-MM-DD HH24:MI
41 :SS'), TO_DATE('2023-06-20', 'YYYY-MM-DD'), 35, '56789012345',
42 1014, 15);
```



```

19 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
20 VALUES (9, 9, TO_DATE('2023-06-21 15:05:00', 'YYYY-MM-DD HH24:MI:
    SS'), TO_DATE('2023-06-21', 'YYYY-MM-DD'), 25, '89012345678',
    1015, 16);
21 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
22 VALUES (10, 18, TO_DATE('2023-06-22 10:40:00', 'YYYY-MM-DD HH24:
    MI:SS'), TO_DATE('2023-06-22', 'YYYY-MM-DD'), 55, '01234567890
    ', 1010, 13);
23 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
24 VALUES (11, 6, TO_DATE('2023-06-23 13:15:00', 'YYYY-MM-DD HH24:MI
    :SS'), TO_DATE('2023-06-23', 'YYYY-MM-DD'), 12, '12345678901',
    1013, 13);
25 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
26 VALUES (12, 13, TO_DATE('2023-06-24 11:25:00', 'YYYY-MM-DD HH24:
    MI:SS'), TO_DATE('2023-06-24', 'YYYY-MM-DD'), 28, '23456789012
    ', 1015, 15);
27 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
28 VALUES (13, 8, TO_DATE('2023-06-25 14:00:00', 'YYYY-MM-DD HH24:MI
    :SS'), TO_DATE('2023-06-25', 'YYYY-MM-DD'), 18, '34567890123',
    1013, 13);
29 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
30 VALUES (14, 16, TO_DATE('2023-06-26 11:50:00', 'YYYY-MM-DD HH24:
    MI:SS'), TO_DATE('2023-06-26', 'YYYY-MM-DD'), 45, '45678901234
    ', 1008, 14);
31 INSERT INTO IMMAGAZZINAMENTO (IDLOTTO, SPAZIOOCCUPATO, ORARIO,
    DATAIMM, QUANTITA, PARTITAIVA, EAN, IDREPARTO)
32 VALUES (15, 11, TO_DATE('2023-06-27 13:25:00', 'YYYY-MM-DD HH24:
    MI:SS'), TO_DATE('2023-06-27', 'YYYY-MM-DD'), 33, '56789012345
    ', 1015, 15);

```

2.3.16 Popolamento tabella dipendente

```

1
2 /*Popolamento tabella dipendente*/
3 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (1, 'Mario', 'Rossi', 30000, '
    Magazziniere');
4 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (2, 'Luca', 'Bianchi', 25000, 'Operaio');
5 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (3, 'Laura', 'Verdi', 35000, 'Ingegnere')
    ;
6 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (4, 'Giovanni', 'Russo', 28000, '
    Responsabile');
7 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (5, 'Francesca', 'Esposito', 32000, '
    Collaudatore');
8 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (6, 'Marco', 'Romano', 27000, '
    Magazziniere');

```

```

9  INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (7, 'Sara', 'Galli', 38000, 'Operaio');
10 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (8, 'Paolo', 'Conti', 29000, 'Ingegnere')
    ;
11 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (9, 'Alessia', 'Ferrari', 33000, '
    Responsabile');
12 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (10, 'Simone', 'Martini', 26000, '
    Collaudatore');
13 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (11, 'Valentina', 'Rizzo', 31000, '
    Magazziniere');
14 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (12, 'Fabio', 'Colombo', 24000, 'Operaio'
    );
15 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (13, 'Elisa', 'Santoro', 37000, '
    Ingegnere');
16 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (14, 'Giuseppe', 'Moretti', 28000, '
    Responsabile');
17 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (15, 'Caterina', 'Gentile', 32000, '
    Collaudatore');
18 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (16, 'Antonio', 'Lombardi', 27000, '
    Magazziniere');
19 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (17, 'Elena', 'Pellegrini', 38000, '
    Operaio');
20 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (18, 'Roberto', 'Ricci', 29000, '
    Ingegnere');
21 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (19, 'Monica', 'Ferri', 33000, '
    Responsabile');
22 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (20, 'Davide', 'Greco', 26000, '
    Collaudatore');
23 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (21, 'Silvia', 'Marini', 31000, '
    Magazziniere');
24 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (22, 'Lorenzo', 'Fabbri', 24000, 'Operaio
    ');
25 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (23, 'Federica', 'Bellini', 37000, '
    Ingegnere');
26 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (24, 'Andrea', 'Gatti', 28000, 'Ingegnere
    ');
27 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (25, 'Martina', 'Palmieri', 32000, '
    Collaudatore');
28 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (26, 'Gabriele', 'Barbieri', 27000, '

```

```

    Magazziniere');
29 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (27, 'Chiara', 'Serra', 38000, 'Operaio')
    ;
30 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (28, 'Stefano', 'Mancini', 29000, '
    Ingegnere');
31 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (29, 'Giulia', 'Villa', 33000, 'Operaio')
    ;
32 INSERT INTO DIPENDENTE (IDDIPENDENTE, NOME, COGNOME, RAL,
    TIPODIPENDENTE) VALUES (30, 'Mattia', 'Rinaldi', 26000, '
    Collaudatore');

```

2.3.17 Popolamento tabella turno

```

1
2 /*Popolamento tabella turno*/
3 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
4     (1, 1, TO_DATE('2023-06-01 08:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 16:00:00', 'YYYY-MM-DD HH24:MI:SS'));
5 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
6     (2, 2, TO_DATE('2023-06-01 13:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 21:00:00', 'YYYY-MM-DD HH24:MI:SS'));
7 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
8     (3, 3, TO_DATE('2023-06-01 09:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 17:00:00', 'YYYY-MM-DD HH24:MI:SS'));
9 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
10     (4, 1, TO_DATE('2023-06-01 10:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 18:00:00', 'YYYY-MM-DD HH24:MI:SS'));
11 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
12     (5, 2, TO_DATE('2023-06-01 14:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 22:00:00', 'YYYY-MM-DD HH24:MI:SS'));
13 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
14     (6, 3, TO_DATE('2023-06-01 11:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 19:00:00', 'YYYY-MM-DD HH24:MI:SS'));
15 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
16     (7, 1, TO_DATE('2023-06-01 12:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 20:00:00', 'YYYY-MM-DD HH24:MI:SS'));
17 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
18     (8, 2, TO_DATE('2023-06-01 15:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-01 23:00:00', 'YYYY-MM-DD HH24:MI:SS'));
19 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
20     (9, 3, TO_DATE('2023-06-02 08:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 16:00:00', 'YYYY-MM-DD HH24:MI:SS'));
21 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
22     (10, 1, TO_DATE('2023-06-02 13:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 21:00:00', 'YYYY-MM-DD HH24:MI:SS'));

```

```

23 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
24     (11, 2, TO_DATE('2023-06-02 09:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 17:00:00', 'YYYY-MM-DD HH24:MI:SS'));
25 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
26     (12, 3, TO_DATE('2023-06-02 10:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 18:00:00', 'YYYY-MM-DD HH24:MI:SS'));
27 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
28     (13, 1, TO_DATE('2023-06-02 14:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 22:00:00', 'YYYY-MM-DD HH24:MI:SS'));
29 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
30     (14, 2, TO_DATE('2023-06-02 11:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 19:00:00', 'YYYY-MM-DD HH24:MI:SS'));
31 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
32     (15, 3, TO_DATE('2023-06-02 12:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 20:00:00', 'YYYY-MM-DD HH24:MI:SS'));
33 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
34     (16, 1, TO_DATE('2023-06-02 15:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-02 23:00:00', 'YYYY-MM-DD HH24:MI:SS'));
35 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
36     (17, 2, TO_DATE('2023-06-03 08:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 16:00:00', 'YYYY-MM-DD HH24:MI:SS'));
37 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
38     (18, 3, TO_DATE('2023-06-03 13:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 21:00:00', 'YYYY-MM-DD HH24:MI:SS'));
39 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
40     (19, 1, TO_DATE('2023-06-03 09:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 17:00:00', 'YYYY-MM-DD HH24:MI:SS'));
41 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
42     (20, 2, TO_DATE('2023-06-03 10:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 18:00:00', 'YYYY-MM-DD HH24:MI:SS'));
43 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
44     (21, 3, TO_DATE('2023-06-03 14:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 22:00:00', 'YYYY-MM-DD HH24:MI:SS'));
45 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
46     (22, 1, TO_DATE('2023-06-03 11:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 19:00:00', 'YYYY-MM-DD HH24:MI:SS'));
47 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
48     (23, 2, TO_DATE('2023-06-03 12:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-03 20:00:00', 'YYYY-MM-DD HH24:MI:SS'));
49 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES
50     (24, 3, TO_DATE('2023-06-04 08:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 16:00:00', 'YYYY-MM-DD HH24:MI:SS'));
51 INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
    VALUES

```

```

52      (25, 1, TO_DATE('2023-06-04 13:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 21:00:00', 'YYYY-MM-DD HH24:MI:SS'));
53  INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
        VALUES
54      (26, 2, TO_DATE('2023-06-04 09:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 17:00:00', 'YYYY-MM-DD HH24:MI:SS'));
55  INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
        VALUES
56      (27, 3, TO_DATE('2023-06-04 10:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 18:00:00', 'YYYY-MM-DD HH24:MI:SS'));
57  INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
        VALUES
58      (28, 1, TO_DATE('2023-06-04 14:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 22:00:00', 'YYYY-MM-DD HH24:MI:SS'));
59  INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
        VALUES
60      (29, 2, TO_DATE('2023-06-04 11:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 19:00:00', 'YYYY-MM-DD HH24:MI:SS'));
61  INSERT INTO TURNO (IDDIPENDENTE, IDREPARTO, ORAINIZIO, ORAFINE)
        VALUES
62      (30, 3, TO_DATE('2023-06-04 12:00:00', 'YYYY-MM-DD HH24:MI:SS'),
        TO_DATE('2023-06-04 20:00:00', 'YYYY-MM-DD HH24:MI:SS'));

```

2.4 Procedure e funzioni

Di seguito le funzioni illustrate nel paragrafo 1.6 (Use case)

2.4.1 Cliente ordina un progetto

```
1 CREATE OR REPLACE FUNCTION ordina_progetto(  
2   p_nome_cliente IN CLIENTE.NOME%TYPE,  
3   p_cognome_cliente IN CLIENTE.COGNOME%TYPE,  
4   p_email_cliente IN CLIENTE.EMAIL%TYPE,  
5   p_telefono_cliente IN CLIENTE.TELEFONO%TYPE,  
6   p_via_cliente IN CLIENTE.VIA%TYPE,  
7   p_citta_cliente IN CLIENTE.CITTA%TYPE,  
8   p_cap_cliente IN CLIENTE.CAP%TYPE,  
9   p_nome_progetto IN PROGETTO.NOME%TYPE  
10 ) RETURN NUMBER  
11 IS  
12   v_id_cliente CLIENTE.IDCLIENTE%TYPE;  
13   v_id_ordine ORDINE.IDORDINE%TYPE;  
14   v_id_progetto PROGETTO.IDPROG%TYPE;  
15 BEGIN  
16   -- Verifica se il cliente esiste gi  o se deve essere creato  
17   SELECT IDCLIENTE INTO v_id_cliente  
18   FROM CLIENTE  
19   WHERE NOME = p_nome_cliente  
20         AND COGNOME = p_cognome_cliente  
21         AND EMAIL = p_email_cliente  
22         AND TELEFONO = p_telefono_cliente  
23         AND VIA = p_via_cliente  
24         AND CITTA = p_citta_cliente  
25         AND CAP = p_cap_cliente;  
26  
27   -- Se il cliente non esiste, crealo  
28   IF v_id_cliente IS NULL THEN  
29     INSERT INTO CLIENTE (IDCLIENTE, NOME, COGNOME, EMAIL, TELEFONO,  
30       VIA, CITTA, CAP)  
31     VALUES (SEQ_CLIENTE.NEXTVAL, p_nome_cliente, p_cognome_cliente,  
32       p_email_cliente, p_telefono_cliente, p_via_cliente,  
33       p_citta_cliente, p_cap_cliente);  
34  
35     v_id_cliente := SEQ_CLIENTE.CURRVAL;  
36   END IF;  
37  
38   -- Ottieni l'ID del progetto  
39   SELECT IDPROG INTO v_id_progetto  
40   FROM PROGETTO  
41   WHERE NOME = p_nome_progetto;  
42  
43   -- Crea un nuovo ordine  
44   INSERT INTO ORDINE (IDORDINE, IDFATTURA, IDCLIENTE, IDSPEDIZIONE,  
45     NUMEROPRODOTTI, DATACREAZIONE)  
46   VALUES (SEQ_ORDINE.NEXTVAL, NULL, v_id_cliente, NULL, 0, SYSDATE)  
47   ;  
48  
49   v_id_ordine := SEQ_ORDINE.CURRVAL;  
50  
51   -- Aggiorna l'ordine con l'ID del progetto  
52   UPDATE ORDINE  
53   SET IDPROG = v_id_progetto
```

```

49 WHERE IDORDINE = v_id_ordine;
50
51 COMMIT;
52
53 RETURN v_id_ordine;
54 END;
55 /
56 /*Questa funzione accetta come parametri i dettagli del cliente (
    nome, cognome, email, telefono, via, citt , CAP) e il nome del
    progetto.
57 La funzione verifica se il cliente esiste gi nella tabella "
    Cliente" in base alle informazioni fornite. Se il cliente non
    esiste, viene
58 creato un nuovo record nella tabella "Cliente". Successivamente,
    viene recuperato l'ID del progetto dalla tabella "Progetto".
    Infine,
59 viene creato un nuovo ordine nella tabella "Ordine" con i dettagli
    del cliente e l'ID del progetto. La funzione restituisce l'ID
    dell'ordine creato.*/

```

2.4.2 Cliente cancella un progetto dall'ordine

```

1 CREATE OR REPLACE FUNCTION cancella_progetto_ordine(
2   p_id_ordine IN ORDINE.IDORDINE%TYPE,
3   p_id_progetto IN PROGETTO.IDPROG%TYPE
4 ) RETURN BOOLEAN
5 IS
6   v_num_progetti NUMBER;
7 BEGIN
8   -- Controlla il numero di progetti presenti nell'ordine
9   SELECT COUNT(*)
10  INTO v_num_progetti
11  FROM ORDINE o
12  JOIN PROGETTO p ON o.IDPROG = p.IDPROG
13  WHERE o.IDORDINE = p_id_ordine;
14
15   -- Verifica se l'ordine esiste e contiene pi di due progetti
16  IF v_num_progetti > 2 THEN
17    -- Cancella il progetto dall'ordine
18    DELETE FROM ORDINE
19    WHERE IDORDINE = p_id_ordine
20    AND IDPROG = p_id_progetto;
21
22    COMMIT;
23
24    RETURN TRUE;
25  ELSE
26    RETURN FALSE;
27  END IF;
28  EXCEPTION
29    WHEN OTHERS THEN
30      RETURN FALSE;
31  END;
32  /
33  /*Questa funzione accetta l'ID dell'ordine e l'ID del progetto come
    parametri. Utilizza una query per contare il numero
34  di progetti presenti nell'ordine specificato. Se il numero di
    progetti maggiore di due, viene eliminato il progetto

```

```

35  specificato dall'ordine. Viene restituito il valore TRUE se il
    progetto viene cancellato con successo, altrimenti viene
36  restituito il valore FALSE.*/

```

2.4.3 Fornitore inserisce una nuova fornitura

```

1  CREATE OR REPLACE PROCEDURE inserisci_fornitura(
2      IN_PARTITAIVAFORNITORE IN VARCHAR2,
3      IN_EANCOMPONENTE IN VARCHAR2,
4      IN_QUANTITA IN NUMBER,
5      IN_DATAACQUISTO IN DATE
6  )
7  AS
8      V_EXISTE_FORNITORE NUMBER;
9      V_EXISTE_COMPONENTE NUMBER;
10 BEGIN
11     -- Verifica se il fornitore esiste nella tabella FORNITORE
12     SELECT COUNT(*) INTO V_EXISTE_FORNITORE
13     FROM FORNITORE
14     WHERE PARTITAIVA = IN_PARTITAIVAFORNITORE;
15
16     IF V_EXISTE_FORNITORE = 0 THEN
17         RAISE_APPLICATION_ERROR(-20001, 'Il fornitore specificato
18             non esiste. ');
19     END IF;
20
21     -- Verifica se il componente esiste nella tabella COMPONENTE
22     SELECT COUNT(*) INTO V_EXISTE_COMPONENTE
23     FROM COMPONENTE
24     WHERE EAN = IN_EANCOMPONENTE;
25
26     IF V_EXISTE_COMPONENTE = 0 THEN
27         RAISE_APPLICATION_ERROR(-20002, 'Il componente specificato
28             non esiste. ');
29     END IF;
30
31     -- Inserisce la nuova riga nella tabella FORNITURA
32     INSERT INTO FORNITURA (PARTITAIVAFORNITORE, EANCOMPONENTE,
33         QUANTITA, DATAACQUISTO)
34     VALUES (IN_PARTITAIVAFORNITORE, IN_EANCOMPONENTE, IN_QUANTITA,
35         IN_DATAACQUISTO);
36
37     COMMIT;
38 EXCEPTION
39     WHEN OTHERS THEN
40         ROLLBACK;
41         RAISE;
42 END;
43 /
44
45 /*Viene verificato se il fornitore esiste nella tabella "FORNITORE"
46    contando il numero di righe corrispondenti
47    alla partita IVA specificata. Se il contatore uguale a zero,
48    viene sollevato un errore con un messaggio appropriato.
49    Viene verificato se il componente esiste nella tabella "COMPONENTE"
50    contando il numero di righe corrispondenti
51    al codice EAN specificato. Se il contatore uguale a zero, viene
52    sollevato un errore con un messaggio appropriato.

```



```

45 Viene eseguita un'istruzione di inserimento nella tabella "
    FORNITURA" con i valori passati come parametri.
46 Viene eseguito un commit per confermare la transazione.
47 In caso di errori durante l'esecuzione della procedura, viene
    eseguito un rollback per annullare la transazione
48 e viene sollevata un'eccezione per segnalare l'errore.
49 In sostanza, la procedura consente di inserire una nuova riga nella
    tabella "FORNITURA" dopo aver verificato che
50 il fornitore e il componente specificati esistano nelle rispettive
    tabelle.*/

```

2.4.4 Stima di consegna di un progetto

```

1  CREATE OR REPLACE FUNCTION stima_consegna_progetto(
2      IN_IDCLIENTE IN NUMBER,
3      IN_IDORDINE  IN NUMBER,
4      IN_IDPROGETTO IN NUMBER
5  ) RETURN DATE
6  IS
7      V_DATAINIZIO PROGETTO.DATAINIZIO%TYPE;
8      V_STIMA_GIORNI ORDINE.NUMEROPRODOTTI%TYPE;
9      V_CONSEGNA_EFFETTIVA DATE;
10 BEGIN
11     -- Recupera la data di inizio del progetto
12     SELECT DATAINIZIO INTO V_DATAINIZIO
13     FROM PROGETTO
14     WHERE IDPROG = IN_IDPROGETTO;
15
16     -- Calcola la stima dei giorni di consegna in base al numero di
        prodotti nell'ordine
17     SELECT NUMEROPRODOTTI INTO V_STIMA_GIORNI
18     FROM ORDINE
19     WHERE IDORDINE = IN_IDORDINE;
20
21     -- Calcola la consegna effettiva aggiungendo la stima dei
        giorni alla data di inizio del progetto
22     V_CONSEGNA_EFFETTIVA := V_DATAINIZIO + V_STIMA_GIORNI;
23
24     RETURN V_CONSEGNA_EFFETTIVA;
25 EXCEPTION
26     WHEN NO_DATA_FOUND THEN
27         RETURN NULL;
28 END;
29 /
30
31 /*La funzione STIMA_CONSEGNA_PROGETTO prende in input l'ID del
    cliente, l'ID dell'ordine e l'ID del progetto.
32 Recupera la data di inizio del progetto e calcola la stima dei
    giorni di consegna basata sul numero di prodotti
33 nell'ordine. Successivamente, calcola la data di consegna effettiva
    aggiungendo la stima dei giorni alla data
34 di inizio del progetto. Infine, restituisce la data di consegna
    effettiva.*/

```

2.5 Viste

Una vista in basi di dati è una rappresentazione virtuale di un insieme di dati preesistenti all'interno del database. Essa è costituita da una query che definisce una visualizzazione specifica dei dati presenti in una o più tabelle del database. La vista permette di accedere e manipolare i dati in modo strutturato, come se fossero contenuti in una tabella reale

2.5.1 Vista ordine cliente

```
1 --Vista per visualizzare i dettagli dell'ordine insieme alle
  informazioni del cliente:
2 CREATE VIEW VISTA_ORDINE_CLIENTE AS
3 SELECT o.IDORDINE, o.DATACREAZIONE, c.NOME, c.COGNOME, c.EMAIL
4 FROM ORDINE o
5 JOIN CLIENTE c ON o.IDCLIENTE = c.IDCLIENTE;
```

2.5.2 Vista fattura dettagliata

```
1 --Vista per visualizzare i dettagli completi di una fattura,
  inclusi i dettagli dell'ordine e le informazioni del cliente:
2
3 CREATE VIEW VISTA_FATTURA_DETTAGLIATA AS
4 SELECT f.IDFATTURA, f.DATAEMISSIONE, f.TOTALE, o.IDORDINE, o.
  DATACREAZIONE, c.NOME, c.COGNOME, c.EMAIL
5 FROM FATTURA f
6 JOIN ORDINE o ON f.IDFATTURA = o.IDFATTURA
7 JOIN CLIENTE c ON o.IDCLIENTE = c.IDCLIENTE;
```

2.5.3 Vista componenti progetto

```
1 --Vista per visualizzare i componenti associati a ciascun progetto
  con le relative quantit :
2 CREATE VIEW VISTA_COMPONENTI_PROGETTO AS
3 SELECT cp.IDPROG, cp.EAN, cp.QUANTITA, c.NOME, c.PREZZO
4 FROM COMPONENTIPROGETTO cp
5 JOIN COMPONENTE c ON cp.EAN = c.EAN;
```

2.5.4 Vista dipendente reparto

```
1 --Vista per visualizzare i dettagli di un dipendente insieme alle
  informazioni del reparto di appartenenza:
2 CREATE VIEW VISTA_DIPENDENTE_REPARTO AS
3 SELECT d.IDDIPENDENTE, d.NOME, d.COGNOME, d.RAL, d.TIPODIPENDENTE,
  r.IDREPARTO, r.TELEFONO
4 FROM DIPENDENTE d
5 JOIN REPARTO r ON d.IDDIPENDENTE = r.IDREPARTO;
```

2.5.5 VISTA CALCOLO PRESENZE

```
1 --Vista per calcolare le presenze di ogni dipendente in ogni giorno
  per un mese nel rispettivo reparto
2 CREATE VIEW VISTA_PRESENZE AS
3 SELECT TURNO.IDDIPENDENTE, TURNO.IDREPARTO, TRUNC(TURNO.ORAINIZIO)
  AS DATA,
```

```

4         SUM(EXTRACT(HOUR FROM TURNO.ORAFINE - TURNO.ORAINIZIO)) AS
        ORE_PRESENZA
5 FROM TURNO
6 WHERE TURNO.ORAFINE - TURNO.ORAINIZIO >= INTERVAL '4' HOUR
7        AND TURNO.ORAINIZIO >= TRUNC(SYSDATE, 'MM') -- Inizio mese
        corrente
8        AND TURNO.ORAINIZIO < TRUNC(SYSDATE, 'MM') + INTERVAL '1'
        MONTH -- Fine mese corrente
9 GROUP BY TURNO.IDDIPENDENTE, TURNO.IDREPARTO, TRUNC(TURNO.ORAINIZIO
        );

```

2.5.6 VISTA PER CALCOLARE LO SPAZIO IN MAGAZZINO

```

1        --Vista per calcolare lo spazio totale e lo spazio occupato in
        ogni magazzino
2 CREATE VIEW VISTA_MAGAZZINO_PIENO AS
3 SELECT M.IDREPARTO, M.SPAZIOTOT, SUM(I.QUANTITA) AS SPAZIO_OCCUPATO
        ,
4        CASE WHEN SUM(I.QUANTITA) >= M.SPAZIOTOT THEN 'PIENO' ELSE '
        NON PIENO' END AS STATO_MAGAZZINO
5 FROM MAGAZZINO M
6 LEFT JOIN IMMAGAZZINAMENTO I ON M.IDREPARTO = I.IDREPARTO
7 GROUP BY M.IDREPARTO, M.SPAZIOTOT;

```

2.6 Data Control Language

In questo paragrafo creiamo gli utenti che possono accedere al database.

```
1  --Creazione utenti
2  CREATE USER proprietario_db_fabbrica IDENTIFIED BY admin;
3
4  CREATE USER responsabile_DB IDENTIFIED BY pass_responsabile;
5  GRANT CONNECT, CREATE SESSION TO responsabile_DB;
6  GRANT UPDATE ON progettazione TO responsabile_DB;
7  GRANT UPDATE ON produzione TO responsabile_DB;
8  GRANT EXECUTE ON STIMA_CONSEGNA_PROGETTO TO responsabile_DB;
9
10
11 CREATE USER cliente_DB IDENTIFIED BY pass_cliente;
12 GRANT CONNECT, CREATE SESSION TO cliente_DB;
13 GRANT SELECT ON cliente TO cliente_DB;
14 GRANT SELECT ON progetto TO cliente_DB;
15 GRANT INSERT ON cliente TO cliente_DB;
16 GRANT INSERT ON ordine TO cliente_DB;
17 GRANT UPDATE ON ordine TO cliente_DB;
18 GRANT EXECUTE ON ordina_progetto TO cliente_DB;
19 GRANT EXECUTE ON cancella_progetto_ordine TO cliente_DB;
20
21
22 CREATE USER fornitore_DB IDENTIFIED BY pass_fornitore;
23 GRANT CONNECT, CREATE SESSION TO fornitore_DB;
24 GRANT SELECT ON fornitore TO fornitore_DB;
25 GRANT SELECT ON componente TO fornitore_DB;
26 GRANT INSERT ON forniture TO fornitore_DB;
27 GRANT EXECUTE ON inserisci_forniture TO fornitore_DB;
```

2.7 Scheduler

```
1  BEGIN
2  DBMS_SCHEDULER.CREATE_JOB(
3      job_name          => 'CANCELLA_ORDINE',
4      job_type          => 'PLSQL_BLOCK',
5      job_action        => 'BEGIN
6                          DELETE FROM ORDINE
7                          WHERE IDORDINE = :order_id;
8                          END;',
9      start_date        => SYSTIMESTAMP,
10     repeat_interval    => 'freq=hourly; interval=5',
11     enabled            => TRUE,
12     auto_drop          => TRUE
13 );
14 END;
15 /
16 /*DBMS_SCHEDULER.CREATE_JOB: Questa procedura crea un nuovo job
17 scheduler con i parametri specificati.
18 job_name: Specifica il nome del job. In questo caso, il nome del
19 job      "CANCELLA_ORDINE".
20 job_type: Specifica il tipo di job come "PLSQL_BLOCK". Indica che
21 il job eseguir un blocco di codice PL/SQL.
22 job_action: Questa l'azione del job, ovvero il blocco di codice
23 PL/SQL da eseguire. Nel caso specifico,
```

```
20  il blocco di codice esegue una query di eliminazione che cancella l
    'ordine dalla tabella ORDINE in base
21  all'ID dell'ordine fornito come parametro :order_id. Nota che :
    order_id      un segnaposto che dovrebbe essere
22  sostituito con il valore effettivo dell'ID dell'ordine da
    cancellare.
23  start_date: Specifica la data di inizio del job. In questo caso,
    viene utilizzata la funzione SYSTIMESTAMP
24  per impostare la data di inizio al momento corrente.
25  repeat_interval: Specifica l'intervallo di ripetizione del job. In
    questo caso, l'intervallo      impostato su
26  "freq=hourly; interval=5", il che significa che il job verr
    eseguito ogni 5 ore.
27  enabled: Specifica se il job      abilitato o meno. Nel caso
    specifico,      impostato su TRUE, quindi il job sar
28  abilitato dopo la sua creazione.
29  auto_drop: Specifica se il job verr      eliminato automaticamente
    dopo l'esecuzione. In questo caso,
30  impostato su TRUE, quindi il job verr      eliminato
    automaticamente dopo aver eseguito la cancellazione dell'ordine.
    */
```
