



# **Progetto Basi di Dati**

## **Gestione forniture Asl**

# Indice

<b>Progettazione</b>	<b>4</b>
<b>Diagramma EE/R</b>	<b>6</b>
<b>Diagramma relazionale</b>	<b>7</b>
<b>Utenti e le loro categorie</b>	<b>8</b>
<b>OPERAZIONI DEGLI UTENTI</b>	<b>9</b>
<b>VOLUMI</b>	<b>10</b>
<b>VINCOLO INTEGRITÀ</b>	<b>11</b>
<b>VERIFICA DI NORMALITÀ</b>	<b>13</b>
	<b>14</b>
<b>IMPLEMENTAZIONE</b>	<b>14</b>
<b>DATA E DEFINITION LANGUAGE</b>	<b>15</b>
<b>Data Manipulation Language</b>	<b>19</b>
<b>Trigger</b>	<b>37</b>
<b>Funzioni e Procedure</b>	<b>44</b>
<b>VISTE</b>	<b>57</b>
<b>Data Control Language</b>	<b>59</b>
	<b>58</b>
	<b>60</b>

Alessandro Massadoro: 0124002450  
Riccardo Andrea Spinosa: 0124002253

**Anno accademico:** 2021/2022

**Categoria:** Gestionale

Progettazione

Gli studenti si recano alla struttura il Frullone in Asl Napoli 1 per discutere le esigenze relative alla gestione del database e allo sviluppo del software a scopo didattico e prezzo non remunerativo non oggetto di esame.

Di tale colloquio emerge dalle indagini e domande poste dagli studenti che l'asl si suddivide in **distretti**, ogni distretto identificato da un codice e composto da varie strutture.

Una **struttura** stipula contratti di fornitura (luce, gas, acqua) riceve N **bollette**.

Ogni bolletta verrà identificata dal numero della fattura. All'interno della bolletta sono presenti i dati necessari dell'asl per il pagamento della bolletta stessa e il controllo dei consumi. Ad ogni contratto è emesso da un **fornitore**, che può rifornire la struttura di gas, acqua o corrente elettrica.

Il pagamento delle bollette all'ASL si gestisce tramite l'emissione di un **mandato** identificato da un codice, ogni mandato può pagare diverse bollette ed è necessariamente redatto da un **dipendente**, il mandato viene poi inviato ad un **ufficio** che gestisce i pagamenti e si occupa della vera e propria parte burocratica.

Sotto richiesta degli studenti vengono rilasciati i documenti per poter eseguire maggiori approfondimenti(allegati).

Date le richieste **non** è possibile eseguire **cancellazioni** dei dati inseriti in quanto è richiesta una traccia indelebile dei dati, sarà possibile solo effettuare **aggiornamenti** ad alcuni campi nelle occasioni in cui è necessario.

Inoltre un contatore può avere un solo contratto attivo. E' comunque importante conservare i contratti non più attivi e poter risalire a quale struttura erano associati in passato.

Lo stesso vale per i dipendenti che emettono un mandato, anche nel caso il dipendente non dovesse più lavorare presso l'ASL si dovrà poter identificare il mandatario.

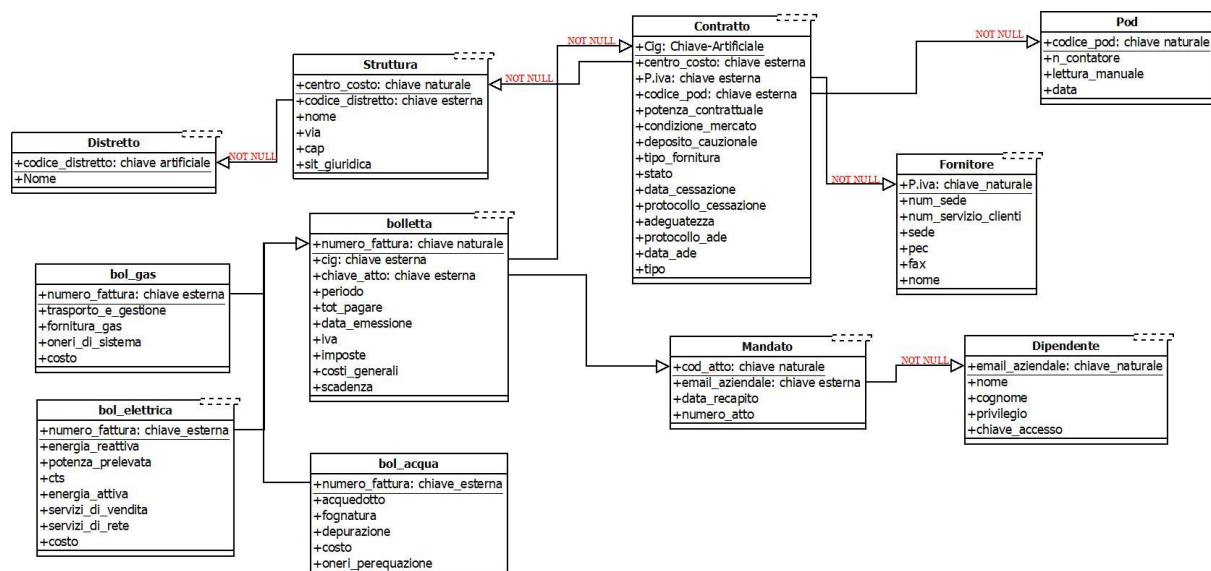
## GLOSSARIO

TERMINE	DEFINIZIONE	SINONIMI
Mandato	Atto utilizzato per il pagamento delle bollette, sottoscritto da un dipendente	
Fornitore	Azienda che fornisce energia, nel nostro caso acqua gas o corrente elettrica.	
Distretto	Gruppo di strutture, della stessa zona	



Una struttura ha necessariamente un contratto in quanto il nostro scopo è gestire le forniture. Per ogni mandato esiste necessariamente almeno una bolletta, un dipendente che lo emette e un ufficio a cui è recapitato. Consideriamo tutte queste considerazioni come **vincoli** che impongono quindi le **totalità** all'interno del nostro schema EE/R. Il costo della bollette è specializzato nonostante sia presente in tutte le bollette in quanto vengono espresse in maniera diversa, il consumo dell'elettricità è espresso in kilowattora mentre per il gas e l'acqua abbiamo il metro cubo.

## Diagramma relazionale



Notiamo come nello schema UML abbiamo riportato le totalità che potevamo rappresentare tramite l'aggiunta del **NOT NULL**., Nella traduzione tra modello EE/R a UML perdiamo gli attributi derivabili che avevamo rappresentato.

## Utenti e le loro categorie

All'interno del nostro database vengono solo gestiti 3 tipi di utenti, solo 1 realmente operativo:

- **Il tecnico dipendente** il quale gestisce l'inserimento nei nuovi utenti lavoratori all'interno del database.
- **utente**, parliamo degli utenti che gestiscono l'inserimento, modifica e visualizzazione dei dati, ma non può modificare i permessi.
- **Licenziato**, questo tipo di utente serve per tenere traccia dei vecchi utenti, difatti questo utente non ha più il permessi perchè revocati(pensione, cambio\_lavoro, ecc)

UTENTE	TIPO	VOLUMI	PERMESSI
Tecnico Dipendente	Amministratore	1	ALL
			SELECT ON DISTRETTO UPDATE ON DISTRETTO INSERT ON DISTRETTO  SELECT ON STRUTTURA UPDATE ON STRUTTURA INSERT ON STRUTTURA

Dipendente	Comune	1	<p>SELECT ON BOLLETTA UPDATE ON BOLLETTA INSERT ON BOLLETTA</p> <p>SELECT ON BOLLETTA_GAS UPDATE ON BOLLETTA_GAS INSERT BOLLETTA_GAS</p> <p>SELECT ON BOLLETTA_ACQUA UPDATE ON BOLLETTA_ACQUA INSERT BOLLETTA_ACQUA</p> <p>SELECT ON BOLLETTA_ELETTRICA UPDATE ON BOLLETTA_ELETTRICA INSERT BOLLETTA_ELETTRICA</p> <p>SELECT ON CONTRATTO UPDATE ON CONTRATTO INSERT ON CONTRATTO</p> <p>SELECT ON MANDATO UPDATE ON MANDATO INSERT ON MANDATO</p> <p>SELECT FORNITORE UPDATE FORNITORE INSERT FORNITORE</p> <p>SELECT CONTATORE UPDATE CONTATORE INSERT CONTATORE</p> <p>SELECT (NOME, COGNOME, EMAIL) SELECT CALCOLO_IVA_TOTALE EXECUTE POTENZA_ANNUO SELECT CALCOLO_TOT_PAGARE SELECT CALCOLO_CONSUMO_ANNUO SELECT COSTO_TOTALE_PRECEDENTE EXECUTE PAGAMENTO_BOLLETTA EXECUTE UPDATE_ADEGUATEZZA EXECUTE UPDATE_CONTRATTO</p> <p>EXECUTE SPESE_TOT_BOLLETTE EXECUTE UPDATE_LETTURA_POD</p>
Licenziato	Nessuno	1	nessun permesso

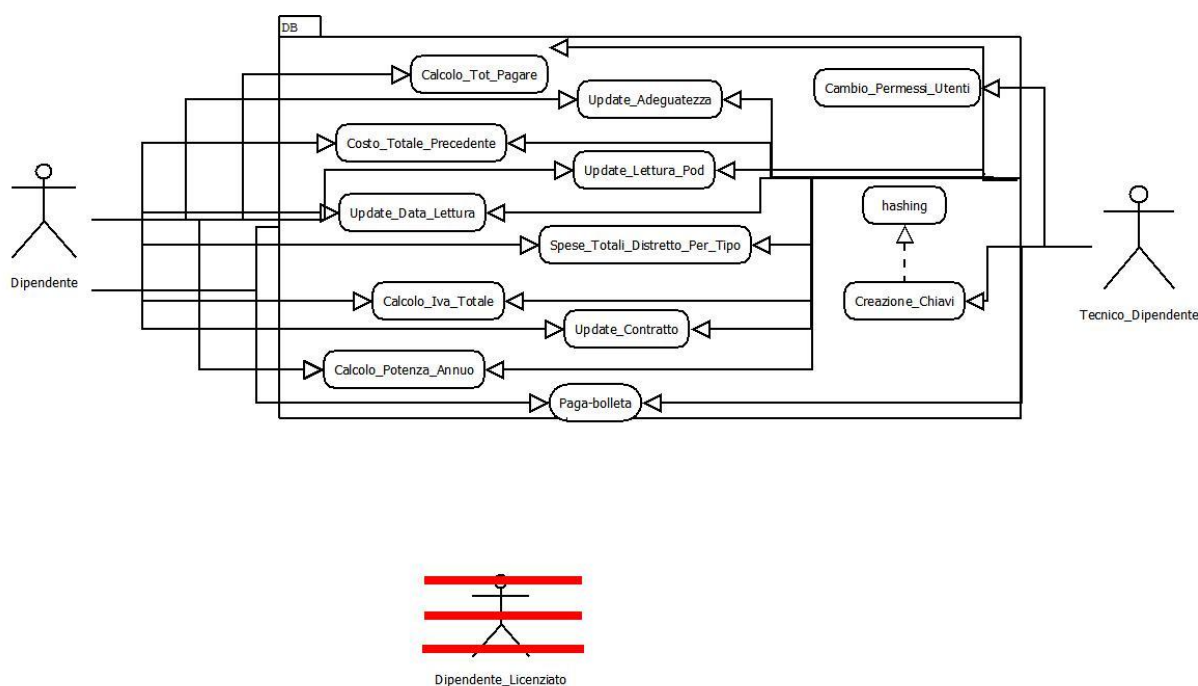


# OPERAZIONI DEGLI UTENTI

L'utente **Tecnico\_Dipendente** è colui che può dare accesso al database, oltre a creare account per gli utenti **dipendenti**, possiede tutti i poteri. Inoltre l'amministratore è fondamentale per rimuovere i permessi di accesso al database, per quei dipendenti che non si occuperanno più della gestione(trasferimenti,pensione,assegnazione di mansioni diverse) e per generare password sicure per i propri dipendenti. Ha poteri sulla creazione di tabelle, inserimento e update.

L'utente **dipendente** accede, al database e ha la possibilità di inserire i dati delle bollette, dei contratti, delle strutture e dei fornitori, inoltre ha la possibilità di creare mandati per il pagamento delle bollette, inoltre ha accesso alla visione dei dati inseriti.

L'utente **licenziato**, è l'utente che ha perso i permessi di accesso, quest'utente ha permessi rimossi dal tecnico\_dipendente che rimuove il suo accesso al database e avrà privilegio = 0 solo per tener traccia del suo non più lavoro sul database. **non sono state inserite le viste.**



## VOLUMI

Nella tavola dei volumi oltre a riportare il numero verosimile di tuple presenti in ciascuna tabella una volta che il DB risulta integrato, rappresenta anche il suo incremento atteso in un periodo di tempo prefissato. In quanto nel nostro database non sono consentite operazioni di eliminazione da parte dei dipendenti assumiamo che ci sia una crescita mensile o annua a seconda delle entità.

TABELLA	TIPO	VOLUME	INCREMENTO	PERIODO
DISTRETTO	E	10000	0	MAI
STRUTTURA	E	10000	0	ANNUALE
BOLLETTE	E	30000	1000	MENSILE
BOLLETTA_ELETTRICA	ES	10000	333	MENSILE
BOLLETTA_GAS	ES	10000	333	MENSILE
BOLLETTA_ACQUA	ES	10000	333	MENSILE
MANDATO	E	200	10	MENSILE
CONTRATTO	E	400	10	ANNO
FORNITORE	E	400	10	ANNO
DIPENDENTE	E	20	5	TRIENNALE
POD	E	400	8	BIENNALE

**Con E intendiamo le entità mentre ES sono le entità specializzate**

## VINCOLO INTEGRITÀ

Proponiamo l'elenco dei vincoli statici e dinamici presenti nel nostro database, evitiamo di riportare quelli ovvi, come vincoli di chiavi primarie e secondarie. **Le Bollette inserite saranno solo esclusivamente quelle Pagate.**

### Statici

- Il privilegio del dipendente può essere 0,1,2 dove:
  - 0 è l'utente senza permessi
  - 1 l'utente con permessi limitati
  - 2 l'utente con permessi massimi
 (trattato col check)
- Il tipo del contratto può essere E, G, A dove:
  - E è un contratto di elettricità

- G è un contratto del gas
- A è un contratto dell'acqua  
(trattato col check)
- L'adeguatezza all'interno del contratto può essere solo Sì, NO:
  - Sì, indica un impianto adeguato
  - No, indica un impianto non adeguato  
(trattato col check)
- Lo stato del contratto può essere solo 0,1 dove:
  - 0 è un contratto disdetto
  - 1 è un contratto attivo  
(trattato col ceck)  
(gestione modifica stato con procedura)  
(controllo di inserimento data cessazione con trigger)
- Nel caso l'impianto non risulti adeguato la data di adeguatezza e il protocollo di adeguatezza non devono essere inserite ed il cts presente in bolletta non dovrà essere pagato per tanto aggiunto e dunque verrà lasciato solo quello di default = 0.  
(regola di business)  
(Trattato con trigger)
- Se un contratto è di tipo energetico la bolletta associata sarà obbligatoriamente di energia, vale in egual modo nel caso fosse di gas o di acqua.  
(Trattato con trigger)
- se c'è un deposito cauzionale sarà maggiore di 0 altrimenti sarà lasciato quello di default = 0  
(trattato con default)
- Inserita una bolletta deve essere calcolato il costo.  
(trattato con trigger)
- Pagare la bolletta appena viene inserita ( risolta attraverso software).
- Quando viene effettuato un controllo tra il consumo del pod e della bolletta, se risultano diversi va stampato un messaggio in cui si avvisa che la bolletta ha consumo diverso.  
(trattato con procedura)
- La chiave dipendente va creata tramite specifico algoritmo e deve contenere almeno 16 cifre.  
(Trattato con funzione)

## Dinamici

- Se un contratto viene cessato deve essere aggiunta data cessazione e il numero protocollo con cui è stato cessato.  
(Trattato con procedura)
- Quando un contratto da non adeguato diventa adeguato è obbligatorio rimuovere il cts e di inserire la data di adeguatezza e numero protocollo con cui è stato adeguato.  
(trattato con procedura)
- solo un contratto può risultare attivo con lo stesso pod. Se un contratto ha lo stesso pod obbligatoriamente deve trattarsi di un contratto inattivo.  
(trattato con trigger)
- Ad ogni inserimento della bolletta controllare se la potenza prelevata è superiore alla potenza contrattuale, se verificato implica che c'è un dispendio di energia superiore a quello previsto, stampare un messaggio che avvisi di tale condizione.  
(trattato con trigger)

## VERIFICA DI NORMALITÀ

Nella verifica di normalità si combatte la ridondanza attraverso la caccia alle dipendenze funzionali anomale. Mentre il rispetto della prima forma normale è in qualche modo dato per scontato, implicito nel modello relazionale, la seconda e la terza vanno verificate analizzando il significato degli attributi, poiché si tratta di una proprietà intensionale, non estensionale. Non è obbligatorio portare il database in terza forma normale ed è opportuno tenere presente che ci sono casi in cui è preferibile per ragioni di prestazioni tenere una forma normale più bassa, in ogni caso le scelte operate devono essere motivate.

### **Prima forma normale**

Il campo data è un tipo primitivo nella maggior parte dei linguaggi di programmazione, e pur essendo in realtà strutturato è convenzionalmente considerato atomico. Vale lo stesso per il campo data e ora di Oracle DBMS. Nel caso dell'entità `pod`, nel codice verrà affrontato come unico attributo, tuttavia essendo la data inserita inerente alla data relativa al consumo letto l'ora non è necessaria e oracle ne setterà una sua di default. Un'entità è nella prima forma normale se non contiene gruppi ripetuti. In termini relazionali, una tabella è nella prima forma normale se non contiene colonne ripetute. Le colonne ripetute rendono i dati meno flessibili, sprecono spazio su disco e rendono più difficile la ricerca dei dati. Nel nostro caso abbiamo evitato ridondanza dei dati, per esempio nel contratto abbiamo creato una nuova entità fornitore così che i suoi dati vengano trascritti una sola volta. Abbiamo rispettato la prima forma normale gestendo i valori atomici ed evitando nel nostro schema ridondanza dei dati.

### **Seconda forma normale**

Tale forma risulta rispettata, Nel nostro schema non risultano chiavi multi attributo e risulta una chiave naturale per ogni entità. Non abbiamo utilizzato chiavi artificiali.

### **Terza forma normale**

una relazione si dice in terza forma normale quando è innanzitutto in seconda forma normale e le colonne della tabelle sono dipendenti non transitivamente dalla chiave primaria ossia non esistono attributi che dipendono da altri attributi non-chiave. Nel nostro caso quest'ultima non è rispettata perchè risultano dipendenze non dalla chiave primaria, per esempio la data cessazione dipende dallo stato del contratto.

### **FORMA NORMALE DI BOYCE E CODD**

Una relazione R è in forma normale di Boyce e Codd se e solo se è in terza forma normale e, per ogni dipendenza funzionale non banale, X è una superchiave per R. Quindi, dato un insieme di relazioni, non è possibile garantire sempre il raggiungimento di questa forma normale; in particolare il mancato raggiungimento di questo obiettivo è indice che la base dati è affetta da un'anomalia di cancellazione (ossia è possibile perdere dati a seguito di un'operazione di cancellazione). Tale condizione non è rispettata in quanto non risulta la terza forma normale.

## **IMPLEMENTAZIONE**

### **Creazione degli utenti**

Il primo passo da compiere per poter gestire il DBMS è accedere al DBMS come amministratore di sistema e creare l'utente proprietario della base di dati. E' possibile creare contestualmente anche altri utenti.

```
CREATE USER c##tecnico_dipendente IDENTIFIED BY tecnico;  
CREATE USER c##dipendente IDENTIFIED BY user;  
GRANT ALL PRIVILEGES TO tecnico_dipendente
```

in seguito occorre disconnettersi e riconnettersi con le credenziali dell'utente tecnico\_dipendente, che è l'amministratore della nuova base di dati e che avrà i permessi per creare tutti gli oggetti che occorrono.

## DATA E DEFINITION LANGUAGE

Il DDL riflette pedissequamente lo schema relazionale, le nove tabelle sono create mediante altrettante istruzioni di CREATE TABLE che includono tutti i vincoli di integrità esprimibili nel modello relazionale . Va sottolineato che in Oracle il tipo DATE include l'ora, e dunque i campi di tipo data e ora (con tutte le loro varianti come il campo ora, pur se separati nel diagramma E/R, sono tradotti con un campo unico di tipo DATE.

```
DROP TABLE DISTRETTO cascade constraints;  
DROP TABLE STRUTTURA cascade constraints;  
DROP TABLE POD cascade constraints;  
DROP TABLE FORNITORE cascade constraints;  
DROP TABLE DIPENDENTE cascade constraints;  
DROP TABLE CONTRATTO cascade constraints;  
DROP TABLE MANDATO cascade constraints;  
DROP TABLE BOLLETTA cascade constraints;  
DROP TABLE BOLLETTA_GAS cascade constraints;  
DROP TABLE BOLLETTA_ELETTRICA cascade constraints;  
DROP TABLE BOLLETTA_ACQUA cascade constraints;
```

```
CREATE TABLE DISTRETTO(  
  CODICE_DISTRETTO CHAR(6) PRIMARY KEY,  
  NOME_DISTRETTO VARCHAR(40) NOT NULL  
);  
CREATE TABLE STRUTTURA(  
  CENTRO_COSTO CHAR(10) PRIMARY KEY,  
  NOME_STRUTTURA VARCHAR(50) NOT NULL,  
  VIA VARCHAR(50) NOT NULL,  
  CAP CHAR(5),  
  SITUAZIONE_GIURIDICA CHAR(20),  
  CODICE_DISTRETTO CHAR(6) NOT NULL,  
  CONSTRAINT FK_DISTRETTO FOREIGN KEY (CODICE_DISTRETTO) REFERENCES
```

```

DISTRETTO(CODICE_DISTRETTO)
);
CREATE TABLE POD(
CODICE_POD VARCHAR(10) PRIMARY KEY,
LETTURA_MANUALE NUMBER(11,3) DEFAULT NULL,
DATA_LETTURA DATE DEFAULT NULL,
NUMERO_CONTATORE VARCHAR(12)
);
CREATE TABLE FORNITORE (
PARTITA_IVA VARCHAR(13) PRIMARY KEY,
NUMERO_SERVIZIO_CLIENTI VARCHAR(10),
NUMERO_SEDE VARCHAR (11),
INDIRIZZO VARCHAR(50) NOT NULL,
PEC VARCHAR(50),
FAX VARCHAR(18),
NOME VARCHAR(25) NOT NULL
);
CREATE TABLE CONTRATTO (
CIG CHAR(10) PRIMARY KEY,
POTENZA_CONTRATTUALE NUMBER(16,4) NOT NULL,
CONDIZIONE_MERCATO VARCHAR(40) NOT NULL,
DEPOSITO_CAUZIONALE NUMBER(9,2) DEFAULT 0,
TIPO_FORNITURA VARCHAR(30) NOT NULL,
STATO CHAR(1) NOT NULL,
DATA_CESSAZIONE DATE DEFAULT NULL,
PROTOCOLLO_CESSAZIONE VARCHAR(12) DEFAULT NULL,
ADEGUATEZZA CHAR(2) NOT NULL,
PROTOCOLLO_ADEGUATEZZA VARCHAR(12),
DATA_ADEGUATEZZA DATE ,
TIPO CHAR(1),
CENTRO_COSTO CHAR(10) NOT NULL,
PARTITA_IVA VARCHAR(13) NOT NULL,
CODICE_POD VARCHAR(10) NOT NULL,
CONSTRAINT CHECK_STATO CHECK(STATO IN ('0','1')),
CONSTRAINT CHECK_ENERGIA CHECK(TIPO IN ('G','E','A')),
CONSTRAINT CHECK_ADEGUATEZZA CHECK(ADEGUATEZZA IN ('SI','NO')),
CONSTRAINT FK_STRUTTURA FOREIGN KEY(CENTRO_COSTO) REFERENCES
STRUTTURA(CENTRO_COSTO),
CONSTRAINT FK_FORNITORE FOREIGN KEY(PARTITA_IVA) REFERENCES
FORNITORE(PARTITA_IVA),
CONSTRAINT FK_POD FOREIGN KEY(CODICE_POD) REFERENCES
POD(CODICE_POD) ON DELETE SET NULL
);
CREATE TABLE DIPENDENTE (
EMAIL_AZIENDALE VARCHAR(50) PRIMARY KEY,
NOME VARCHAR(20) NOT NULL,
COGNOME VARCHAR(20) NOT NULL,
PRIVILEGIO CHAR(1) NOT NULL,

```

```
CHIAVE_ACCESSO CHAR(16) DEFAULT NULL,  
CONSTRAINT CHECK_PRIVILEGIO CHECK(PRIVILEGIO IN ('0','1','2'))  
);
```

```
CREATE TABLE MANDATO (  
CODICE_ATTO VARCHAR(8) PRIMARY KEY,  
DATA_RECAPITO DATE NOT NULL,  
NUMERO_ATTO VARCHAR(9),  
TOTALE_PAGAMENTO NUMBER (10,2),  
EMAIL_AZIENDALE VARCHAR(40) NOT NULL,  
CONSTRAINT FK_DIPENDENTE FOREIGN KEY(EMAIL_AZIENDALE) REFERENCES  
DIPENDENTE(EMAIL_AZIENDALE) ON DELETE CASCADE  
);
```

```
CREATE TABLE BOLLETTA(  
NUMERO_FATTURA VARCHAR(15) PRIMARY KEY,  
PERIODO VARCHAR(24) NOT NULL,  
TOT_PAGARE NUMBER (9,0) NOT NULL,  
DATA_EMISSIONE DATE NOT NULL,  
IVA NUMBER(7,2) NOT NULL,  
IMPOSTE NUMBER (6,2) NOT NULL,  
COSTI_GENERALI NUMBER (7,2) NOT NULL,  
SCADENZA DATE NOT NULL,  
CODICE_ATTO VARCHAR(8) DEFAULT NULL,  
CIG CHAR(10) NOT NULL,  
CONSTRAINT FK_MANDATO1 FOREIGN KEY(CODICE_ATTO) REFERENCES  
MANDATO(CODICE_ATTO),  
CONSTRAINT FK_CONTRATTO1 FOREIGN KEY(CIG) REFERENCES CONTRATTO(CIG)  
);
```

```
CREATE TABLE BOLLETTA_GAS(  
TRASPORTO_E_GESTIONE NUMBER (9,2) NOT NULL,  
FORNITURA_GAS NUMBER (9,2) NOT NULL,  
COSTO NUMBER(9,2) DEFAULT NULL,  
CONSUMO_PDR NUMBER(9,2) NOT NULL,  
ONERI_DI_SISTEMA NUMBER(9,2) NOT NULL,  
NUMERO_FATTURA VARCHAR(15) PRIMARY KEY,  
CONSTRAINT FK_BOLLETTA_GAS FOREIGN KEY(NUMERO_FATTURA) REFERENCES  
BOLLETTA(NUMERO_FATTURA)  
);
```

```
CREATE TABLE BOLLETTA_ACQUA(  
COSTO NUMBER(9,2) DEFAULT NULL,  
ACQUEDOTTO NUMBER (9,2) NOT NULL,  
FOGNATURA NUMBER (9,2) NOT NULL,  
DEPURAZIONE NUMBER(9,2) NOT NULL,  
CONSUMO_ACQUA NUMBER(9,2) NOT NULL,  
ONERI_PEREQUAZIONE NUMBER(9,2) NOT NULL,  
NUMERO_FATTURA VARCHAR(15) PRIMARY KEY NOT NULL,  
CONSTRAINT FK_BOLLETTA_ACQUA FOREIGN KEY(NUMERO_FATTURA)  
REFERENCES BOLLETTA(NUMERO_FATTURA)
```



```
);  
CREATE TABLE BOLLETTA_ELETTRICA(  
COSTO NUMBER(9,2) DEFAULT NULL,  
ENERGIA_REATTIVA NUMBER (9,2) NOT NULL,  
POTENZA_PRELEVATA NUMBER (9,2) NOT NULL,  
CTS NUMBER(9,2) DEFAULT 0,  
ENERGIA_ATTIVA NUMBER (9,2) NOT NULL,  
NUMERO_FATTURA VARCHAR(15) PRIMARY KEY NOT NULL,  
SERVIZI_RETE NUMBER(9,2),  
SERVIZI_DI_VENDITA NUMBER(9,2),  
CONSTRAINT FK_BOLLETTA_ELETTRICA FOREIGN KEY(NUMERO_FATTURA)  
REFERENCES BOLLETTA(NUMERO_FATTURA)  
);
```

## Data Manipulation Language

**Da notare che i dati aggiunti per il popolamento sono solo a puro scopo didattico e sono stati aggiunti quelli necessari ai test.**

```
INSERT INTO DIPENDENTE  
(EMAIL_AZIENDALE,NOME,COGNOME,PRIVILEGIO,CHIAVE_ACCESSE) VALUES
```

```

('mauriziospinosa@gmail.com','Maurizio','Spinosa',2,'QMWNEBRVTCYZ99P$');
INSERT INTO DIPENDENTE
(EMAIL_AZIENDALE,NOME,COGNOME,PRIVILEGIO,CHIAVE_ACCESSE) VALUES
('riccardoandreaspinosa@gmail.com','Riccardo
Andrea','Spinosa',1,'QOWNEBHVTTCYZ89P$');
INSERT INTO DIPENDENTE
(EMAIL_AZIENDALE,NOME,COGNOME,PRIVILEGIO,CHIAVE_ACCESSE) VALUES
('alessandromassadoro@live.it','Alessandro','Massadoro',1,'ALWNEBHVTTCYZ89P$');
INSERT INTO DIPENDENTE
(EMAIL_AZIENDALE,NOME,COGNOME,PRIVILEGIO,CHIAVE_ACCESSE) VALUES
('palloncinodolce@live.it','Alessandro','Dolce',0,'DOWNEBHVTTCYZ89P$');

```

```

INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('SJK4BZ','Distretto Sanitario 24');
INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('K28ACH','Distretto Sanitario 25');
INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('6C33FX','DISTRETTO SANITARIO 26');
INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('0RZBGZ','DISTRETTO SANITARIO 28');
INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('PYCBDK','DISTRETTO SANITARIO 29');
INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('0RZBGS','DISTRETTO SANITARIO 30');
INSERT INTO DISTRETTO(CODICE_DISTRETTO, NOME_DISTRETTO) VALUES
('7P4NCT','DISTRETTO SANITARIO 31');

```

```

INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2410505','RSA','Via
Vincenzo padula 1', '80123','PROPRIETA', 'SJK4BZ');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2410206','Loreto
Crispi','Via Michelangelo Schipa 9', '80122','PROPRIETA', 'SJK4BZ');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2410401','SEDE','Via
Chiatamone 33', '81040','PROPRIETA', 'SJK4BZ');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2610505','Fasce deboli
RSA','Via Adriano 117', '81055','PROPRIETA', '6C33FX');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2610602','Materno
infantile','Via Giorgio De Grassi ', '80126','PROPRIETA', '6C33FX');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2810507','RSA','Via
Emilio Scaglione 498', '80129','FITTO', '0RZBGZ');

```

```

INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2810206','SCIUTI','Via
San Gennaro ad Antignano 42', '80145','PROPRIETA', '0RZBGZ');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2910401','SAN
GENNARO','VIA S GENNARO POV 23A', '80136','PROPRIETA', 'PYCBDK');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES
('LDS2910206','Poliambulatorio','CORSO AMEDEO SAVOIA 220', '80136','PROPRIETA',
'PYCBDK');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2910505','RSA','Via
Fratelli Cervi ', '80131','FITTO', 'PYCBDK');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDSSA10401','ELENA D
AOSTA','VIA CAGNAZZI L EM 29', '80136','PROPRIETA', 'PYCBDK');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS3010504','ASSISTENZA
ANZIANI','Corso Secondigliano 209', '80144','FITTO', '0RZBGS');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS3010406','MEDICINA
LEGALE','Via Janfolla 444', '80145','FITTO', '0RZBGZ');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2510401','Sede
Distretto','Via degni', '80125','FITTO', 'K28ACH');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS2510602','FUTURO
MATERNO INFANTILE E GINECOLOGIA','Via Winspeare 6', '80125','PROPRIETA',
'K28ACH');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS3110401','Sede
Distretto','Via Vespucci 9', '80142','FITTO', '7P4NCT');
INSERT INTO STRUTTURA(CENTRO_COSTO, NOME_STRUTTURA, VIA, CAP,
SITUAZIONE_GIURIDICA, CODICE_DISTRETTO) VALUES ('LDS3110206','SS.
ANNUNZIATA','VIA EGIZIACA A FORCELLA n° 18', '80139','PROPRIETA', '7P4NCT');

```

```

INSERT INTO MANDATO
(CODICE_ATTO,NUMERO_ATTO,TOTALE_PAGAMENTO,DATA_RECAPITO,EMAIL_AZIE
NDALE) VALUES
('279203','03',1800000,'28/FEB/2022','mauriziospinosa@gmail.com');

```

```

INSERT INTO MANDATO
(CODICE_ATTO,NUMERO_ATTO,TOTALE_PAGAMENTO,DATA_RECAPITO,EMAIL_AZIE
NDALE) VALUES
('54647','04',1300000,'28/FEB/2022','mauriziospinosa@gmail.com');

```

```

INSERT INTO MANDATO

```

(CODICE\_ATTO,NUMERO\_ATTO,TOTALE\_PAGAMENTO,DATA\_RECAPITO,EMAIL\_AZIE  
NDALE) VALUES  
('83644','09',1100000,'28/FEB/2022','mauriziospinosa@gmail.com');

INSERT INTO MANDATO  
(CODICE\_ATTO,NUMERO\_ATTO,TOTALE\_PAGAMENTO,DATA\_RECAPITO,EMAIL\_AZIE  
NDALE) VALUES  
('278374','02',10000,'03/FEB/2022','alessandromassadoro@live.it');

INSERT INTO MANDATO  
(CODICE\_ATTO,NUMERO\_ATTO,TOTALE\_PAGAMENTO,DATA\_RECAPITO,EMAIL\_AZIE  
NDALE) VALUES  
('11240','09',1230000,'03/FEB/2022','riccardoandreaspinosa@gmail.com');  
INSERT INTO MANDATO  
(CODICE\_ATTO,NUMERO\_ATTO,TOTALE\_PAGAMENTO,DATA\_RECAPITO,EMAIL\_AZIE  
NDALE) VALUES  
('282636','05',1280000,'03/FEB/2022','palloncinodolce@live.it');

INSERT INTO FORNITORE(PARTITA\_IVA,NUMERO\_SERVIZIO\_CLIENTI,INDIRIZZO,  
PEC, NOME) VALUES ('IT06328131211','0542843189','Via Molino Rosso, 8 - 40026 Imola  
(BO)','clienti.MUI@gruppohera.it','HERA COMM S.p.A.');

INSERT INTO FORNITORE(PARTITA\_IVA,NUMERO\_SEDE,INDIRIZZO, PEC, NOME)  
VALUES ('06328131211','800104104','Corso Umberto I, 330/D,  
Marigliano','clienti@estraspa.it', 'Estra');

INSERT INTO  
FORNITORE(PARTITA\_IVA,NUMERO\_SERVIZIO\_CLIENTI,NUMERO\_SEDE,INDIRIZZO,  
PEC, NOME) VALUES ('07679350632','800884021','800009585','Via Ventaglieri 83, 80135  
Napoli','segreteria generale@abnapoli.telecompost.it','Acqua Bene Comune Napoli');

-- INSERISCO I POD A CUI VIENE ATTRIBUITA UN CONTRATTO

INSERT INTO POD (CODICE\_POD,LETTURA\_MANUALE,DATA\_LETTURA) VALUES  
('00186839',222300.36, '10/NOV/2020');

INSERT INTO POD (CODICE\_POD,LETTURA\_MANUALE,DATA\_LETTURA)  
VALUES('00068002', 1540.00, '03/FEB/2022');

INSERT INTO POD (CODICE\_POD) VALUES ('041284');

INSERT INTO POD (CODICE\_POD) VALUES ('054284');

INSERT INTO POD (CODICE\_POD) VALUES ('043458');

INSERT INTO POD (CODICE\_POD) VALUES ('095384');

INSERT INTO POD (CODICE\_POD) VALUES ('082784');

INSERT INTO POD (CODICE\_POD) VALUES ('055284');

INSERT INTO POD (CODICE\_POD) VALUES ('043484');

INSERT INTO POD (CODICE\_POD) VALUES ('054254');

INSERT INTO POD (CODICE\_POD) VALUES ('041454');

INSERT INTO POD (CODICE\_POD) VALUES ('052334');

INSERT INTO POD (CODICE\_POD) VALUES ('012384');

INSERT INTO POD (CODICE\_POD) VALUES ('112384');

INSERT INTO POD (CODICE\_POD) VALUES ('122384');

INSERT INTO POD (CODICE\_POD) VALUES ('132384');

```
INSERT INTO POD (CODICE_POD) VALUES ('142384');
INSERT INTO POD (CODICE_POD) VALUES ('142394');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('919198919',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','1','G','SI','03/FEB/2022','LDS2410505','06328131211','132384');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD,DATA_CESSAZIONE) VALUES
('919828919',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','G','SI','03/FEB/2022','LDS2910206','06328131211','132384','30/JUN/2018');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('919198920',124,'FORNITURA ULTIMA ISTANZA','19','BASSA Tensione
Trifase','1','A','SI','03/FEB/2022','LDS2410505','06328131211','00068002');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('0101010121',560,'FORNITURA ULTIMA ISTANZA','10','Media Tensione
Trifase','1','G','SI','03/FEB/2022','LDS2410206','06328131211','142384');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD,DATA_CESSAZIONE) VALUES
('0101010190',560,'FORNITURA ULTIMA ISTANZA','10','Media Tensione
Trifase','0','A','SI','03/FEB/2022','LDS2410206','06328131211','142384','13/OCT/2012');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, CENTRO_COSTO,
PARTITA_IVA, CODICE_POD) VALUES
('1010101010',600,'FORNITURA ULTIMA ISTANZA','10','Media Tensione
Trifase','1','E','NO','LDS2410401','IT06328131211','00186839');
```

```
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('0101010101',500,'FORNITURA ULTIMA ISTANZA','10','Bassa Tensione
Trifase','1','G','SI','03/FEB/2022','LDS2610505','07679350632','041284');
```

```
INSERT INTO CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,
DEPOSITO_CAUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA,
```

```

DATA_ADEGUATEZZA, CENTRO_COSTO, PARTITA_IVA, CODICE_POD)
VALUES ('1100110011',900,'FORNITURA ULTIMA ISTANZA','19','Bassa
TensioneTrifase','1','G','SI','03/FEB/2022','LDS2610602','07679350632','054284');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('1234567890',1200,'FORNITURA ULTIMA ISTANZA','19','ALTA Tensione
Trifase','1','E','SI','03/FEB/2022','LDS2810507','06328131211','043458');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE, CODICE_POD) VALUES
('0011001100',120,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','G','SI','03/FEB/2022','LDS2810206','06328131211','03/FEB/2022','054254');
INSERT INTO CONTRATTO(CIG,
POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_CAUZIONALE,
TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('1234567220',1200,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','1','E','SI','03/FEB/2022','LDS2810206','06328131211','054254');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('1110001110',120,'FORNITURA ULTIMA ISTANZA','19','ALTA Tensione
Trifase','1','E','SI','03/FEB/2022','LDS2910401','07679350632','095384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('1118801110',120,'FORNITURA ULTIMA ISTANZA','19','BASSA Tensione
Trifase','1','E','SI','03/FEB/2022','LDS2910401','07679350632','043484');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('1118807710',120,'FORNITURA ULTIMA ISTANZA','19','BASSA Tensione
Trifase','1','G','SI','03/FEB/2022','LDS2910401','07679350632','052334');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA,
DATA_ADEGUATEZZA,CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE,
CODICE_POD)
VALUES ('0011001110',120,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','G','SI','03/FEB/2022','LDS2810206','06328131211','20/MAR/2022','095384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C

```

```

AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('1919191919',120,'FORNITURA ULTIMA ISTANZA','19','ALTA Tensione
Trifase','1','G','SI','03/FEB/2022','LDS2910505','07679350632','055284');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE, CODICE_POD) VALUES
('9191919191',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','G','SI','03/FEB/2022','LDSSA10401','07679350632','03/FEB/2022','012384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('9191919161',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','1','G','SI','03/FEB/2022','LDS3010504','06328131211','142394');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE, CODICE_POD) VALUES
('9191980092',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','A','SI','03/FEB/2022','LDS3010406','06328131211','03/FEB/2022','055284');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('9123989193',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','1','A','SI','03/FEB/2022','LDS2510401','06328131211','112384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('9123229193',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','1','G','SI','03/FEB/2022','LDS2510401','06328131211','012384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, CODICE_POD) VALUES
('9191981194',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','1','A','SI','03/FEB/2022','LDS2510602','06328131211','122384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE, CODICE_POD) VALUES
('9191989123',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','G','SI','03/FEB/2022','LDS3110401','06328131211','03/FEB/2022','095384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C

```

```

AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE, CODICE_POD) VALUES
('9194569191',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','E','SI','03/FEB/2022','LDS3110206','06328131211','03/FEB/2022','082784');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE, CODICE_POD) VALUES
('5671989191',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','A','SI','03/FEB/2022','LDS3010504','06328131211','03/FEB/2022','142384');
INSERT INTO
CONTRATTO(CIG,POTENZA_CONTRATTUALE,CONDIZIONE_MERCATO,DEPOSITO_C
AUZIONALE, TIPO_FORNITURA, STATO, TIPO, ADEGUATEZZA, DATA_ADEGUATEZZA,
CENTRO_COSTO, PARTITA_IVA, DATA_CESSAZIONE,CODICE_POD) VALUES
('9191989191',124,'FORNITURA ULTIMA ISTANZA','19','MEDIA Tensione
Trifase','0','A','SI','03/FEB/2022','LDS3010504','06328131211','03/FEB/2022','132384');

```

```

INSERT INTO BOLLETTA (NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('8117799','JUL-AUG',42807.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279
203','919828919');
INSERT INTO BOLLETTA (NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('8307799','SEP-OCT',42697.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279
203','0101010190');
INSERT INTO BOLLETTA (NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('2307799','SEP-OCT',1607.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','2792
03','9191989123');
INSERT INTO BOLLETTA (NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('2309999','SEP-OCT',46607.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279
203','9123229193');
INSERT INTO BOLLETTA (NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('2309399','SEP-OCT',707.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','27920
3','0011001110');
INSERT INTO BOLLETTA (NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('2309347','SEP-OCT',207.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','27920
3','919198920');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('2309947','SEP-OCT',4607.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','2792
03','919198919');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,CIG)
VALUES('2309317','SEP-OCT',42601.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279

```



```

203','5671989191');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309388','SEP-OCT',42557.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279
203','9191980092');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,
CIG)
VALUES('2309344','OCT-NOV',45507.17,'03/FEB/2022',400,2302,73432,'03/FEB/2022','279
203','0101010121');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309348','SEP-OCT',22607.02,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279
203','1010101010');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,
CIG)
VALUES('2309349','NOV-DEC',42657.01,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','27
9203','0101010101');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309346','NOV-DEC',42657.01,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','27
9203','9123989193');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309350','SEP-OCT',22507.10,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','5464
7','1100110011');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309351','SEP-OCT',22507.10,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','5464
7','9191981194');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309355','FEB-MAR',72507.11,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','5464
7','1234567890');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309365','FEB-MAR',72507.11,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','5464
7','1234567220');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309352','FEB-MAR',14507.99,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','5464
7','0011001100');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309389','FEB-MAR',14507.99,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','5464
7','9194569191');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,

```

```

DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,
CIG)VALUES('2309353','MAY-JUN',2507.93,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','8
3644','1110001110');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309354','JUN-JUL',272507.83,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','836
44','1110001110');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309322','JUL-AUG',9507.53,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','27837
4','1919191919');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309357','JAN-FEB',8555.33,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','11240
','9191919161');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309358','MAR-APR',100000.24,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','112
40','1118807710');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309359','APR-MAY',9999.24,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','28263
6','5671989191');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CIG)
VALUES('2109361','MAY-JUN',16581.73,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','9191
919161');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CIG)
VALUES('2509363','JAN-FEB',8555.33,'03/FEB/2022',456.2,300,2302,'03/FEB/2022','56719
89191');

INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('23093333','OCT-NOV',826097.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','
279203','9191989191');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,
CIG)
VALUES('2322347','MAR-APR',72607.17,'03/FEB/2022',1456.2,460,2302,'03/FEB/2022','27
9203','1118801110');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309397','NOV-DEC',72607.17,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','27
9203','0101010121');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO, CIG)
VALUES('2309227','JUN-JUL',42607.23,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279

```

```
203','9191989191');
INSERT INTO BOLLETTA(NUMERO_FATTURA, PERIODO, TOT_PAGARE,
DATA_EMISSIONE, IVA, IMPOSTE, COSTI_GENERALI, SCADENZA, CODICE_ATTO,
CIG)
VALUES('2309907','JAN-FEB',42607.17,'03/FEB/2022',1456.2,400,2302,'03/FEB/2022','279
203','0101010121');
```

```
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS, ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2309348',900,8273);
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS, ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2309353',900,8273);
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS,
ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2309354',900,8273);
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS, ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2309389',900,8273);
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS, ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2322347',900,8273);
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS, ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2309355',900,8273);
INSERT INTO BOLLETTA_ELETTRICA(ENERGIA_REATTIVA, POTENZA_PRELEVATA,
CTS,
ENERGIA_ATTIVA, NUMERO_FATTURA, SERVIZI_RETE, SERVIZI_DI_VENDITA)
VALUES (300,567,9,24,'2309365',900,8273);
```

```
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'8307799');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309227');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'23093333');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309359');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309346');
```

```

INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309317');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2509363');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309388');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309347');
INSERT INTO BOLLETTA_ACQUA(ACQUEDOTTO, FOGNATURA, DEPURAZIONE,
CONSUMO_ACQUA, ONERI_PEREQUAZIONE, NUMERO_FATTURA) VALUES
(1300,567,766,8930,44,'2309351');

```

```

/*INSERT INTO MANDATO
(CODICE_ATTO,NUMERO_ATTO,TOTALE_PAGAMENTO,DATA_RECAPITO,EMAIL_AZIE
NDALE) VALUES
('282636','05',1280000,'03/FEB/2022','palloncinodolce@live.it');*/

```

```

INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE,
FORNITURA_GAS,CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA)
VALUES (1300,567,4000, 44,'2309999');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2109361');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309322');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309399');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309397');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309947');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309344');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,

```

```
44,'2309349');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309358');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA)
VALUES (1300,567,4000, 44,'2309350');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309352');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2307799');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES
(1300,567,4000,44,'8117799');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567,4000,
44,'2309357');
INSERT INTO BOLLETTA_GAS(TRASPORTO_E_GESTIONE, FORNITURA_GAS,
CONSUMO_PDR, ONERI_DI_SISTEMA, NUMERO_FATTURA) VALUES (1300,567, 4000,
44,'2309907');
```

# Trigger

I trigger DML sono un male necessario e sono utili principalmente per il controllo di vincoli dinamici in fase di immissione o aggiornamento dei dati. E' possibile cum grano salis usarli anche per implementare regole di business o per calcolare, generare o sovrascrivere il valore di alcuni campi. I trigger delle bollette specializzate acqua, gas, elettrica sono stati suddivisi per coerenza logica in due diversi trigger, trigger di calcolo costo che automatizzano il calcolo del costo dato dalla somma dei propri attributi ed i trigger di avviso i quali avvisano l'utente di errori di inserimento.

**I trigger trattati sono richieste reali.**

## **TRIGGER CALCOLO\_COSTO\_ELETTRICA**

### **COSTO CALCOLATO DALLA SOMMA DI ENERGIA\_ATTIVA + NEW.CTS+COSTI**

```
CREATE OR REPLACE TRIGGER CALCOLO_COSTO_ELETTRICO  
BEFORE INSERT ON BOLLETTA_ELETTRICA
```

```
FOR EACH ROW
```

```
DECLARE
```

```
COSTI NUMBER (7,2);
```

```
BEGIN
```

```
SELECT COSTI_GENERALI INTO COSTI FROM BOLLETTA WHERE  
NUMERO_FATTURA = :NEW.NUMERO_FATTURA;
```

```
:NEW.COSTO := :NEW.ENERGIA_ATTIVA + :NEW.CTS + COSTI;
```

```
END CALCOLO_COSTO_ELETTRICO;
```

## **TRIGGER\_CONTROLLO\_BOLLETTA\_ACQUA**

**Il Trigger controlla che le bollette di acqua vengano inserite effettivamente come bollette di acqua. Effettua tale controllo attraverso il tipo del contratto che ha generato la bolletta. Ciò consente di evitare errori di inserimento e di non andare incontro a dati corrotti. Da notare che è stato separato dal trigger del calcolo costo d'acqua solo ed esclusivamente per dividerli logicamente ma è più efficiente farne uno solo.**

```

CREATE OR REPLACE TRIGGER Controllo_Bolletta_Acqua
BEFORE INSERT ON BOLLETTA_ACQUA
FOR EACH ROW
DECLARE
CONTROLLO CHAR(1);
ECCE EXCEPTION;
BEGIN
SELECT TIPO INTO CONTROLLO
FROM CONTRATTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
WHERE :NEW.NUMERO_FATTURA = BOLLETTA.NUMERO_FATTURA;
IF CONTROLLO = 'G' OR CONTROLLO = 'E' THEN
RAISE ECCE;
END IF;
EXCEPTION
WHEN ECCE THEN RAISE_APPLICATION_ERROR(-20002,'Inserimento
Fallito');
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20002,'Inserimento Fallito');
END;

```

### **TRIGGER\_CONTROLLO\_BOLLETTA\_ELETTRICA**

**Il Trigger controlla che le bollette elettriche vengano inserite effettivamente come bollette elettriche. Effettua tale controllo attraverso il tipo del contratto che ha generato la bolletta. Ciò consente di evitare errori di inserimento e di non andare incontro a dati corrotti. Da notare che è stato separato dal trigger del calcolo costo elettrico solo ed esclusivamente per dividerli logicamente ma è più efficiente farne uno solo.**

```

CREATE OR REPLACE TRIGGER Controllo_Bolletta_ELETTRICA
BEFORE INSERT ON BOLLETTA_ELETTRICA
FOR EACH ROW
DECLARE
CONTROLLO CHAR(1);
ECCE EXCEPTION;

BEGIN
SELECT TIPO INTO CONTROLLO
FROM CONTRATTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
WHERE :NEW.NUMERO_FATTURA = BOLLETTA.NUMERO_FATTURA;

```

```

IF CONTROLLO = 'A' OR CONTROLLO = 'G' THEN
  RAISE ECCE;
END IF;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    RAISE_APPLICATION_ERROR(-20002,'Inserimento Fallito');
END;

```

### **TRIGGER\_AVVISO\_BOLLETTA\_ELETTRICA**

**Nel caso la bolletta risulti con potenza prelevata maggiore della potenza contrattuale implica che qualcuno possa usufruire gratuitamente a spese dei cittadini della fornitura. Il trigger avvisa di tale condizione così che possano essere effettuati i dovuti controlli. Dato che non è stato richiesto per acqua e gas perchè non di interesse e che infatti non è presente l'attributo potenza prelevata nella tabella acqua e gas tale trigger è stato gestito solo per bollette elettriche.**

```

CREATE OR REPLACE TRIGGER AVVISO
BEFORE INSERT ON BOLLETTA_ELETTRICA
FOR EACH ROW
DECLARE
  PIPPO NUMBER(9,2);
  PAPERINO NUMBER(16,4);
BEGIN

```

```

SELECT CONTRATTO.POTENZA_CONTRATTUALE INTO PAPERINO
FROM CONTRATTO JOIN BOLLETTA ON CONTRATTO.CIG =
BOLLETTA.CIG
WHERE :NEW.NUMERO_FATTURA = BOLLETTA.NUMERO_FATTURA;

```

```

IF :NEW.POTENZA_PRELEVATA > PAPERINO THEN
  DBMS_OUTPUT.PUT_LINE('ATTENZIONE POTENZA PRELEVATA
MAGGIORE DI POTENZA CONTRATTUALE!');
END IF;
END AVVISO;

```



## **TRIGGER\_CONTROLLO\_BOLLETTA\_GAS**

**Il Trigger controlla che le bollette elettriche vengano inserite effettivamente come bollette gas. Effettua tale controllo attraverso il tipo del contratto che ha generato la bolletta. Ciò consente di evitare errori di inserimento e di non andare incontro a dati corrotti. Da notare che è stato separato dal trigger del calcolo costo elettrico solo ed esclusivamente per dividerli logicamente ma è più efficiente farne uno solo.**

```
CREATE OR REPLACE TRIGGER Controllo_Bolletta_GAS
BEFORE INSERT ON Bolletta_GAS
FOR EACH ROW
DECLARE
    CONTROLLO CHAR(1);
    ECCE EXCEPTION;

BEGIN
    SELECT TIPO INTO CONTROLLO
    FROM CONTRATTO JOIN BOLLETTA ON CONTRATTO.CIG =
BOLLETTA.CIG
    WHERE :NEW.NUMERO_FATTURA = BOLLETTA.NUMERO_FATTURA;
    IF CONTROLLO = 'A' OR CONTROLLO = 'E' THEN
        RAISE ECCE;
    END IF;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20002,'Inserimento Fallito');
END;
```

## **TRIGGER\_CONTROLLO\_CONTRATTO**

**Il trigger di controllo contratto presenta 3 trigger uniti in uno solo e gestiscono vari aspetti dell'inserimento del contratto. Ciò ci consente di essere più efficienti.**

**Primo Trigger controlla che nel caso il contratto sia inserito come inattivo esso abbia una data di cessazione, in quanto se inattivo il contratto non può non avere una data di cessazione.**

**Secondo Trigger controlla l'adeguatezza, nel caso il contratto non sia adeguato il protocollo di adeguatezza non deve essere inserito, in questo caso però non generiamo errore in quanto ci basta settare a null la data.**

**Terzo ed ultimo Trigger avviene sulla quantità di pod che presentano un contratto attivo, di fatto se superiore ad 1 non può essere inserito un contratto attivo, quindi o è presente un errore nello stato, oppure deve prima essere disattivato il contratto già attivo tramite apposita procedura di update contratto.**

```
CREATE OR REPLACE TRIGGER CONTROLLO_DATA_CESSAZIONE
BEFORE INSERT ON CONTRATTO
FOR EACH ROW
DECLARE
    DATA_NO_INSERTITA EXCEPTION;
    ERRORE EXCEPTION;
    CONTROLLO NUMBER(1,0);
BEGIN
```

```
IF :new.STATO = '0' THEN
    IF :new.DATA_CESSAZIONE IS NULL THEN
        RAISE DATA_NO_INSERTITA;
    END IF;
END IF;
IF :new.ADEGUATEZZA = 'NO' THEN
    IF :new.PROTOCOLLO_ADEGUATEZZA IS NOT NULL OR
:new.DATA_ADEGUATEZZA IS NOT NULL THEN
        :new.PROTOCOLLO_ADEGUATEZZA := NULL;
        :new.DATA_ADEGUATEZZA := NULL;
    END IF;
END IF;
```

```
IF :NEW.STATO = '1' THEN
    SELECT COUNT(CONTRATTO.CODICE_POD) INTO CONTROLLO
```

```

FROM POD JOIN CONTRATTO ON CONTRATTO.CODICE_POD =
POD.CODICE_POD
WHERE POD.CODICE_POD = :NEW.CODICE_POD AND STATO = '1';
    IF CONTROLLO = 1 THEN
        RAISE ERRORE;
    END IF;
END IF;
EXCEPTION
WHEN DATA_NO_INSERTITA THEN
    RAISE_APPLICATION_ERROR(-20001,'Inserimento Fallito');
WHEN ERRORE THEN
    RAISE_APPLICATION_ERROR(-20006,'Inserimento fallito');
END CONTROLLO_DATA_CESSAZIONE;

```

## Funzioni e Procedure

Le procedure sono la parte più strettamente legata alla logica di business e all'automazione della Base di Dati. Se nel DDL e nel DML il problema principale è decidere la rappresentazione ottima dei dati e garantire rispetto dei vincoli di integrità su di essi definiti, in questa fase il problema è come sfruttare al meglio tale rappresentazione per automatizzarne la gestione e trarne il massimo profitto

**Le funzioni e procedure realizzate sono richieste reali.**

### **Funzione calcolo\_iva\_totale**

**Questa funzione prende in input l'anno e il distretto da cui calcolare l'iva totale, in output restituisce l'iva totale calcolata di un determinato distretto. Un distretto è un insieme di strutture, per tanto l'iva restituita sarà uguale alla somma totale di tutte le strutture appartenenti a quel specifico distretto. Sono state definite due eccezioni oltre alla query che calcola la potenza. La prima eccezione gestisce il caso venga inserita una data maggiore della data attuale, restituita dal sistema, che sarà massimo uguale alla data di inserimento. La seconda eccezione gestisce il caso in cui venga inserito un distretto non esistente.**

```

CREATE OR REPLACE FUNCTION CALCOLO_IVA_ANNUI (ANNO CHAR, DISTRETTO1
CHAR)

```

```

RETURN NUMBER IS
IVA_TOT NUMBER(8,2);
ANNO_FUTURO EXCEPTION;
DISTRETTO_INESISTENTE CHAR(6);
BEGIN
SELECT DISTRETTO.CODICE_DISTRETTO INTO DISTRETTO_INESISTENTE FROM
DISTRETTO WHERE DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

IF ANNO>TO_CHAR(SYSDATE,'YYYY') THEN
  RAISE ANNO_FUTURO;
END IF;
SELECT SUM(IVA) INTO IVA_TOT
FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =
DISTRETTO.CODICE_DISTRETTO
JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;
RETURN IVA_TOT;
EXCEPTION
WHEN ANNO_FUTURO THEN
  RAISE_APPLICATION_ERROR(-20010,'Anno errato');
  RETURN NULL;
WHEN NO_DATA_FOUND THEN
  RAISE_APPLICATION_ERROR(-20011,'DISTRETTO ERRATO');
  RETURN NULL;

END CALCOLO_IVA_ANNUO;

```

### **FUNZIONE CALCOLO\_POTENZA\_MEDIA\_ANNUO**

**La funzione calcola la potenza media prelevata annua di un distretto, e gestisce due eccezione, un anno futuro ed un distretto inserito inesistente.**

```

CREATE OR REPLACE FUNCTION CALCOLO_POTENZA_MEDIA_ANNUO (ANNO
CHAR, DISTRETTO1 CHAR)
RETURN NUMBER IS
POT_PREL NUMBER(9,2);
ANNO_FUTURO EXCEPTION;
DISTRETTO_INESISTENTE CHAR(6);
BEGIN
SELECT DISTRETTO.CODICE_DISTRETTO INTO DISTRETTO_INESISTENTE FROM
DISTRETTO WHERE DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

IF ANNO>TO_CHAR(SYSDATE,'YYYY') THEN
  RAISE ANNO_FUTURO;
END IF;

```

```

SELECT AVG(POTENZA_PRELEVATA) INTO POT_PREL FROM BOLLETTA_ELETTRICA
JOIN BOLLETTA ON
BOLLETTA.NUMERO_FATTURA=BOLLETTA_ELETTRICA.NUMERO_FATTURA
JOIN CONTRATTO ON CONTRATTO.CIG = BOLLETTA.CIG JOIN STRUTTURA ON
STRUTTURA.CENTRO_COSTO=CONTRATTO.CENTRO_COSTO
JOIN DISTRETTO ON
DISTRETTO.CODICE_DISTRETTO=STRUTTURA.CODICE_DISTRETTO WHERE
EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE) = ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;
RETURN POT_PREL;
EXCEPTION
WHEN ANNO_FUTURO THEN
  RAISE_APPLICATION_ERROR(-20010,'Anno errato');
  RETURN NULL;
WHEN NO_DATA_FOUND THEN
  RAISE_APPLICATION_ERROR(-20011,'DISTRETTO ERRATO');
  RETURN NULL;

END CALCOLO_POTENZA_MEDIA_ANNUO;

```

### **FUNZIONE POTENZA\_ANNUO**

**Questa funzione permette di calcolare la potenza kwh totale di un distretto.**

**La funzione Prende in ingresso l'anno e il distretto1, dove l'anno rappresenta l'anno in cui vogliamo la somma totale della potenza di un distretto, il distretto1 indica il distretto su cui vogliamo eseguire la funzione. Sono state definite due eccezioni oltre alla query che calcola la potenza. La prima eccezione gestisce il caso venga inserita una data maggiore della data attuale, restituita dal sistema, che sarà massimo uguale alla data di inserimento. La seconda eccezione gestisce il caso in cui venga inserito un distretto non esistente.**

```

CREATE OR REPLACE FUNCTION CALCOLO_POTENZA_ANNUO (ANNO CHAR,
DISTRETTO1 CHAR)
RETURN NUMBER IS
POTENZA_TOT NUMBER(8,2);
ANNO_FUTURO EXCEPTION;
DISTRETTO_INESISTENTE CHAR(6);
BEGIN
SELECT DISTRETTO.CODICE_DISTRETTO INTO DISTRETTO_INESISTENTE FROM
DISTRETTO WHERE DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

IF ANNO>TO_CHAR(SYSDATE,'YYYY') THEN
  RAISE ANNO_FUTURO;
END IF;

```

```

SELECT SUM(POTENZA_PRELEVATA) INTO POTENZA_TOT
FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =
DISTRETTO.CODICE_DISTRETTO
JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG JOIN BOLLETTA_ELETTRICA
ON BOLLETTA.NUMERO_FATTURA = BOLLETTA_ELETTRICA.NUMERO_FATTURA
WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;
RETURN POTENZA_TOT;
EXCEPTION
WHEN ANNO_FUTURO THEN
  RAISE_APPLICATION_ERROR(-20010,'ANNO ERRATO');
WHEN NO_DATA_FOUND THEN
  RAISE_APPLICATION_ERROR(-20011,'DISTRETTO NON ESISTENTE');
RETURN NULL;
END CALCOLO_POTENZA_ANNUO;

```

## **FUNZIONE CALCOLO\_TOT\_PAGARE**

**La funzione calcola la somma di tutte le bollette appartenenti al distretto senza differenziare il tipo. Prende in ingresso l'anno e il distretto e gestisce due diverse eccezioni oltre alla query che restituisce l'output desiderato. La prima eccezione gestisce il caso venga inserita una data maggiore della data attuale, restituita dal sistema, che sarà massimo uguale alla data di inserimento. La seconda eccezione gestisce il caso in cui venga inserito un distretto non esistente.**

```

CREATE OR REPLACE FUNCTION CALCOLO_TOT_PAGARE_ANNUO (ANNO CHAR,
DISTRETTO CHAR)
RETURN NUMBER IS
PAGARE_TOT NUMBER(8,2);
ANNO_FUTURO EXCEPTION;
DISTRETTO_INESISTENTE CHAR(6);
BEGIN
SELECT DISTRETTO.CODICE_DISTRETTO INTO DISTRETTO_INESISTENTE FROM
DISTRETTO WHERE DISTRETTO.CODICE_DISTRETTO = DISTRETTO;

IF ANNO>TO_CHAR(SYSDATE,'YYYY') THEN
  RAISE ANNO_FUTURO;
END IF;
SELECT SUM(TOT_PAGARE) INTO PAGARE_TOT
FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =

```

```

DISTRETTO.CODICE_DISTRETTO
JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO;
RETURN PAGARE_TOT;
EXCEPTION
WHEN ANNO_FUTURO THEN
RAISE_APPLICATION_ERROR(-20010,'Anno errato');
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20011,'DISTRETTO NON ESISTENTE');
RETURN NULL;
END CALCOLO_TOT_PAGARE_ANNUO;

```

### **FUNZIONE CALCOLO\_CONSUMO\_ANNUO**

**la funzione restituisce il calcolo dei consumi totali di tutte le bollette di un distretto di un determinato anno passato in ingresso. Prende in ingresso l'anno e il distretto e gestisce due diverse eccezioni oltre alla query che restituisce l'output desiderato. La prima eccezione gestisce il caso venga inserita una data maggiore della data attuale, restituita dal sistema, che sarà massimo uguale alla data di inserimento. La seconda eccezione gestisce il caso in cui venga inserito un distretto non esistente.**

```

CREATE OR REPLACE FUNCTION CALCOLO_CONSUMO_ANNUO (ANNO CHAR,
DISTRETTO1 CHAR)
RETURN NUMBER IS

```

```

CONSUMO_E NUMBER(8,2);
CONSUMO_A NUMBER(8,2);
CONSUMO_G NUMBER(8,2);
CONSUMO_TOT NUMBER (12,2);
ANNO_FUTURO EXCEPTION;
DISTRETTO_INESISTENTE CHAR(6);

```

```

BEGIN

```

```

--dalla select controlliamo se il distretto inserito esiste, se esiste andiamo avanti altrimenti
andiamo in eccezione in quanto il distretto non esiste

```

```

SELECT DISTRETTO.CODICE_DISTRETTO INTO DISTRETTO_INESISTENTE FROM
DISTRETTO WHERE DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

```

```

--controlliamo se l'anno inserito è maggiore di quello attuale, in quel caso andiamo in
eccezione altrimenti continuiamo

```

```

IF ANNO>TO_CHAR(SYSDATE,'YYYY') THEN
RAISE ANNO_FUTURO;
END IF;

```

```

--prendiamo la somma del consumo totale di tutte le bollette, acqua gas ed elettriche.

```

```

SELECT SUM(COSTO) INTO CONSUMO_E
FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =
DISTRETTO.CODICE_DISTRETTO
JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
JOIN BOLLETTA_ELETTRICA ON BOLLETTA.NUMERO_FATTURA =
BOLLETTA_ELETTRICA.NUMERO_FATTURA
WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

```

```

SELECT SUM(COSTO) INTO CONSUMO_A
FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =
DISTRETTO.CODICE_DISTRETTO
JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
JOIN BOLLETTA_ACQUA ON BOLLETTA.NUMERO_FATTURA =
BOLLETTA_ACQUA.NUMERO_FATTURA
WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

```

```

SELECT SUM(COSTO) INTO CONSUMO_G
FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =
DISTRETTO.CODICE_DISTRETTO
JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
JOIN BOLLETTA_GAS ON BOLLETTA.NUMERO_FATTURA =
BOLLETTA_GAS.NUMERO_FATTURA
WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= ANNO AND
DISTRETTO.CODICE_DISTRETTO = DISTRETTO1;

```

```

CONSUMO_TOT := CONSUMO_G +CONSUMO_A+CONSUMO_E;
RETURN CONSUMO_TOT;
EXCEPTION
WHEN ANNO_FUTURO THEN
RAISE_APPLICATION_ERROR(-20010,'ANNO ERRATO');
RETURN NULL;
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR (-20011,'DISTRETTO INESISTENTE');
RETURN NULL;
END CALCOLO_CONSUMO_ANNUO;

```

## **FUNZIONE TOT\_PAGARE\_PRECEDENTE**

**la funzione prende in ingresso il numero fattura della bolletta e restituisce il totale**



**pagamento arrivato dall'ultima bolletta prima dell'attuale nella stessa struttura. Gestisce una sola eccezione nel caso venga passata in ingresso una bolletta non esistente.**

```
CREATE OR REPLACE FUNCTION TOT_PAGARE_PRECEDENTE(N_FATT VARCHAR)
RETURN NUMBER IS
SCOOPYDOO VARCHAR(15);
MINNIE NUMBER (9,2);
RONNIE DATE;
D_BOLLETTA DATE;
BEGIN
SELECT (BOLLETTA.NUMERO_FATTURA) INTO SCOOPYDOO FROM BOLLETTA
WHERE BOLLETTA.NUMERO_FATTURA=N_FATT;

SELECT (DATA_EMISSIONE) INTO D_BOLLETTA FROM BOLLETTA WHERE
BOLLETTA.NUMERO_FATTURA=N_FATT;
SELECT TOT_PAGARE INTO MINNIE FROM BOLLETTA WHERE DATA_EMISSIONE =
(SELECT MAX(DATA_EMISSIONE) FROM BOLLETTA WHERE DATA_EMISSIONE
<D_BOLLETTA);
RETURN MINNIE;

EXCEPTION
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20012,'FATTURA ERRATA');
RETURN NULL;
END TOT_PAGARE_PRECEDENTE;
```

#### **PROCEDURA UPDATE\_ADEGUATEZZA**

**Quando un contratto da non adeguato diventa adeguato la data di adeguatezza e numero protocollo con cui è stato adeguato.**

**il cts è settato =0 per i contratti adeguati**

**la procedura trasforma un contratto non adeguato in contratto adeguato. Sono gestite due eccezioni: una in caso contratto non esiste e l'altra nel caso la data inserita sia oltre il presente e risulti futura.**

```
CREATE OR REPLACE PROCEDURE UPDATE_ADEGUATEZZA(CON_CIG CHAR,
DATA_ADE DATE, PROT CHAR)
IS
CIG_CONT CHAR(10);
CIG_EXCEPT EXCEPTION;
DATA_FUTURA EXCEPTION;
BEGIN
SELECT CIG INTO CIG_CONT
FROM CONTRATTO WHERE CIG= CON_CIG;
```

```

IF DATA_ADE > SYSDATE THEN
RAISE DATA_FUTURA;
END IF;
UPDATE CONTRATTO SET ADEGUATEZZA = 'SI', DATA_ADEGUATEZZA= DATA_ADE,
PROTOCOLLO_ADEGUATEZZA = PROT
WHERE CIG = CON_CIG;
EXCEPTION
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20015,'CONTRATTO INESISTENTE');
WHEN DATA_FUTURA THEN
RAISE_APPLICATION_ERROR(-20010,'DATA_ERRATA');
END UPDATE_ADEGUATEZZA;

```

### **PROCEDURA UPDATE\_CONTRATTO**

**La procedura consente di settare un contratto attivo in inattivo e di conseguenza di settare la data di cessazione protocollo e il numero protocollo con cui e' stata cessata. Sono gestite due eccezioni: una in caso contratto non esiste e l'altra nel caso la data inserita sia oltre il presente e risulti futura.**

```

CREATE OR REPLACE PROCEDURE UPDATE_CONTRATTO(CON_CIG CHAR,
DATA_CES DATE, PROT CHAR)
IS
CIG_CONT CHAR(10);
DATA_FUTURA EXCEPTION;
ERRORE EXCEPTION;
ATTIVO CHAR(1);
BEGIN
--QUESTA QUERY SE IL CONTRATTO È NON ATTIVO RISULTA NO DATAFOUND DI
CONSEGUENZA RISULTA NON DISATTIVABILE IL CONTRATTO poichè già disattivo

```

```

SELECT CIG INTO CIG_CONT
FROM CONTRATTO WHERE CIG= CON_CIG;

```

```

IF DATA_CES > SYSDATE THEN
RAISE DATA_FUTURA;
END IF;
SELECT STATO INTO ATTIVO
FROM CONTRATTO
WHERE CIG = CON_CIG;
IF ATTIVO = '0' THEN
RAISE ERRORE;
END IF;
UPDATE CONTRATTO SET DATA_CESSAZIONE = DATA_CES, STATO= '0',
PROTOCOLLO_CESSAZIONE = PROT
WHERE CIG = CON_CIG;

```

```

EXCEPTION
WHEN NO_DATA_FOUND THEN
    RAISE_APPLICATION_ERROR(-20015,'CONTRATTO ERRATO');
WHEN DATA_FUTURA THEN
    RAISE_APPLICATION_ERROR(-20010,'DATA ERRATA');
WHEN ERRORE THEN
    RAISE_APPLICATION_ERROR(-20030,'CONTRATTO GIÀ INATTIVO');

END UPDATE_CONTRATTO;

```

### **PROCEDURA SPESE\_TOT\_BOLLETTE**

**Questa procedura consente di calcolare con un unica query le spese totali di un distretto suddivise in base al suo tipo, acqua,energia,gas. In questo caso il ciclo for è un male necessario in quanto restituiamo 2 colonne e 3 righe. Sono gestite due eccezioni: 1 per la data e l'altra nel caso il distretto inserito non risulti esistente.**

```

CREATE OR REPLACE PROCEDURE SPESE_TOT_BOLLETTE (DATAIN CHAR,
DISTRETTO1 CHAR)
IS
    TOTALE NUMBER;
    anno_futuro EXCEPTION;
    PIPPO CHAR (1);
    DISTRETTO_INESISTENTE CHAR(6);

    app bolletta.TOT_PAGARE%type;
    app1 CONTRATTO.TIPO%type;

    CURSOR c1
    IS (
        SELECT SUM(TOT_PAGARE) as Totale, CONTRATTO.TIPO as Tipo
        FROM DISTRETTO JOIN STRUTTURA ON STRUTTURA.CODICE_DISTRETTO =
DISTRETTO.CODICE_DISTRETTO
        JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
        JOIN BOLLETTA ON CONTRATTO.CIG = BOLLETTA.CIG
        WHERE EXTRACT(YEAR FROM BOLLETTA.DATA_EMISSIONE)= '2021' AND
DISTRETTO.CODICE_DISTRETTO ='SJK4BZ'
        GROUP BY CONTRATTO.TIPO);

```

BEGIN

--dalla select controlliamo se il distretto inserito esiste, se esiste andiamo avanti altrimenti andiamo in eccezione in quanto il distretto non esiste

SELECT DISTRETTO.CODICE\_DISTRETTO INTO DISTRETTO\_INESISTENTE FROM  
DISTRETTO WHERE DISTRETTO.CODICE\_DISTRETTO = DISTRETTO1;

--controllo sulla data se futura

IF DATAIN>TO\_CHAR(sysdate,'yyyy')THEN

RAISE anno\_futuro;

END IF;

OPEN c1;

FETCH c1 INTO app,app1;

WHILE c1%FOUND LOOP

DBMS\_OUTPUT.PUT\_LINE('Tipo = ' || app1 || ' Totale =' || app);

FETCH c1 INTO app,app1;

END LOOP;

-- dalla select, per avere chiaro quale fosse il consumo di chi abbiamo sia la somma sia il tipo di contratto in un unica tabella di ritorno

EXCEPTION

when anno\_futuro then

RAISE\_APPLICATION\_ERROR(-20010,'ANNO INESISTENTE');

WHEN NO\_DATA\_FOUND THEN

RAISE\_APPLICATION\_ERROR(-20011,'DISTRETTO INESISTENTE');

END SPESE\_TOT\_BOLLETTE;

## **PROCEDURA UPDATE\_LETTURA\_POD**

L'utente effettuerà l'update della lettura manuale con lo scopo di memorizzarla e confrontarla con bolletta. Le letture appartenenti allo stesso pod verranno sovrascritte con quelle di interesse. Lo scopo è controllare ogni qualvolta viene fatta una lettura manuale di una determinata data decisa da chi effettua tale lettura, di confrontarla con bolletta e controllare se risultino uguali. Ciò viene fatto per controllare se le bollette inviate siano coerenti ai consumi registrati dal contatore. Nel caso la lettura non corrisponde a quella della bolletta passata in ingresso la procedura stampa un messaggio che avvisa di tale condizione. La funzione prende in ingresso 4 parametri il pod, la bolletta, lettura del pod, e la data. Viene gestito il caso in

**cui bolletta o il pod non esistano.**

```
CREATE OR REPLACE PROCEDURE UPDATE_LETTURA_POD(DATA_LET DATE, POD1
CHAR, LETTURA NUMBER, BOLLETTA1 CHAR)
```

```
IS
```

```
CONTROLLO CHAR(1);
```

```
CON_NUM NUMBER(10,2);
```

```
ECC EXCEPTION;
```

```
BEGIN
```

```
    UPDATE POD
```

```
    SET DATA_LETTURA = DATA_LET, LETTURA_MANUALE = LETTURA
```

```
    WHERE CODICE_POD = POD1;
```

```
    SELECT TIPO INTO CONTROLLO
```

```
    FROM POD JOIN CONTRATTO ON CONTRATTO.CODICE_POD = POD.CODICE_POD
```

```
    WHERE POD.CODICE_POD = POD1 AND CONTRATTO.STATO = '1';
```

```
    IF CONTROLLO = 'A' THEN
```

```
        SELECT COSTO INTO CON_NUM
```

```
        FROM POD JOIN CONTRATTO ON CONTRATTO.CODICE_POD = POD.CODICE_POD
```

```
        JOIN BOLLETTA ON BOLLETTA.CIG = CONTRATTO.CIG
```

```
        JOIN BOLLETTA_ACQUA ON BOLLETTA.NUMERO_FATTURA =
```

```
BOLLETTA_ACQUA.NUMERO_FATTURA
```

```
        WHERE BOLLETTA.NUMERO_FATTURA = BOLLETTA1;
```

```
        IF CON_NUM != LETTURA THEN
```

```
            RAISE ECC;
```

```
        END IF;
```

```
    END IF;
```

```
    IF CONTROLLO = 'G' THEN
```

```
        SELECT COSTO INTO CON_NUM
```

```
        FROM POD JOIN CONTRATTO ON CONTRATTO.CODICE_POD =
```

```
POD.CODICE_POD
```

```
        JOIN BOLLETTA ON BOLLETTA.CIG = CONTRATTO.CIG
```

```
        JOIN BOLLETTA_GAS ON BOLLETTA.NUMERO_FATTURA =
```

```
BOLLETTA_GAS.NUMERO_FATTURA
```

```
        WHERE BOLLETTA.NUMERO_FATTURA = BOLLETTA1;
```

```
        IF CON_NUM != LETTURA THEN
```

```
            RAISE ECC;
```

```
        END IF;
```

```
    END IF;
```

```
    IF CONTROLLO = 'E' THEN
```

```
        SELECT COSTO INTO CON_NUM
```

```
        FROM POD JOIN CONTRATTO ON CONTRATTO.CODICE_POD =
```

```
POD.CODICE_POD
```

```

        JOIN BOLLETTA ON BOLLETTA.CIG = CONTRATTO.CIG
        JOIN BOLLETTA_ELETTRICA ON BOLLETTA.NUMERO_FATTURA =
BOLLETTA_ELETTRICA.NUMERO_FATTURA
        WHERE BOLLETTA.NUMERO_FATTURA = BOLLETTA1;
        IF CON_NUM != LETTURA THEN
            RAISE ECC;
        END IF;
    END IF;

EXCEPTION
WHEN ECC THEN
    DBMS_OUTPUT.PUT_LINE('La lettura inserita non corrisponde con quella della bolletta,
attenzione!');
WHEN NO_DATA_FOUND THEN

    RAISE_APPLICATION_ERROR(-20022, 'ERRORE');

END;
```

### **PROCEDURA CAMBIO\_PERMESSI**

**Questa procedura cambia i permessi dei dipendenti per distinguere e tenere traccia dei dipendenti che hanno ancora i permessi da chi invece risultano revocati. La procedura prende in ingresso due parametri e gestisce due diversi tipi di errore, uno nel caso venga inserita una email non esistente e l'altro nel caso venga assegnato un permesso non valido I PERMESSI EFFETTIVI VENGONO POI RIMOSSI TRAMITE IL REMOVE DEI PERMESSI.**

```

CREATE OR REPLACE PROCEDURE CAMBIO_PERMESSI(EMAIL CHAR, PER CHAR)
IS
    ERRORE EXCEPTION;
    CON_EMAIL CHAR(50);
BEGIN
    SELECT EMAIL_AZIENDALE INTO CON_EMAIL FROM DIPENDENTE WHERE
    EMAIL_AZIENDALE = EMAIL;

    IF (PER < '0') OR (PER > '3') THEN
        RAISE ERRORE;
    END IF;
    UPDATE DIPENDENTE
    SET PRIVILEGIO = PER
    WHERE EMAIL_AZIENDALE = EMAIL;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE_APPLICATION_ERROR(-20100,'EMAIL NON ESISTENTE');
    WHEN ERRORE THEN
        RAISE_APPLICATION_ERROR(-20120,'PERMESSO NON VALIDO');
```

END;

### **PROCEDURA CREAZIONE\_CHIAVI**

**Questa procedura è richiamata dal tecnico dipendente.**

**Come parametro d'ingresso ha l'email del dipendente di cui vuole generare la chiave, nel caso l'email non risulti corretta l'inserimento è annullato.**

**La procedura richiama al suo interno la funzione di hashing, la quale genera una chiave per l'utente casuale e di 16cifre. Nel caso si volesse generare una chiave più grande o più piccola bisogna cambiare i parametri della funzione richiamata all'interno della procedura.**

```
CREATE OR REPLACE PROCEDURE CREAZIONE_CHIAVI(EMAIL CHAR)
IS
PASS CHAR(16);
NOME_PR VARCHAR(20);
BEGIN
SELECT NOME INTO NOME_PR FROM DIPENDENTE WHERE EMAIL_AZIENDALE =
EMAIL; SELECT FUNZIONE_DI_HASHING(3,3,6,4) INTO PASS FROM DUAL;
UPDATE DIPENDENTE
SET CHIAVE_ACCESSO = PASS WHERE EMAIL_AZIENDALE = EMAIL;
EXCEPTION
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20100,'ERRORE');
END;
```

### **FUNZIONE DI HASHING**

**La funzione prende in ingresso quattro parametri per la creazione della password da assegnare all'utente, i parametri devono essere tutti di tipo numerico e definiscono la complessità della chiave stessa, una volta definito il tipo di chiave che si vuole anche con gli stessi parametri si otterrà sempre una configurazione di chiave diversa dalla precedente in quanto le combinazioni possibili sono enormi.**

```
create or replace FUNCTION FUNZIONE_DI_HASHING(
  nums    IN NUMBER,
  ch_sp   IN NUMBER,
  ch_min  IN NUMBER,
  ch_maiusc IN NUMBER) return varchar2
IS
  lun      number := nums + ch_sp + ch_min + ch_maiusc;
  pw      varchar2(200);
  v_iterations  number := 0;
  max_iter number := 500;
BEGIN
  loop
    pw := dbms_random.string('p',lun);
    v_iterations := v_iterations + 1;
```

```

        exit when (regexp_count(pw,'[a-z]') = ch_min
            and regexp_count(pw,'[A-Z]') = ch_maiusc
            and regexp_count(pw,'[0-9]') = nums)
            or v_iterations=max_iter;

end loop;

if v_iterations = max_iter THEN
    pw := '';
end if;

return(pw);
END;
```

## VISTE

Le viste trattate sono richieste reali.

- **VISTA MOSTRARE TUTTI I DIPENDENTI CHE NON LAVORANO PIU'**

```

CREATE VIEW UTENTI_LICENZIATI AS
SELECT Nome,Cognome
FROM Dipendente
WHERE privilegio = '0';
```

- **VISTA MOSTRARE TUTTI I DIPENDENTI ATTIVI**

```

CREATE VIEW UTENTI_ATTIVI AS
SELECT Nome,Cognome
FROM Dipendente
WHERE privilegio = '1';
```

- **VISTA MOSTRA TUTTI I DIPENDENTI CON MASSIMI PERMESSI**



```
CREATE VIEW SUPERUTENTI AS
SELECT Nome,Cognome
FROM Dipendente
WHERE privilegio = '2';
```

- **VISTA CHE MOSTRA LE INFORMAZIONI DELLE BOLLETTE ELETTRICHE DELLE STRUTTURE,**

```
CREATE OR REPLACE VIEW TABELLONE_STRUTTURA_LUCE AS
SELECT STRUTTURA.CENTRO_COSTO,
BOLLETTA.DATA_EMISSIONE,BOLLETTA.NUMERO_FATTURA,
BOLLETTA_ELETTRICA.COSTO, BOLLETTA_ELETTRICA.ENERGIA_ATTIVA,
BOLLETTA_ELETTRICA.ENERGIA_REATTIVA,
BOLLETTA_ELETTRICA.POTENZA_PRELEVATA, SERVIZI_DI_VENDITA,
SERVIZI_RETE, BOLLETTA.IMPOSTE, BOLLETTA.IVA
FROM STRUTTURA JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON BOLLETTA.CIG = CONTRATTO.CIG
JOIN BOLLETTA_ELETTRICA ON BOLLETTA_ELETTRICA.NUMERO_FATTURA =
BOLLETTA.NUMERO_FATTURA
```

- **VISTA CHE MOSTRA LE INFORMAZIONI DELLE BOLLETTE ACQUA DELLE STRUTTURE,**

```
CREATE OR REPLACE VIEW TABELLONE_STRUTTURA_ACQUA AS
SELECT STRUTTURA.CENTRO_COSTO,
BOLLETTA.DATA_EMISSIONE,BOLLETTA.NUMERO_FATTURA,
BOLLETTA_ACQUA.COSTO,
BOLLETTA_ACQUA.ACQUEDOTTO, BOLLETTA_ACQUA.FOGNATURA,
BOLLETTA_ACQUA.ONERI_PEREQUAZIONE, BOLLETTA.IMPOSTE,
BOLLETTA.IVA
FROM STRUTTURA JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON BOLLETTA.CIG = CONTRATTO.CIG
JOIN BOLLETTA_ACQUA ON BOLLETTA_ACQUA.NUMERO_FATTURA =
BOLLETTA.NUMERO_FATTURA
```

- **VISTA CHE MOSTRA LE INFORMAZIONI DELLE BOLLETTE GAS DELLE STRUTTURE,**

```
CREATE OR REPLACE VIEW TABELLONE_STRUTTURA_GAS AS
SELECT STRUTTURA.CENTRO_COSTO,
BOLLETTA.DATA_EMISSIONE,BOLLETTA.NUMERO_FATTURA,
BOLLETTA_GAS.COSTO,
BOLLETTA_GAS.TRASPORTO_E_GESTIONE,
BOLLETTA_GAS.FORNITURA_GAS,
BOLLETTA_GAS.ONERI_DI_SISTEMA, BOLLETTA.IMPOSTE, BOLLETTA.IVA
FROM STRUTTURA JOIN CONTRATTO ON CONTRATTO.CENTRO_COSTO =
STRUTTURA.CENTRO_COSTO
JOIN BOLLETTA ON BOLLETTA.CIG = CONTRATTO.CIG
JOIN BOLLETTA_GAS ON BOLLETTA_GAS.NUMERO_FATTURA =
BOLLETTA.NUMERO_FATTURA
```

## Data Control Language

**GRANT CONNECT, CREATE SESSION TO TECNICO;**

```
GRANT EXECUTE ON CAMBIO_PERMESSI TO TECNICO;
GRANT EXECUTE ON CREAZIONE_CHIAVI TO TECNICO;
GRANT SELECT ON HASHING TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA_ELETTRICA TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA_ACQUA TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA_GAS TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON CONTRATTO TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON DISTRETTO TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON STRUTTURA TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON MANDATO TO TECNICO;
GRANT SELECT, INSERT, UPDATE ON DIPENDENTE TO TECNICO;
GRANT SELECT,INSERT,UPDATE ON FORNITORE_POD TO TECNICO;
```

**GRANT CONNECT, CREATE SESSION TO USER;**

```

GRANT SELECT ON CALCOLO_IVA_TOTALE TO USER;
GRANT SELECT ON POTENZA_ANNUO TO USER;
GRANT SELECT ON CALCOLO_TOT_PAGARE TO USER;
GRANT SELECT ON CALCOLO_CONSUMO_ANNUO TO USER;
GRANT SELECT ON COSTO_TOTALE_PRECEDENTE TO USER;
GRANT EXECUTE ON PAGAMENTO_BOLLETTA TO USER;
GRANT EXECUTE ON UPDATE_ADEGUATEZZA TO USER;
GRANT EXECUTE ON UPDATE_CONTRATTO TO USER;
GRANT EXECUTE ON SPESE_TOT_BOLLETTE TO USER;
GRANT EXECUTE ON UPDATE_LETTURA_POD TO USER;
GRANT SELECT ON UTENTI_LICENZIATI TO USER;
GRANT SELECT ON UTENTI_ATTIVI TO USER;
GRANT SELECT ON SUPERUTENTI TO USER;
GRANT SELECT ON TABELLONE_STRUTTURA_LUCE TO USER ;
GRANT SELECT ON TABELLONE_STRUTTURA_GAS TO USER;

```

```

GRANT SELECT,INSERT,UPDATE ON BOLLETTA TO USER;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA_ELETTRICA TO USER;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA_ACQUA TO USER;
GRANT SELECT,INSERT,UPDATE ON BOLLETTA_GAS TO USER;
GRANT SELECT,INSERT,UPDATE ON CONTRATTO TO USER;
GRANT SELECT,INSERT,UPDATE ON DISTRETTO TO USER;
GRANT SELECT,INSERT,UPDATE ON STRUTTURA TO USER;
GRANT SELECT,INSERT,UPDATE MANDATO TO USER;
GRANT SELECT, INSERT, UPDATE ON DIPENDENTE TO USER;
GRANT SELECT,INSERT,UPDATE ON FORNITORE_POD TO USER;
GRANT SELECT ON DIPENDENTE TO USER;

```

Quando vogliamo rimuovere un dipendente dobbiamo oltre che settare il suo privilegio logico inserito nel database(solo per tener traccia che sia non attivo) dobbiamo effettuare la remove all privileges dall'utente in questione

## Future Implementazioni

Le estensioni possibili sono innumerevoli e il limite è dato solo dalla fantasia:  
 Vincoli dinamici/regole di business alternativi a quelli dati, ognuno dei quali comporta modifiche più o meno estese al codice proposto, sono:

- la chiave dopo essere generata verrà inviata direttamente tramite email al nuovo utente che richiede la chiave dal tecnico dipendente.

- Creare una procedura come Calcolo\_costo\_Totale per acqua e gas
- Creare un trigger che controlli un pod abbia un solo tipo di fornitura così che ad esso anche quando i contratti vengono disattivati il pod avrà obbligatoriamente sempre contratti dello stesso tipo (acqua,gas,luce)..

### **TABELLA RAISE APPLICATION ERROR**

<b>Numero</b>	<b>Descrizione</b>
-20002	La bolletta non corrisponde alla tipologia del contratto.
-20120	Il permesso inserito non è corretto
-20006	Esiste già un contratto attivo con quel pod, aggiornare lo stato del contratto presente.
-20001	Data non inserita
-20010	L'anno inserito risulta futuro
-20011	Il distretto non esiste
-20012	Il numero fattura della bolletta non esiste
-20090	Codice atto o numero fattura non esiste oppure
-20015	Il contratto inserito risulta inesistente
-20030	Il contratto risulta essere non attivo, impossibile modificare lo stato
-20022	Il pod o la bolletta inserita risultano essere inesistenti

-20100	Email non esistente
--------	---------------------