

Università degli Studi di Napoli Parthenope

Corso di laurea in informatica

Dipartimento di Scienze e Tecnologie

## COMANDO DI POLIZIA PROVINCIALE



Pasqualina ERRICO 0124/2786

Lorenzo MENNA 0124/2815

A.A 2023/2024

## Sommario

<b>1. Capitolo 1 .....</b>	<b>3</b>
1.1. Progettazione .....	3
1.1.1. Sintesi dei requisiti .....	3
1.1.2. Glossario.....	4
1.1.3. Modellazione concettuale.....	5
E/R.....	5
1.1.4. Analisi delle entità .....	6
1.1.5. Analisi delle associazioni .....	7
1.1.6. Modello relazionale .....	9
Traduzione tabelle .....	10
1.1.7. Volumi.....	11
1.1.8. Utenti, permessi e operazioni .....	12
Utenti e permessi.....	12
Use Case .....	12
Operazioni degli utenti.....	13
Tavole delle operazioni.....	18
1.1.9. Vincoli .....	19
Statici .....	19
Dinamici .....	20
1.1.10.Normalizzazione .....	21
Prima forma normale .....	21
Seconda forma normale .....	21
Terza forma normale.....	21
<b>2. Capitolo 2 .....</b>	<b>22</b>
2.1. Implementazione.....	22
2.1.1. Utenti .....	22
2.1.2. DDL.....	24
2.1.3. DML .....	29
2.1.4. Viste .....	80
2.1.5. Triggers.....	81
2.1.6. Procedure .....	84

# CAPITOLO 1

## Progettazione

### Sintesi dei requisiti

Si vuole realizzare un database per la gestione di un comando di polizia provinciale. Con il seguente, si vuole gestire la parte riguardante vari controlli su vari ambiti, tali controlli sono definiti sulle denunce, sulle multe, sul controllo dei documenti appartenenti alle persone e sulla gestione della turnazione di lavoro dei poliziotti, che si suddividono in poliziotti in piantone e in pattuglia, i quali avranno attrezzature specifiche.

Di ogni turno vengono registrati la sede di riferimento, la data e l'orario effettivo di lavoro.

Di ogni poliziotto vengono registrati i suoi dati anagrafici e il suo codice univoco.

Per quanto riguarda le denunce vengono registrati la loro data di emissione e prevede la registrazione sia di chi ha deposto la denuncia e, se riconosciuto l'imputato, anche quest'ultimo.

Invece per i controlli effettuati alle persone, viene registrato un documento preso dalla persona in questione per il controllo, e vanno registrati la data di scadenza del documento, il suo numero di documento, la data di emissione del documento e il tipo di documento esaminato.

Tali documenti vengono rilasciati da enti specializzati, i quali hanno un codice, un indirizzo e inoltre deve essere specificato il tipo di ente.

Parlando invece dell'attrezzatura specifica assegnata ai soli poliziotti in pattuglia, dobbiamo definire l'orario di acquisizione, di reso e la data di utilizzo.

I poliziotti in piantone possono sia gestire gli uffici denunce alla quale vengono assegnati, i quali verranno definiti per numero di ufficio, sia controllare le armi possedute nel comando, in quanto hanno un codice unico per ogni arma e che vengono definite anche per il modello e per il tipo.

## Glossario

La seguente tabella spiega il significato di alcuni termini particolari utilizzati nella sintesi:

TERMINE	DEFINIZIONE	SINONIMI
<i>Poliziotto in piantone</i>	<i>Un poliziotto che svolge, generalmente parlando, il compito di sorveglianza e di controllo all'interno di una struttura specifica</i>	Poliziotto di guardia, Sentinella
<i>Poliziotto in pattuglia</i>	<i>Un poliziotto che svolge il compito di monitorare una determinata area, a piedi, o con un qualunque mezzo in dotazione prestato dal comando di riferimento</i>	Poliziotto in perlustrazione, Rondista, Pattugliatore.
<i>Sede</i>	<i>La sede è il luogo dove avviene il servizio del poliziotto</i>	Dipartimento, Base, Stabilimento
<i>Data di emissione</i>	<i>La data di emissione è la data di rilascio di un documento da parte di un ente autorizzato</i>	Data di rilascio, Data di uscita
<i>Ente</i>	<i>Un ente è un'organizzazione o una struttura che svolge funzioni pubbliche o amministrative. Può essere un'istituzione creata per gestire servizi pubblici, come un comune, una regione, una scuola pubblica, o un'azienda sanitaria. Gli enti statali operano per il bene comune e sono finanziati con fondi pubblici.</i>	Organismo, Istituzione, Autorità

Tabella del glossario

# Modellazione Concettuale

E/R

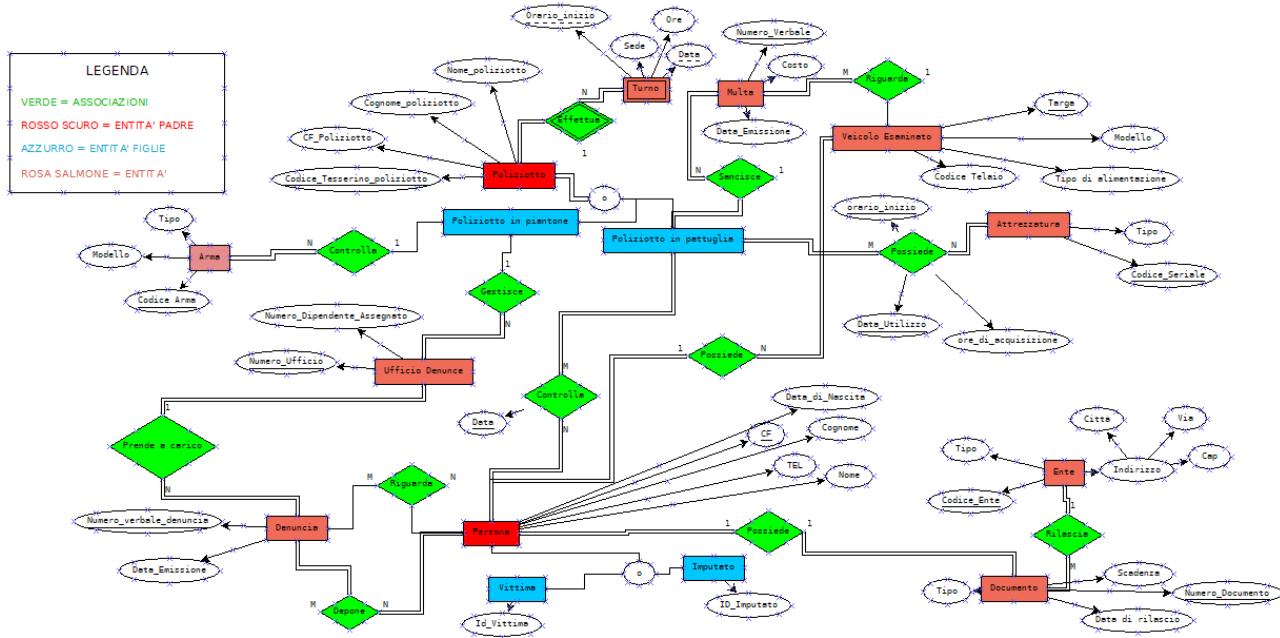


Figura 1.1: Diagramma EE/R

## Analisi entità

- **Persona** (CF, Nome, Cognome, Data\_nascita, telefono) Entità necessaria per la specializzazione. Questa entità permette di specificare se la persona in questione è una vittima, un imputato, tutti e due o nessuno dei due. La chiave utilizzata è il codice fiscale della persona.
- **Imputato** (ID\_imputato) Specializzazione di Persona.  
Rappresenta l'ipotetico colpevole che viene denunciato.
- **Vittima** (ID\_vittima) Specializzazione di Persona.  
Rappresenta la persona che depone la denuncia.
- **Documento** (Numero\_documento, tipo, scadenza, data di rilascio) Entità che rappresenta i documenti posseduti dalla persona. La chiave utilizzata è il numero documento, una chiave naturale generata alla creazione del documento.
- **Ente** (Codice\_ente, tipo, via, cap, città) Entità che rappresenta l'ente fornitore del documento in questione. La chiave naturale utilizzata è il codice dell'ente.
- **Denuncia** (Numero\_verbale, Data\_emissione) L'entità denuncia rappresenta il modello che la vittima depone; infatti, la sua chiave naturale è il numero\_verbale della denuncia.
- **Ufficio Denunce** (Numero\_ufficio, Nome\_dipendente\_assegnato) L'entità ufficio denunce rappresenta tutti gli sportelli disponibili per la raccolta delle denunce. Al suo interno troviamo anche il nome del dipendente che lo gestisce. La chiave naturale utilizzata è il numero ufficio.
- **Poliziotto** (Codice\_tesserino, CF, Nome\_poliziotto, Cognome\_poliziotto) Entità necessaria per la specializzazione. Questa entità permette di specificare se il poliziotto è un poliziotto in pattuglia oppure in piantone. Assegnandogli di conseguenza mansioni diverse. La chiave utilizzata è il codice tesserino poliziotto.
- **Poliziotto in pattuglia** Specializzazione di Poliziotto. Rappresenta il ruolo che può assumere il poliziotto in una determinata giornata.
- **Poliziotto in piantone** Specializzazione di Poliziotto. Rappresenta un altro ruolo che può assumere il poliziotto in una determinata giornata.
- **Arma** (Codice\_arma, tipo, modello) L'entità arma definisce tutte le armi che vengono controllate dai poliziotti in piantone. La chiave utilizzata è il codice arma, univoco per ogni singola arma.
- **Attrezzatura** (Codice\_attrezzatura, tipo) Quest'entità rappresenta l'attrezzatura extra equipaggiata dai soli poliziotti in pattuglia. La chiave utilizzata è il codice attrezzatura, che identifica in modo permanente ogni attrezzatura.
- **Turno** (Data, orario d'inizio, sede, ore) Entità debole di Poliziotto. Rappresenta il turno assegnato ad ogni poliziotto presente nel comando di polizia. Le chiavi deboli usate sono la Data e l'orario d'inizio.
- **Multa** (Numero\_verbale\_multa, data\_emissione, costo) Questa entità rappresenta le multe che possono essere applicate ai veicoli esaminati. La chiave utilizzata è il numero verbale della multa, creatasi al momento della verbalizzazione.
- **Veicolo Esaminato** (Targa, modello, tipo\_di\_alimentazione, codice\_telaio) Questa entità rappresenta tutti i veicoli esaminati, che saranno sottoposti alla sanzione. La chiave utilizzata è la targa, applicata all'immatricolazione della macchina.

## Analisi Associazioni

- **[PERSONA, DENUNCIA]**

### **MOLTEPLICITÀ N:M**

Una persona può deporre più denunce, e una denuncia può essere effettuata da un gruppo.

#### **TOTALITÀ**

Tutte le persone possono deporre delle denunce, e tutte le denunce sono deposte da persone.

- **[PERSONA, DENUNCIA]**

### **MOLTEPLICITÀ N:M**

Una persona può ricevere più denunce, ed una denuncia può riguardare più persone.

#### **TOTALITÀ**

Non tutte le denunce riguardano persone, e non si conoscono tutte le persone denunciate, esistono quindi delle denunce contro ignoti.

- **[PERSONA, DOCUMENTO]**

### **MOLTEPLICITÀ 1:1**

Una persona può possedere un solo documento, anche di tipo diverso ma sempre e solo uno, un documento è posseduto da una sola persona.

#### **TOTALITÀ**

Tutte le persone possiedono un documento e tutti i documenti sono posseduti da persone.

- **[ENTE, DOCUMENTO]**

### **MOLTEPLICITÀ 1:N**

Un ente rilascia più documenti ma un documento è rilasciato da un solo ente.

#### **TOTALITÀ**

Tutti gli enti rilasciano documenti, e tutti i documenti sono rilasciati da enti.

- **[POLIZIOTTO IN PATTUGLIA, PERSONA]**

### **MOLTEPLICITÀ N:M**

Un poliziotto in pattuglia controlla più persone e una persona può essere controllata da più poliziotti in pattuglia.

#### **TOTALITÀ**

Tutte le persone sono controllate da poliziotti in pattuglia, tutti i poliziotti in pattuglia controllano le persone.

- **[POLIZIOTTO IN PATTUGLIA, ATTREZZATURA]**

### **MOLTEPLICITÀ N:M**

Un poliziotto in pattuglia possiede più attrezzature in varie giornate e un'attrezzatura in varie giornate è posseduto da più poliziotti in pattuglia.

- **[POLIZIOTTO IN PATTUGLIA, MULTA]**

### **MOLTEPLICITÀ 1:N**

Un poliziotto in pattuglia sancisce più multe, una multa è sancita da un solo poliziotto in pattuglia.

## **TOTALITÀ**

Tutti i poliziotti in pattuglia sanciscono multe e tutte le multe sono sancite da poliziotti in pattuglia.

- **[MULTA, VEICOLO ESAMINATO]**

### **MOLTEPLICITÀ N:1**

Un veicolo esaminato può avere più multe, ma una multa può riguardare un solo veicolo esaminato.

## **TOTALITÀ**

Tutte le multe riguardano veicoli esaminati ma non tutti i veicoli esaminati devono necessariamente avere una multa.

- **[UFFICIO DENUNCE, DENUNCIA]**

### **MOLTEPLICITÀ 1:N**

Un ufficio denunce prende a carico più denunce e una denuncia può essere presa a carico solo da un ufficio denunce.

## **TOTALITÀ**

Tutti gli uffici denunce prendono a carico le denunce e tutte le denunce vengono prese a carico dagli uffici denunce.

- **[POLIZIOTTO IN PIANTONE, UFFICIO DENUNCE]**

### **MOLTEPLICITÀ 1:N**

Un poliziotto in piantone gestisce più uffici denunce e un ufficio denunce è gestito da un solo poliziotto in piantone.

## **TOTALITÀ**

Non tutti i poliziotti in piantone gestiscono gli uffici denunce ma tutti gli uffici denunce vengono gestiti da poliziotti in piantone.

- **[POLIZIOTTO IN PIANTONE, ARMA]**

### **MOLTEPLICITÀ 1:N**

Un poliziotto in piantone controlla più armi e un'arma viene controllata da un solo poliziotto in piantone.

## **TOTALITÀ**

Non tutti i poliziotti in piantone controllano armi, ma tutte le armi vengono controllate da poliziotti in piantone.

- **[POLIZIOTTO, TURNO]**

### **MOLTEPLICITÀ 1:N**

Un poliziotto svolge più turni e un turno è svolto da un poliziotto.

## **TOTALITÀ**

Tutti i poliziotti svolgono dei turni e tutti i turni sono svolti da poliziotti.

## Modello Relazionale

### Traduzione tabelle

Le tabelle sono state tradotte dallo schema EE/R secondo le regole di traduzione studiate durante il corso.

Nel caso di molteplicità 1:N la chiave è stata inserita nella tabella con molteplicità N, dove la chiave posizionata in N ha una freccia che la collega alla chiave primaria dell'entità con molteplicità 1.

In caso di associazioni N:M si crea una tabella di transizione, ovvero si crea una nuova tabella contenente la chiave delle altre due entità ed eventuali attributi.

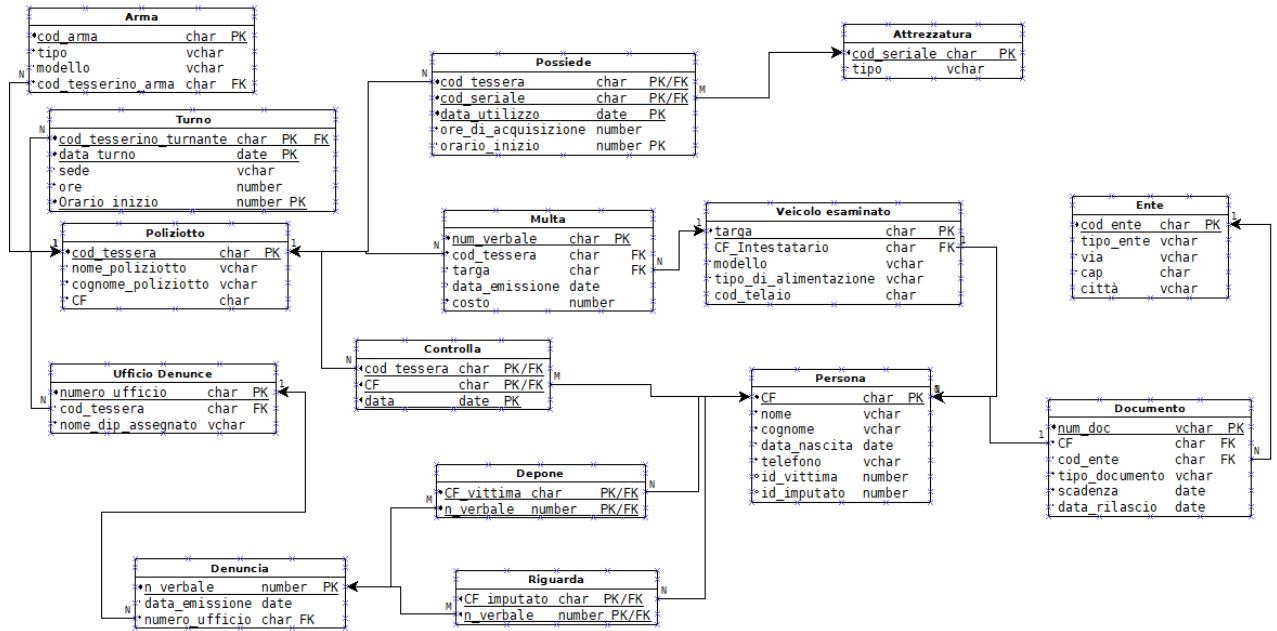


Figura 1.2: Modello Relazionale

## Volumi

Lo schema seguente descrive quanti dati sono presenti al momento della creazione nelle tabelle e l'incremento dei dati dopo un certo lasso di tempo.

TABELLA	VOLUME	INCREMENTO	TEMPO
Vittima	20.000	10.000	Anno
Imputato	10.000	5.000	Anno
Persona	40.000	20.000	Anno
Turno	8.000	3.000	Mese
Multa	10.000	2.000	Anno
Veicolo esaminato	13.000	6.000	Anno
Arma	900	90	Anno
Ufficio denunce	50	3	Anno
Poliziotto	400	40	Anno
Attrezzatura	300	30	Anno
Poliziotto Controlla Persona	25.000	15.000	Anno
Denuncia	25.000	11.000	Anno
Persona Depone Denuncia	25.000	11.000	Anno
Denuncia Riguarda Persona	10.000	5.000	Anno
Poliziotto Possiede Attrezzatura	300	30	Anno
Documento	50.000	22.000	Anno
Ente	500	30	Anno

Tali dati rappresentano in modo verosimile l'andamento generale di una regione di media-piccola dimensione.

## Utenti, permessi e operazioni

### Utenti e permessi

Nella tabella qui presente vengono riportati gli utenti del database, la loro tipologia, una stima sui volumi e tutti i loro permessi.

Utente	Tipo	Volume	Permessi
DB_Commando	Admin	1	ALL
Colonnello	Comune	10	*
Poliziotto	Comune	400	*
Armiere	Comune	50	*
Dipendente	Comune	50	*

\*I permessi vengono elencati a pag. 22

### Use Case

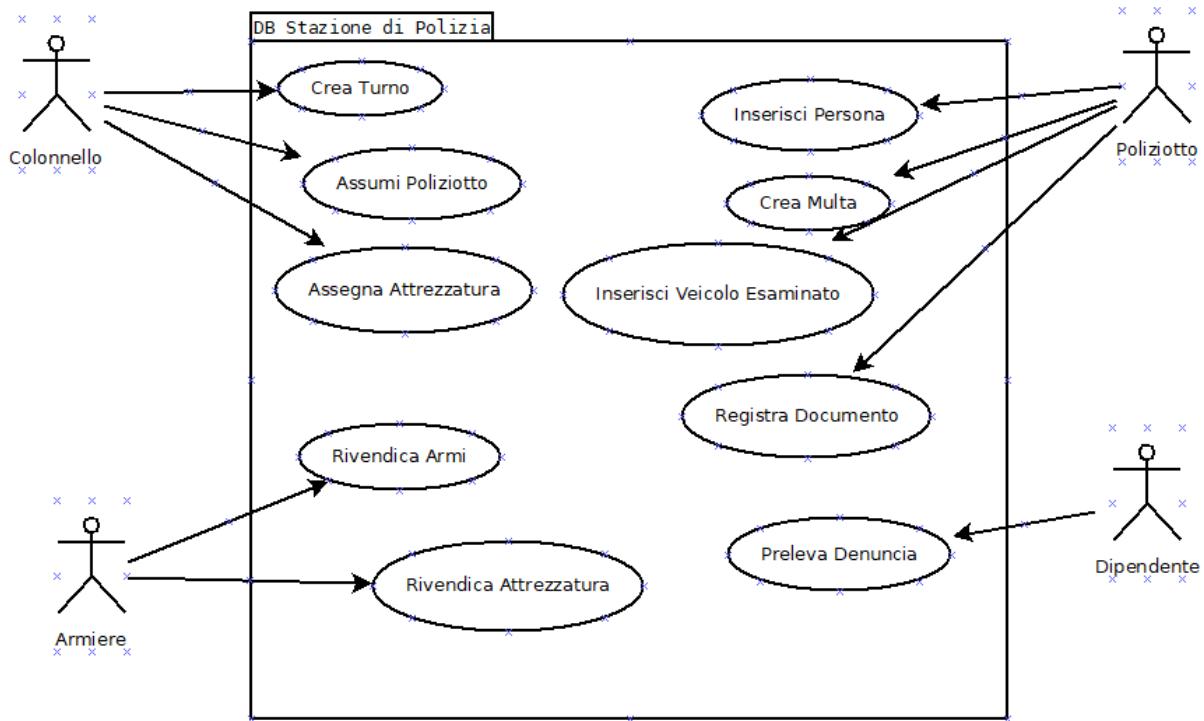


Figura 1.3 Diagramma dei casi d'uso

## Operazioni Utenti

Le operazioni effettuate dal colonnello sono quelle di **creazione turni, assunzione poliziotti e assegnazione delle attrezzature**.

Con l'assegnazione del turno si intende prima fare un controllo se il poliziotto in questione in un dato intervallo di tempo ha già ricevuto un turno preposto. Se tale controllo da esito negativo il poliziotto viene registrato al seguente turno.

L'operazione di assunzione poliziotti invece permette di assumere un poliziotto. Nessun caso specifico, devono solo essere inseriti correttamente tutti i dati per poterlo inserire correttamente.

L'operazione di assegnazione attrezzature ha lo scopo di assegnare ad un poliziotto in pattuglia l'attrezzatura. Prima verifica se attraverso un'altra operazione, rivendica attrezzatura, l'attrezzatura è già stata registrata dall'armiere, poi verifica se il poliziotto esiste. Se tutte le verifiche sono negative allora avviene tale operazione.

Le operazioni effettuate dal poliziotto sono quelle dell'**inserimento persone, creazione multe, inserimento di veicoli esaminati e registrazione di documenti**.

L'operazione di inserimento persone permette di inserire le persone nelle tuple. Nessun controllo particolare.

L'operazione di creazione multe permette al poliziotto di inserire le multe effettuate nel corso della giornata. È importante ricordarsi che per effettuare la multa il poliziotto nel caso di un nuovo veicolo esaminato debba prima inserire attraverso l'operazione apposita la targa del veicolo.

L'operazione di inserimento dei veicoli esaminati permette al poliziotto di inserire i nuovi veicoli controllati giornalmente.

L'operazione registrazione di documenti permette al poliziotto di inserire i nuovi documenti visionati durante il giorno.

Le operazioni effettuate dall'armiere sono quelle di **rivendicazione arma e rivendicazione attrezzatura**.

L'operazione di rivendicazione arma permette all'armiere di inserire le nuove armi disponibili e di assegnarle al poliziotto che le sorveglierà. Controllerà l'esistenza del tesserino, controllerà il codice dell'arma se è definito e alla fine inserirà la nuova arma.

L'operazione di rivendicazione attrezzatura permette all'armiere di inserire i nuovi equipaggiamenti disponibili senza però poterle assegnarle ai poliziotti.

Le operazioni effettuate dal dipendente sono quelle di poter **prelevare delle denunce**.

Qui, infatti, l'operazione di prelievo delle denunce permette al dipendente di poter raccogliere le denunce delle vittime. Verranno effettuati vari tipi di controlli:

Il primo consiste nel controllare se esiste già un id della persona che denuncia, altrimenti la si crea appositamente per la nuova vittima.

Il secondo consiste di fare lo stesso anche per la seconda vittima (in caso ci siano più persone a denunciare).

Il terzo consiste nel verificare se la denuncia è contro ignoti, in caso si crea un codice fiscale fittizio, una data fittizia di nascita e un numero di telefono fittizio prima di creare la denuncia.

Infine, se tutto è correttamente verificato, inserisce la denuncia.

Operazione	Crea turni
Scopo	Creare un turno per il poliziotto
Argomenti	Codice_tesserino_turnante Data_turno ora_inizio ore sede
Risultato	Aggiunta tupla Turno
Errori	Codice tesserino non valido Ora di inizio o durata del turno non valida
Tabelle	Turno Poliziotto

Tabella 1.3: Descrittore operazione: Crea turni

Operazione	Assumi poliziotto
Scopo	Assumere un poliziotto
Argomenti	Codice_tesserino Nome_poliziotto Cognome_poliziotto CF_poliziotto
Risultato	Aggiunta tupla Poliziotto
Errori	Codice tesserino non valido Codice fiscale non valido
Tabelle	Poliziotto

Tabella 1.4: Descrittore operazione: Assumi poliziotto

Operazione	Assegna Attrezzatura
Scopo	Assegna attrezzatura ad un poliziotto
Argomenti	Codice_seriale Codice_tesserino_poliziotto Data_utilizzo Orario_Inizio Ore_di_acquisizione
Risultato	Aggiunta tupla Possiede
Errori	Codice tesserino non valido Attrezzatura non registrata
Tabelle	Poliziotto Possiede Attrezzatura

Tabella 1.5: Descrittore operazione: Assegna Attrezzatura

Operazione	Inserisci persona
Scopo	Inserisce una persona nel database
Argomenti	CF nome cognome data_nascita telefono id_vittima id_imputato
Risultato	Aggiunta tupla Persona
Errori	Violazione di vincolo di chiave CF
Tabelle	Persona

Tabella 1.6: Descrittore operazione: Inserisci Persona

Operazione	Definisci Multa
Scopo	Inserisce una multa
Argomenti	numero_verbale_multa data_emissione_multa costo codice_tesserino_multante targa_multata
Risultato	Aggiunta tupla Multa
Errori	Errore durante l'inserimento della multa
Tabelle	Multa Poliziotto

Tabella 1.7: Descrittore operazione: Definisci multa

Operazione	Definisci veicolo esaminato
Scopo	Inserisce un veicolo esaminato
Argomenti	targa modello_veicolo tipo_di_alimentazione codice_telaio CF_intestatario
Risultato	Aggiunta tupla veicolo esaminato
Errori	Errore durante l'inserimento del veicolo esaminato
Tabelle	Veicolo Esaminato Persona

Tabella 1.8: Descrittore operazione: Definisci veicolo esaminato

Operazione	Registra documento
Scopo	Inserisce un documento
Argomenti	numero_documento tipo_documento scadenza data_di_rilascio codice_ente_rilasciatore CF_posessore
Risultato	Aggiunta tupla documento
Errori	Errore durante la registrazione del documento
Tabelle	Documento Persona

Tabella 1.9: Descrittore operazione: Registra documento

Operazione	Rivendica Arma
Scopo	Inserisce un'arma
Argomenti	codice_arma tipo modello codice_tesserino_arma
Risultato	Aggiunta tupla arma
Errori	Codice tesserino non valido Codice arma non valido Tutti i dati devono essere forniti Errore di conversione di tipo Codice arma è già presente
Tabelle	Arma Poliziotto

Tabella 2.0 Descrittore operazione: Rivendica Arma

Operazione	Rivendica Attrezzatura
Scopo	Inserisce un'attrezzatura

Argomenti	codice_seriale tipo_attrezzatura
Risultato	Aggiunta tupla Attrezzatura
Errori	Tipo attrezzatura non valido Codice seriale non valido Tutti i dati devono essere forniti Errore di conversione di tipo Codice seriale è già presente
Tabelle	Attrezzatura

Tabella 2.1 Descrittore operazione: Rivendica Attrezzatura

Operazione	Preleva Denuncia
Scopo	Inserisce una denuncia
Argomenti	numero_verbale_denuncia numero_ufficio data_emissione id_vittima id_vittima id_imputato
Risultato	Aggiunta tupla Denuncia
Errori	Violazione di un vincolo di chiave unica Nessun dato trovato Errore generico durante l'inserimento della denuncia
Tabelle	Denuncia Persona Riguarda Depone

Tabella 2.2 Descrittore operazione: Preleva denuncia

## Tavola delle operazioni

Le operazioni appena descritte vengono analizzate secondo la seguente tabella per il loro utilizzo (frequenza e periodo).

Operazione	Frequenza	Periodo
Crea turni	3.000	Mese
Assumi poliziotto	40	Anno
Assegna attrezzatura	200	Giorno
Inserisci persona	10.000	Anno
Definisci multa	2.000	Anno
Definisci veicolo esaminato	6.000	Anno
Registra documento	15.000	Anno
Rivendica arma	90	Anno
Rivendica attrezzatura	30	Anno
Preleva denuncia	11.000	Anno

## Vincoli

In questa sezione vengono elencati e descritti i vincoli di tipo statico e dinamico introdotti nel codice della creazione delle tabelle (DML) e nei trigger.

### Statici

Nelle tabelle

- Persona
- Poliziotto
- Depone
- Riguarda
- Controlla

Viene definito un vincolo sulla chiave CF. Viene definita come una stringa di 8 caratteri contenenti 2 cifre iniziali e 6 caratteri.

Nelle tabelle

- Poliziotto
- UfficioDenunce
- Turno
- Multa
- Controlla

Viene definito un vincolo sull'attributo chiave codice\_tesserino. Esso viene definita come una stringa di 10 caratteri contenenti 6 caratteri e 4 cifre finali.

Nella tabella Arma, l'attributo chiave codice\_arma viene definito come una stringa di 10 caratteri, contenenti 8 cifre iniziali e 2 caratteri.

Nella tabella Attrezzatura, l'attributo chiave codice\_seriale viene definito anche esso come una stringa di 10 caratteri contenenti 8 cifre iniziali e 2 caratteri finali.

Inoltre, nell'attributo tipo\_attrezzatura, i tipi di Kit definiti sono 3: kit base, kit medio e kit operazioni speciali. Non esistono altri tipi di Kit.

Nella tabella VeicoloEsaminato, il codice\_telaio viene definita come una stringa di 10 cifre.

Inoltre, i tipi di alimentazione vengono definiti di questi seguenti tipi:

- Benzina
- Diesel
- GPL
- Metano
- Ibrido
- Elettrico

Nella tabella Ente l'attributo chiave codice\_ente viene definita da una stringa da 7 caratteri formata da sole cifre.

Nella tabella Multa è stato creato un vincolo dove l'attributo costo deve essere maggiore di 0

### Dinamici

Per motivi logici la multa non potrà essere registrata con una data futura a quella corrente.

Il turno massimo di un poliziotto non può durare più di 36 ore a settimana.

Se l'attrezzatura viene presa da un poliziotto, non potrà essere utilizzata da altri poliziotti, almeno finché il periodo di utilizzo non sarà scaduto.

Verifica se un poliziotto è stato assegnato all'ufficio di competenza. Se ciò non è ancora avvenuto l'ufficio non potrà prelevare la denuncia.

Verifica se a un poliziotto è già stato assegnato un turno, e in caso affermativo, durante quel periodo non gli potrà essere assegnato un altro turno.

Verifica se il documento controllato non è già scaduto, in caso lo fosse, quest'ultimo non può essere registrato.

## Normalizzazione

### Prima forma normale

La prima forma normale viene rispettata in quanto nel database non vengono utilizzati attributi multivalue, attributi composti e le loro combinazioni.

### Seconda forma normale

Le tabelle con chiavi primarie composte, come Turno, Possiede, e Controlla, non hanno attributi non chiave che dipendono solo da una parte della chiave composta.

Le tabelle con chiavi primarie singole, come Persona, Poliziotto, e UfficioDenunce, non hanno attributi non chiave che dipendono parzialmente dalla chiave.

Di conseguenza anche la seconda forma normale viene rispettata.

### Terza forma normale

La terza forma normale non viene rispettata quando almeno un attributo non chiave dipende funzionalmente da un altro attributo, o un insieme di essi, non chiave né primi per cui la seconda forma normale deve essere rispettata.

Facendo riferimento a questa definizione la terza forma normale viene rispettata.

# CAPITOLO 2

## Implementazione

A seguire viene riportato il codice per la creazione di DB Comando.

### Utenti

Gli utenti e i loro permessi vengono creati dal seguente script:

```
Create user DB_Commando identified by Comando;
Grant all privileges to DB_Commando;

--Colonnello
Create user Colonnello identified by Colonnello;
Grant connect, create session to Colonnello;
Grant execute on creaTurno to Colonnello;
Grant execute on assumiPoliziotto to Colonnello;
Grant execute on assegnaAttrezzatura to Colonnello;
Grant select on DenunceUltimoMese to Colonnello;
Grant select on VistaImputati to Colonnello;
Grant select on VistaVittime to Colonnello;
Grant select on MulteUltimiTreMesi to Colonnello;
Grant all on Turno to Colonnello;
Grant all on Poliziotto to Colonnello;
Grant all on Attrezzatura to Colonnello;
--Armiere
Create user Armiere identified by Armiere;
Grant connect, create session to Armiere;
Grant execute on rivendicaArma to Armiere;
Grant execute on rivendicaAttrezzatura to Armiere;
Grant select on Attrezzatura to Armiere;
Grant select on Armi to Armiere;

--Dipendente
Create user Dipendente identified by Dipendente;
Grant connect, create session to Dipendente;
Grant execute on prelevaDenuncia to Dipendente;
Grant select on DenunceUltimoMese to Dipendente;
Grant select on VistaImputati to Dipendente;
Grant select on VistaVittime to Dipendente;
Grant select on Denuncia to Dipendente;
Grant select on Depone to Dipendente;
Grant select on Riguarda to Dipendente;
```

```
--Poliziotto
Create user Poliziotto identified by Poliziotto;
Grant connect, create session to Poliziotto;
Grant execute on inserisciPersona to Poliziotto;
Grant execute on definisciMultta to Poliziotto;
Grant execute on definisciVeicoloEsaminato to Poliziotto;
Grant execute on registraDocumento to Poliziotto;
Grant select on DenunceUltimoMese to Poliziotto;
Grant select on VistaImputati to Poliziotto;
Grant select on VistaVittime to Poliziotto;
Grant select on MulteUltimiTreMesi to Poliziotto;
Grant all on Persona to Poliziotto;
Grant all on VeicoloEsaminato to Poliziotto;
Grant all on Multa to Poliziotto;
Grant all on Documento to Poliziotto;
```

## DDL

Prima della creazione delle tabelle vengono settati delle variabili di sistema atte a personalizzare l'esperienza d'uso del db \*.

```

drop table Possiede cascade constraints;
drop table Controlla cascade constraints;
drop table Depone cascade constraints;
drop table Riguarda cascade constraints;
drop table Documento cascade constraints;
drop table Ente cascade constraints;
drop table Multa cascade constraints;
drop table VeicoloEsaminato cascade constraints;
drop table Attrezzatura cascade constraints;
drop table Turno cascade constraints;
drop table Arma cascade constraints;
drop table Denuncia cascade constraints;
drop table UfficioDenunce cascade constraints;
drop table Poliziotto cascade constraints;
drop table Persona cascade constraints;

create table Persona(
    CF char(8) primary key,
    nome varchar (20) not null,
    cognome varchar (20) not null,
    data_nascita date not null,
    telefono varchar (16) not null,
    id_vittima number(10,0) unique,
    id_imputato number(10,0) unique,
    constraint CF_corretto check(REGEXP_LIKE(CF, '^\\d{2}[A-Z]{6}$'))
);

create table Poliziotto(
    codice_tesserino char(10) primary key,
    nome_poliziotto varchar (20) not null,
    cognome_poliziotto varchar(20) not null,
    CF_poliziotto char(8) not null unique,
    constraint CF_corretto_poliziotto check(REGEXP_LIKE(CF_poliziotto, '^\\d{2}[A-Z]{6}$')),
    constraint codice_tesserino_corretto check (REGEXP_LIKE(codice_tesserino, '^[A-Z]{6}\\d{4}$'))
);

create table UfficioDenunce(
    numero char(3) primary key,
    codice_tesserino_ctrl char(10) not null,
    nome_dipendente_assegnato varchar(20) not null,

```

```

constraint fkTesserino foreign key (codice_tesserino_ctrl) references
Poliziotto(codice_tesserino),
constraint codice_tesserino_corretto_ctrl check
(REGEXP_LIKE(codice_tesserino_ctrl, '^[A-Z]{6}\d{4}$'))
);

create table Denuncia(
    numero_verbale_denuncia number(5,0) primary key,
    numero_ufficio char(3) not null,
    data_emissione date not null,
    constraint fkNumUff foreign key (numero_ufficio) references
UfficioDenunce(numero)
);

create table Arma(
    codice_arma char(10) primary key,
    tipo varchar(20) not null,
    modello varchar(20) not null,
    codice_tesserino_arma char(10) not null,
    constraint fkTesserinoArma foreign key (codice_tesserino_arma) references
Poliziotto (codice_tesserino),
    constraint codice_arma_corretto check (REGEXP_LIKE(codice_arma, '^\\d{8}[A-
Z]{2}$')),
    constraint codice_tesserino_corretto_arma check
(REGEXP_LIKE(codice_tesserino_arma, '^[A-Z]{6}\d{4}$'))
);

create table Turno(
    codice_tesserino_turnante char (10) not null,
    data_turno date not null,
    ora_inizio number(2,0) not null,
    ore number(2,0) not null,
    sede varchar(20) not null,
    primary key (codice_tesserino_turnante, data_turno, ora_inizio),
    constraint fkTesserinoTurno foreign key (codice_tesserino_turnante) references
Poliziotto (codice_tesserino),
    constraint codice_tesserino_corretto_turnante check
(REGEXP_LIKE(codice_tesserino_turnante, '^[A-Z]{6}\d{4}$')),
    constraint ora_valida check (ora_inizio >= 0 and ora_inizio <= 23 and ore > 0
and ora_inizio + ore <= 24)
);

create table Attrezzatura(
    codice_seriale char(10) primary key,
    tipo_attrezzatura varchar(25) not null check (tipo_attrezzatura in ('Kit base',
'Kit medio', 'Kit operazioni speciali')),
    constraint codice_seriale_corretto check (REGEXP_LIKE(codice_seriale, '^\\d{8}[A-
Z]{2}$'))
);

```

```

);

create table VeicoloEsaminato(
    targa char(7) primary key,
    modello_veicolo varchar(20) not null,
    tipo_di_alimentazione varchar(20) not null check (tipo_di_alimentazione in
('Diesel', 'Benzina', 'GPL', 'Metano', 'Ibrido', 'Elettrico')),
    codice_telaio char(10) not null unique,
    CF_intestatario char(8) not null,
    constraint fkCFintest foreign key (CF_intestatario) references Persona(CF),
    constraint CF_corretto_intestatario check(REGEXP_LIKE(CF_intestatario,
'^\d{2}[A-Z]{6}$')),
    constraint codice_telaio_corretto check (REGEXP_LIKE(codice_telaio,'^\d{10}$'))
);

create table Multa(
    numero_verbale_multa char(5) primary key,
    data_emissione_multa date not null,
    costo number(5,2) not null check (costo > 0),
    codice_tesserino_multante char(10) not null,
    targa_multata char(7) not null,
    constraint fkTesserinoMultta foreign key (codice_tesserino_multante) references
Poliziotto(codice_tesserino),
    constraint fkTargaMultta foreign key (targa_multata) references
VeicoloEsaminato(targa),
    constraint codice_tesserino_corretto_multante check
(REGEXP_LIKE(codice_tesserino_multante,'^[A-Z]{6}\d{4}$'))
);

create table Ente(
    codice_ente char(7) primary key,
    tipo_ente varchar(20) not null,
    via_ente varchar(20) not null,
    cap_ente char(5) not null,
    citta varchar(30) not null,
    constraint codice_ente_corretto check (REGEXP_LIKE(codice_ente,'^\d{7}$'))
);

create table Documento(
    numero_documento varchar(10) primary key,
    tipo_documento varchar(20) not null,
    scadenza date not null,
    data_di_rilascio date not null,
    codice_ente_rilasciatario char(7) not null,
    CF_possessore char(8) not null,
    constraint fkEnteRilascia foreign key (codice_ente_rilasciatario) references
Ente(codice_ente),
    constraint fkCFPossesso foreign key (CF_possessore) references Persona(CF),

```

```

constraint CF_corretto_posessore check(REGEXP_LIKE(CF_posessore, '^\\d{2}[A-Z]{6}$')),
constraint Codice_ente_corretto_rilasciatario
check(REGEXP_LIKE(codice_ente_rilasciatario, '^\\d{7}$'))
);

create table Riguarda(
    CF_imputato char(8),
    numero_verbale_a_carico number(5,0) not null,
    primary key (CF_imputato, numero_verbale_a_carico),
    constraint fkCFImp foreign key (CF_imputato) references Persona(CF),
    constraint fkVerbaleImp foreign key (numero_verbale_a_carico) references
Denuncia(numero_verbale_denuncia),
    constraint CF_corretto_imputato check(REGEXP_LIKE(CF_imputato, '^\\d{2}[A-Z]{6}$'))
);

create table Depone(
    CF_vittima char(8),
    numero_verbale_deposto number(5,0) not null,
    primary key (CF_vittima, numero_verbale_deposto),
    constraint fkCFVitt foreign key (CF_vittima) references Persona(CF),
    constraint fkVerbaleVitt foreign key (numero_verbale_deposto) references
Denuncia(numero_verbale_denuncia),
    constraint CF_corretto_vittima check(REGEXP_LIKE(CF_vittima, '^\\d{2}[A-Z]{6}$'))
);

create table Controlla(
    codice_tesserino_controllore char(10) not null,
    CF_persona_controllata char(8) not null,
    data_controllo date not null,
    primary key(codice_tesserino_controllore, CF_persona_controllata,
data_controllo),
    constraint fkCTcontr foreign key (codice_tesserino_controllore) references
Poliziotto(codice_tesserino),
    constraint fkCFcontrollata foreign key (CF_persona_controllata) references
Persona(CF),
    constraint CF_corretto_persona_controllata check
(REGEXP_LIKE(CF_persona_controllata, '^\\d{2}[A-Z]{6}$')),
    constraint codice_tesserino_corretto_controllore check
(REGEXP_LIKE(codice_tesserino_controllore, '^[A-Z]{6}\\d{4}$'))
);

```

```

create table Possiede(
    codice_seriale_posseduto char(10) not null,
    codice_tesserino_posessore char(10) not null,
    data_utilizzo date not null,

```

```
orario_inizio number(2,0) not null,  
ore_di_acquisizione number(2,0) not null,  
  
primary key(codice_seriale_posseduto, codice_tesserino_posessore,  
data_utilizzo, orario_inizio),  
constraint fkCTcontrpss foreign key (codice_tesserino_posessore) references  
Poliziotto(codice_tesserino),  
constraint fkCSpss foreign key (codice_seriale_posseduto) references  
Attrezzatura(codice_seriale),  
constraint codice_seriale_controllato_posseduto check  
(REGEXP_LIKE(codice_seriale_posseduto, '^\\d{8}[A-Z]{2}$')),  
constraint codice_tesserino_corretto_posessore check  
(REGEXP_LIKE(codice_tesserino_posessore, '^[A-Z]{6}\\d{4}$')),  
constraint check_orario_inizio check (orario_inizio >= 0 and orario_inizio <=  
23),  
constraint check_ore check (ore_di_acquisizione >= 0)  
);
```

\* Nella progettazione delle tabella unica Poliziotto e Persona, abbiamo scelto di non includere il discriminante per mantenere la chiarezza e la leggibilità dei dati. Riteniamo che questa scelta semplifichi l'interpretazione delle informazioni senza compromettere l'accuratezza.

## DML

Per quanto riguarda la creazione del popolamento viene utilizzato un file script di dimensione ridotte (circa 15 tuple per tabella, per alcune sono state implementate più tuple per il corretto funzionamento dimostrativo).

```
delete from Riguarda;
delete from Depone;
delete from Controlla;
delete from Possiede;
delete from Multa;
delete from VeicoloEsaminato;
delete from Arma;
delete from Turno;
delete from Denuncia;
delete from UfficioDenunce;
delete from Documento;
delete from Attrezzatura;
delete from Ente;
delete from Persona;
delete from Poliziotto;

/*Poliziotto*/
insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'MTTGVN0010',
    'Mattia',
    'Giovinelli',
    '05GVNMTT'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'LCUDDT0100',
    'Luca',
    'Diodato',
    '07DDTLCU'
);
```

```
insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'FDRPRM1110',
    'Federica',
    'Parmi',
    '10PRMFDR'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'LSESPS1211',
    'Elisa',
    'Esposito',
    '13SPSLSE'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'GBRBCC1002',
    'Gabriele',
    'Bocci',
    '12BCCGBR'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'VTTPLL1012',
    'Vittorio',
    'Pellicani',
    '13PLLVT'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
    'GNNSPS0021',
    'Gennaro',
    'Esposito',
    '11SPSGNN'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
```

```

values(
  'FRCTFN0090',
  'Francesco',
  'Tufano',
  '12TFNFRC'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'GRGMLT0091',
  'Giorgio',
  'Melito',
  '20MLTGRG'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'VLNBTT0024',
  'Valentina',
  'Battaglia',
  '24BTTVLN'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'PLATSS0909',
  'Paolo',
  'Tessitore',
  '22TSSPLA'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'GNLMNN0304',
  'Gianluca',
  'Manni',
  '23MNNGNL'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'GVNLCR0809',

```

```

'Giovanni',
'Lucarelli',
'12LCRGVN'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'GPGMSC0900',
  'Giuseppe',
  'Moscati',
  '21MSCGSP'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'VNCDCR1013',
  'Vincenzo',
  'De Crescenzo',
  '12DCRVNC'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'FDRSPS2020',
  'Federico',
  'Esposito',
  '15SPSFDR'
);

insert into Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
CF_poliziotto)
values(
  'MTTDCR1013',
  'Mattia',
  'De Crescenzo',
  '11DCRMTT'
);

```

```

/*Persona*/
insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
  '02LCTFRC',
  'Francesca',

```

```

'Licata',
to_date('11/07/2002', 'dd/mm/yyyy'),
'33348364740',
0000000001,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'90GNVNTN',
'Antonio',
'Genovese',
to_date('10/01/1990', 'dd/mm/yyyy'),
'32348364740',
0000000002,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'01RRCPQL',
'Pasqualina',
'Errico',
to_date('27/04/2001', 'dd/mm/yyyy'),
'3283364740',
0000000003,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'81STFMRC',
'Mariacarla',
'Staffa',
to_date('11/02/1981', 'dd/mm/yyyy'),
'33948364740',
0000000004,
null
);

```

```

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'03INCNDR',
'Andrea',
'Iannaccone',
to_date('19/11/2003', 'dd/mm/yyyy'),

```

```
'34492364740',
0000000005,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'88RRCNNA',
'Anna',
'Errico',
to_date('07/10/1988', 'dd/mm/yyyy'),
'3398364740',
0000000006,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'92RSSGNN',
'Giovanni',
'Rossi',
to_date('23/09/1992', 'dd/mm/yyyy'),
'3888364740',
0000000007,
0000000001
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'00BNCLSS',
'Alessandro',
'Bianchi',
to_date('09/07/2000', 'dd/mm/yyyy'),
'32288364740',
null,
null
);
```

```
insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '78CMSFRN',
    'Francesco',
    'Camastrà',
    to_date('11/05/1978', 'dd/mm/yyyy'),
    '3448364740',
    null,
    000000002
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '91MRTNTN',
    'Antonio',
    'Maratea',
    to_date('25/06/1991', 'dd/mm/yyyy'),
    '34248364740',
    null,
    000000003
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '99INNCRM',
    'Carmela',
    'Iannone',
    to_date('22/11/1999', 'dd/mm/yyyy'),
    '3278364740',
    null,
    000000004
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '66BRTPQL',
    'Pasquale',
    'Bortone',
    to_date('30/11/1966', 'dd/mm/yyyy'),
    '3374606740',
    null,
    000000005
);
```

```
insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
```

```

values(
    '67BLNCTR',
    'Caterina',
    'Bianchetti',
    to_date('11/10/1967', 'dd/mm/yyyy'),
    '33348364741',
    null,
    000000006
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '78LSSBNT',
    'Benito',
    'Alessandrini',
    to_date('11/01/1978', 'dd/mm/yyyy'),
    '339127392083',
    null,
    000000007
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '95LRAFCC',
    'Ilaria',
    'Ficocelli',
    to_date('10/10/1994', 'dd/mm/yy'),
    '3339174990',
    null,
    000000008
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
    '98DBLSVT',
    'Salvatore',
    'Di Bello',
    to_date('21/11/1998', 'dd/mm/yyyy'),
    '3661087080',
    000000008,
    null
);

```

```

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(

```

```

'80NREGNN',
'Giovanna',
'Neri',
to_date('20/10/1980', 'dd/mm/yyyy'),
'3492389264',
null,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'70SPSFBA',
'Fabio',
'Esposito',
to_date('11/10/1970', 'dd/mm/yyyy'),
'3485063264',
null,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'99RCCLNE',
'Elena',
'Ricci',
to_date('17/03/1999', 'dd/mm/yyyy'),
'3392379264',
null,
null
);

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'34FNTGNN',
'Giovanni',
'Fontana',
to_date('31/01/1934', 'dd/mm/yyyy'),
'3492389964',
null,
null
);

```

```

insert into Persona(CF, nome, cognome, data_nascita, telefono, id_vittima, id_imputato)
values(
'59GLLFRC',

```

```
'Francesca',
'Gallo',
to_date('22/04/1959', 'dd/mm/yyyy'),
'3492300064',
null,
null
);

/*Ente*/
insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0000120',
'Comune',
'Via San Felici',
'80131',
'Milano'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0000121',
'Comune',
'Piazza Spartani',
'82130',
'Firenze'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0010010',
'Questura',
'Vicolo Sant Egidio',
'80133',
'Milano'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0023004',
'Questura',
'Viale Marconi',
'80110',
'Napoli'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0060500',
```

```

'Motorizzazione',
'Piazza Martini',
'80150',
'Bari'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
  '0030291',
  'Questura',
  'Via Del Commercio',
  '80123',
  'Cagliari'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
  '0001231',
  'Motorizzazione',
  'Vicolo Corto',
  '80139',
  'Torino'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
  '0002148',
  'Motorizzazione',
  'Piazza Trieste',
  '80191',
  'Palermo'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
  '0100112',
  'Comune',
  'Piazza Campidoglio',
  '80111',
  'Potenza'
);

```

```

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
  '0002111',
  'Comune',
  'Vicolo Stretto',

```

```

'81120',
'Trieste'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0003221',
'Questura',
'Piazza Dante',
'81121',
'Marcianise'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0002234',
'Questura',
'Viale Dei Giardini',
'81220',
'Castellammare Di Stabia'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0010002',
'Motorizzazione',
'Via Roma',
'81130',
'Casoria'
);

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0001220',
'Motorizzazione',
'Piazza Giulio Cesare',
'81138',
'Milano'
);

```

```

insert into Ente(codice_ente, tipo_ente, via_ente, cap_ente, citta)
values(
'0002341',
'Comune',
'Via Marco Polo',
'80100',

```

```
'Caserta'  
);  
  
/*Veicolo Esaminato*/  
insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,  
codice_telaio, CF_intestatario)  
values(  
    'AP200PO',  
    'Hyundai',  
    'GPL',  
    '0040392012',  
    '03INCNDR'  
);  
  
insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,  
codice_telaio, CF_intestatario)  
values(  
    'DL200LO',  
    'Peugeout',  
    'Benzina',  
    '0042232012',  
    '00BNCLSS'  
);  
  
insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,  
codice_telaio, CF_intestatario)  
values(  
    'EF300PL',  
    'Mazda',  
    'Diesel',  
    '0040201049',  
    '03INCNDR'  
);  
  
insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,  
codice_telaio, CF_intestatario)  
values(  
    'GH1000P',  
    'Suzuki',  
    'Benzina',  
    '1002085493',  
    '00BNCLSS'  
);  
  
insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,  
codice_telaio, CF_intestatario)  
values(  
    'AZ200PA',
```

```
'Ford',
'Benzina',
'0241392022',
'91MRTNTN'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
  'CV033FF',
  'Ford',
  'GPL',
  '0030495032',
  '99INNCRM'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
  'CF102TL',
  'Mercedes',
  'Diesel',
  '0020332076',
  '66B RTPQL'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
  'BF200FL',
  'Audi',
  'Diesel',
  '0110392112',
  '67BLNCTR'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
  'GA211L0',
  'Fiat',
  'Elettrico',
  '2090344012',
  '78LSSBNT'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
```

```
values(
    'BM450LO',
    'Volkswagen',
    'Metano',
    '0050692444',
    '95LRAFCC'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
    'FA100PL',
    'Opel',
    'GPL',
    '0029485021',
    '98DBLSVT'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
    'FI100RL',
    'Vauxhall',
    'Benzina',
    '0200391102',
    '80NREGNN'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
    'FF121MN',
    'Ferrari',
    'Benzina',
    '0022399952',
    '70SPSFBA'
);

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
    'DE200PO',
    'Porsche',
    'Benzina',
    '5120392234',
    '99RCCLNE'
);
```

```

insert into VeicoloEsaminato(targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
values(
    'DF140LP',
    'Tesla',
    'Ibrido',
    '0392899557',
    '99RCCLNE'
);

/*Multa*/
insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00001',
    to_date('05/09/2022', 'dd/mm/yyyy'),
    125.00,
    'LCUDDT0100',
    'FF121MN'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00002',
    to_date('05/09/2022', 'dd/mm/yyyy'),
    75.00,
    'LCUDDT0100',
    'GH1000P'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00003',
    to_date('10/10/2022', 'dd/mm/yyyy'),
    500.00,
    'MTTGVN0010',
    'FF121MN'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00004',
    to_date('01/02/2023', 'dd/mm/yyyy'),
    50.00,
    'GBRBCC1002',

```

```
'FA100PL'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00005',
    to_date('02/01/2023', 'dd/mm/yyyy'),
    140.00,
    'LSESPS1211',
    'FA100PL'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00006',
    to_date('05/03/2023', 'dd/mm/yyyy'),
    130.00,
    'LSESPS1211',
    'BM450LO'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00007',
    to_date('06/04/2023', 'dd/mm/yyyy'),
    55.00,
    'GNNSPS0021',
    'AZ200PA'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00008',
    to_date('10/04/2023', 'dd/mm/yyyy'),
    500.00,
    'GRGMLT0091',
    'CV033FF'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00009',
    to_date('20/10/2023', 'dd/mm/yyyy'),
```

```
500.00,
'GNNSPS0021',
'GA211LO'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00010',
    to_date('21/11/2023', 'dd/mm/yyyy'),
    500.00,
    'GRGMLT0091',
    'AZ200PA'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00011',
    to_date('23/12/2023', 'dd/mm/yyyy'),
    500.00,
    'GVNLCR0809',
    'FF121MN'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00012',
    to_date('23/12/2023', 'dd/mm/yyyy'),
    500.00,
    'GVNLCR0809',
    'FI100RL'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
    '00013',
    to_date('12/01/2024', 'dd/mm/yyyy'),
    500.00,
    'VLNBT0024',
    'CV033FF'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
```

```

'00014',
to_date('13/01/2024', 'dd/mm/yyyy'),
500.00,
'PLATSS0909',
'DF140LP'
);

insert into Multa(numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
values(
'00015',
to_date('12/02/2024', 'dd/mm/yyyy'),
500.00,
'VNCDCR1013',
'FI100RL'
);

/*Arma*/
insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
'04039485PI',
'Pistola',
'Glock 17',
'GNLMNN0304'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
'04030385FA',
'Fucile automatico',
'AK-47',
'FRCTFN0090'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
'04209485MG',
'Mitraglietta',
'PKM',
'GPGMSC0900'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
'04039411TS',
'Taser',
'X2',
'FDRPRM1110'
);

```

```
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04039423PI',
    'Pistola',
    'Beretta 92FS',
    'GNLMNN0304'
);
insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04011485FA',
    'Fucile Automatico',
    'M16',
    'FRCTFN0090'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '00039485TS',
    'Taser',
    'Pulse',
    'FDRPRM1110'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '00000485FS',
    'Fucile S_Automatico',
    'AR-15',
    'FRCTFN0090'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04039325MG',
    'Mitraglietta',
    'MG3',
    'GPGMSC0900'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04039445MG',
    'Mitraglietta',
    'M240',
    'GPGMSC0900'
);
```

```

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '00019485FA',
    'Fucile Automatico',
    'FN SCAR',
    'FRCTFN0090'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04039026TS',
    'Taser',
    'Bolt',
    'FDRPRM1110'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04039665FA',
    'Fucile Automatico',
    'G36',
    'FRCTFN0090'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04030928PI',
    'Pistola',
    'Colt M1911',
    'GNLMNN0304'
);

insert into Arma(codice_arma, tipo, modello, codice_tesserino_arma)
values(
    '04039005MG',
    'Mitraglietta',
    'M249 SAW',
    'GPGMSC0900'
);

/*Turno*/
insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'LCUDDT0100',
    to_date('05/09/2022', 'DD/MM/YYYY'),
    17,
    6,
    'AREA1'
)

```

```
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'LCUDDT0100',
    to_date('12/09/2022', 'DD/MM/YYYY'),
    14,
    6,
    'AREA2'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'FDRPRM1110',
    to_date('20/01/2023', 'DD/MM/YYYY'),
    15,
    6,
    'AREA2'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'FDRPRM1110',
    to_date('23/02/2023', 'DD/MM/YYYY'),
    7,
    6,
    'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GNNSPS0021',
    to_date('06/04/2023', 'DD/MM/YYYY'),
    17,
    6,
    'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GNNSPS0021',
    to_date('20/10/2023', 'DD/MM/YYYY'),
    17,
    6,
    'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
```

```
values(
    'GPGMSC0900',
    to_date('11/05/2023', 'DD/MM/YYYY'),
    6,
    6,
    'AREA4'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GPGMSC0900',
    to_date('01/06/2023', 'DD/MM/YYYY'),
    10,
    6,
    'AREA5'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'FRCTFN0090',
    to_date('02/06/2023', 'DD/MM/YYYY'),
    1,
    6,
    'AREA5'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'FRCTFN0090',
    to_date('02/06/2023', 'DD/MM/YYYY'),
    12,
    6,
    'AREA5'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'FRCTFN0090',
    to_date('10/06/2023', 'DD/MM/YYYY'),
    17,
    4,
    'AREA6'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'FDRPRM1110',
    to_date('20/07/2023', 'DD/MM/YYYY'),
```

```
13,
6,
'AREA6'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'FDRPRM1110',
to_date('23/08/2023', 'DD/MM/YYYY'),
9,
6,
'AREA7'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'GNLMNN0304',
to_date('30/10/2023', 'DD/MM/YYYY'),
15,
6,
'AREA7'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'GNLMNN0304',
to_date('13/12/2023', 'DD/MM/YYYY'),
15,
6,
'AREA8'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'GNLMNN0304',
to_date('05/01/2024', 'DD/MM/YYYY'),
10,
6,
'AREA8'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'LSESPS1211',
to_date('02/01/2023', 'DD/MM/YYYY'),
14,
6,
'AREA8'
```

```
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'LSESPS1211',
    to_date('05/03/2023', 'DD/MM/YYYY'),
    10,
    6,
    'AREA5'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'MTTGVN0010',
    to_date('10/10/2022', 'DD/MM/YYYY'),
    7,
    6,
    'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GBRBCC1002',
    to_date('01/02/2023', 'DD/MM/YYYY'),
    10,
    6,
    'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'VTTPLL1012',
    to_date('05/02/2023', 'DD/MM/YYYY'),
    10,
    6,
    'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GRGMLT0091',
    to_date('10/04/2023', 'DD/MM/YYYY'),
    9,
    6,
    'AREA2'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
```

```
values(
    'GRGMLT0091',
    to_date('21/11/2023', 'DD/MM/YYYY'),
    7,
    6,
    'AREA6'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'VLNBT0024',
    to_date('12/01/2024', 'DD/MM/YYYY'),
    8,
    6,
    'AREA6'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'VLNBT0024',
    to_date('15/05/2024', 'DD/MM/YYYY'),
    8,
    6,
    'AREA6'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'PLATSS0909',
    to_date('13/01/2024', 'DD/MM/YYYY'),
    8,
    6,
    'AREA10'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GVNLCR0809',
    to_date('23/12/2023', 'DD/MM/YYYY'),
    8,
    6,
    'AREA9'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
    'GVNLCR0809',
    to_date('23/12/2023', 'DD/MM/YYYY'),
```

```
18,
3,
'AREA9'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'VNCDCR1013',
to_date('12/02/2024', 'DD/MM/YYYY'),
15,
6,
'AREA4'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'FDRSPS2020',
to_date('10/05/2024', 'DD/MM/YYYY'),
8,
6,
'AREA3'
);

insert into Turno(codice_tesserino_turnante, data_turno, ora_inizio, ore, sede)
values(
'MTTDCR1013',
TO_DATE('13/04/2024', 'DD/MM/YYYY'),
8,
6,
'AREA8'
);

/*Attrezzatura*/
insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
'00000001KB',
'Kit base'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
'00000002KB',
'Kit base'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
```

```
'00000003KB',
'Kit base'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '00000004KB',
  'Kit base'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '00000005KB',
  'Kit base'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '10000001KM',
  'Kit medio'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '10000002KM',
  'Kit medio'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '10000003KM',
  'Kit medio'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '10000004KM',
  'Kit medio'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '10000005KM',
  'Kit medio'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
```

```
'20000001KS',
'Kit operazioni speciali'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '20000002KS',
  'Kit operazioni speciali'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '20000003KS',
  'Kit operazioni speciali'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '20000004KS',
  'Kit operazioni speciali'
);

insert into Attrezzatura(codice_seriale, tipo_attrezzatura)
values(
  '20000005KS',
  'Kit operazioni speciali'
);

/*Documento*/
insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_posessore)
values(
  'CA94673GG',
  'Carta identita',
  to_date('11/07/2032', 'dd/mm/yyyy'),
  to_date('22/05/2021', 'dd/mm/yyyy'),
  '0000120',
  '02LCTFRC'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_posessore)
values(
  'CE94933NN',
  'Carta identita',
  to_date('10/01/2033', 'dd/mm/yyyy'),
```

```
to_date('22/08/2022', 'dd/mm/yyyy'),
'0000120',
'90GNVNTN'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_posessore)
values(
'CA940727GM',
'Carta identita',
to_date('27/04/2034', 'dd/mm/yyyy'),
to_date('22/11/2023', 'dd/mm/yyyy'),
'0100112',
'01RRCPQL'
);
insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_posessore)
values(
'CA00384TT',
'Carta identita',
to_date('11/02/2028', 'dd/mm/yyyy'),
to_date('17/10/2017', 'dd/mm/yyyy'),
'0100112',
'81STFMRC'
);
insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_posessore)
values(
'CA85434VG',
'Carta identita',
to_date('19/11/2027', 'dd/mm/yyyy'),
to_date('22/05/2016', 'dd/mm/yyyy'),
'0002111',
'03INCNDR'
);
insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_posessore)
values(
'CA017593HA',
'Carta identita',
to_date('07/10/2032', 'dd/mm/yyyy'),
to_date('29/05/2021', 'dd/mm/yyyy'),
```

```
'0002111',
'88RRCNNA'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CQ92843PO',
'Carta identita',
to_date('23/09/2025', 'dd/mm/yyyy'),
to_date('30/12/2014', 'dd/mm/yyyy'),
'0002341',
'92RSSGNN'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CV92843RR',
'Carta identita',
to_date('09/07/2027', 'dd/mm/yyyy'),
to_date('14/06/2016', 'dd/mm/yyyy'),
'0002341',
'00BNCLSS'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CA43759PZ',
'Carta identita',
to_date('11/05/2035', 'dd/mm/yyyy'),
to_date('02/03/2024', 'dd/mm/yyyy'),
'0002341',
'78CMSFRN'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'YB8465438',
'Passaporto',
to_date('25/06/2032', 'dd/mm/yyyy'),
to_date('04/09/2022', 'dd/mm/yyyy'),
```

```
'0010010',
'91MRTNTN'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CP94792VV',
'Carta identita',
to_date('22/11/2028', 'dd/mm/yyyy'),
to_date('22/05/2017', 'dd/mm/yyyy'),
'0002111',
'99INNCRM'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CQ87654GS',
'Carta identita',
to_date('30/11/2030', 'dd/mm/yyyy'),
to_date('11/12/2019', 'dd/mm/yyyy'),
'0002111',
'66B RTPQL'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CA94673GU',
'Carta identita',
to_date('11/10/2022', 'dd/mm/yyyy'),
to_date('14/08/2011', 'dd/mm/yyyy'),
'0100112',
'67BLNCTR'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'YA8543718',
'Passaporto',
to_date('11/01/2025', 'dd/mm/yyyy'),
to_date('28/02/2015', 'dd/mm/yyyy'),
```

```
'0003221',
'78LSSBNT'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CZ85400ED',
'Carta identita',
to_date('10/10/2033', 'dd/mm/yyyy'),
to_date('27/04/2022', 'dd/mm/yyyy'),
'0100112',
'95LRAFCC'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CU84933FR',
'Carta identita',
to_date('21/11/2035', 'dd/mm/yyyy'),
to_date('07/01/2024', 'dd/mm/yyyy'),
'0100112',
'98DBLSVT'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'YA8493569',
'Passaporto',
to_date('20/10/2029', 'dd/mm/yyyy'),
to_date('11/05/2019', 'dd/mm/yyyy'),
'0002234',
'80NREGNN'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
'CA94673GF',
'Carta identita',
to_date('11/10/2024', 'dd/mm/yyyy'),
to_date('10/07/2013', 'dd/mm/yyyy'),
```

```

'0002341',
'70SPSFBA'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
  'CK84933PP',
  'Carta identita',
  to_date('17/03/2026', 'dd/mm/yyyy'),
  to_date('23/07/2015', 'dd/mm/yyyy'),
  '0002341',
  '99RCCLNE'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
  'MI96054839',
  'Patente',
  to_date('31/01/2032', 'dd/mm/yyyy'),
  to_date('21/06/2022', 'dd/mm/yyyy'),
  '0001220',
  '34FNTGNN'
);

insert into
Documento(numero_documento,tipo_documento,scadenza,data_di_rilascio,codice_ente_ril
asciatario,CF_possessore)
values(
  'MI93040607',
  'Patente',
  to_date('22/04/2033','dd,mm,yyyy'),
  to_date('18/05/2023','dd,mm,yyyy'),
  '0010002',
  '59GLLFRC'
);

/*Ufficio Denunce*/
insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '001',
  'VTPPLL1012',
  'Filippo'
);

```

```
insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '002',
    'VTTPLL1012',
    'Giovanni'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '020',
    'VTTPLL1012',
    'Giuseppe'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '005',
    'VTTPLL1012',
    'Lorenzo'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '007',
    'VTTPLL1012',
    'Luca'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '006',
    'FDRSPS2020',
    'Martina'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '010',
    'FDRSPS2020',
    'Maria'
);
```

```
insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '009',
  'FDRSPS2020',
  'Lisa'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '012',
  'FDRSPS2020',
  'Paolo'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '014',
  'FDRSPS2020',
  'Giovanni'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '003',
  'MTTDCR1013',
  'Andrea'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '004',
  'MTTDCR1013',
  'Arianna'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
  '015',
  'MTTDCR1013',
  'Adriana'
);
```

```
insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '031',
    'MTTDCR1013',
    'Francesco'
);

insert into UfficioDenunce(numero, codice_tesserino_ctrl,
nome_dipendente_assegnato)
values(
    '033',
    'MTTDCR1013',
    'Francesca'
);

/*Denuncia*/
insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00001,
    '001',
    to_date('13/01/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00002,
    '020',
    to_date('16/01/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00003,
    '001',
    to_date('19/01/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00004,
    '007',
    to_date('04/03/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
```

```
00005,
'002',
to_date('20/05/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
00006,
'002',
to_date('22/08/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
00007,
'005',
to_date('10/10/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
00008,
'020',
to_date('15/12/2023', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
00009,
'006',
to_date('01/01/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
00010,
'020',
to_date('04/02/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
00011,
'010',
to_date('07/03/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
```

```
values(
    00012,
    '020',
    to_date('10/03/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00013,
    '009',
    to_date('12/04/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00014,
    '012',
    to_date('15/05/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00015,
    '014',
    to_date('18/06/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00016,
    '003',
    to_date('18/06/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00017,
    '004',
    to_date('25/07/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00018,
    '015',
    to_date('28/07/2024', 'dd,mm,yyyy')
);
```

```
insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00019,
    '031',
    to_date('28/07/2024', 'dd,mm,yyyy')
);

insert into Denuncia(numero_verbale_denuncia,numero_ufficio,data_emissione)
values(
    00020,
    '033',
    to_date('29/07/2024', 'dd,mm,yyyy')
);

/*Depone*/
insert into Depone(CF_vittima, numero_verbale_deposto)
values(
    '02LCTFRC',
    00001
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
    '02LCTFRC',
    00002
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
    '90GNVNTN',
    00003
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
    '01RRCPQL',
    00003
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
    '81STFMRC',
    00004
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
```

```
'81STFMRC',
00005
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'03INCNDR',
00006
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'88RRCNNA',
00007
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'92RSSGNN',
00008
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'98DBLSVT',
00009
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'88RRCNNA',
00010
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'92RSSGNN',
00011
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'01RRCPQL',
00011
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
```

```
'03INCNDR',
00012
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'03INCNDR',
00013
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'92RSSGNN',
00014
);

insert into Depone(CF_vittima, numero_verbale_deposto)
values(
'98DBLSVT',
00015
);

/*Riguarda*/
insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
'92RSSGNN',
00001
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
'78CMSFRN',
00002
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
'91MRTNTN',
00003
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
'99INNCRM',
00004
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
```

```
values(
    '66BRTPQL',
    00005
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '67BLNCTR',
    00006
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '78LSSBNT',
    00007
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '95LRAFCC',
    00008
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '78CMSFRN',
    00009
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '78LSSBNT',
    00010
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '99INNCRM',
    00011
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '95LRAFCC',
    00012
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
```

```
values(
    '67BLNCTR',
    00013
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '66BRTPQL',
    00014
);

insert into Riguarda(CF_imputato, numero_verbale_a_carico)
values(
    '91MRTNTN',
    00015
);

/*Controlla*/
insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'MTTGVN0010',
    '70SPSFBA',
    to_date('10/10/2022','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'LCUDDT0100',
    '70SPSFBA',
    to_date('05/09/2022','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'LCUDDT0100',
    '00BNCLSS',
    to_date('05/09/2022','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'LSESPS1211',
    '98DBLSVT',
    to_date('02/01/2023','dd/mm/yyyy')
);
```

```
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'LSESPS1211',
    '95LRAFCC',
    to_date('05/03/2023','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GBRBCC1002',
    '98DBLSVT',
    to_date('05/10/2023','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GBRBCC1002',
    '01RRCPQL',
    to_date('01/02/2023' , 'dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GNNSPS0021',
    '91MRTNTN',
    to_date('06/04/2023','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GNNSPS0021',
    '78LSSBNT',
    to_date('20/10/2023','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GRGMLT0091',
    '99INNCRM',
    to_date('10/04/2023','dd/mm/yyyy')
);
```

```
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GRGMLT0091',
    '91MRTNTN',
    to_date('21/11/2023','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'VLTBTT0024',
    '99INNCRM',
    to_date('12/01/2024','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'VLTBTT0024',
    '99INNCRM',
    to_date('15/05/2024','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'PLATSS0909',
    '99RCCLNE',
    to_date('13/01/2024','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GVNLCR0809',
    '70SPSFBA',
    to_date('23/12/2023','dd/mm/yyyy')
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'GVNLCR0809',
    '80NREGNN',
    to_date('23/12/2023','dd/mm/yyyy')
)
```

```
);

insert into Controlla(codice_tesserino_controllore, CF_persona_controllata,
data_controllo)
values(
    'VNCDCR1013',
    '80NREGNN',
    to_date('12/02/2024','dd/mm/yyyy')
);

/*Possiede*/

insert into Possiede(codice_seriale_posseduto, codice_tesserino_posessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000001KB',
    'LCUDDT0100',
    to_date('05/09/2022','dd/mm/yyyy'),
    18,
    6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_posessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000002KB',
    'LCUDDT0100',
    to_date('12/09/2022','dd/mm/yyyy'),
    14,
    6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_posessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000002KB',
    'MTTGVN0010',
    to_date('10/10/2022','dd/mm/yyyy'),
    7,
    6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_posessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000003KB',
    'GBRBCC1002',
    to_date('01/02/2023','dd/mm/yyyy'),
```

```
10,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'10000001KM',
'LSESPS1211',
to_date('02/01/2023','dd/mm/yyyy'),
14,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'20000003KS',
'GBRBCC1002',
to_date('01/02/2023','dd/mm/yyyy'),
10,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'20000003KS',
'LSESPS1211',
to_date('05/03/2023','dd/mm/yyyy'),
10,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'00000003KB',
'GNNSPS0021',
to_date('06/04/2023','dd/mm/yyyy'),
22,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'00000003KB',
```

```
'GNNSPS0021',
to_date('20/10/2023','dd/mm/yyyy'),
22,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'00000003KB',
'GRGMLT0091',
to_date('10/04/2023','dd/mm/yyyy'),
9,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'00000004KB',
'GRGMLT0091',
to_date('21/11/2023','dd/mm/yyyy'),
7,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'00000003KB',
'GVNLCR0809',
to_date('23/12/2023','dd/mm/yyyy'),
8,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
'00000005KB',
'GVNLCR0809',
to_date('23/12/2023','dd/mm/yyyy'),
22,
6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
```

```
values(
    '00000002KB',
    'VLNBT0024',
    to_date('12/01/2024','dd/mm/yyyy'),
    8,
    6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000002KB',
    'VLNBT0024',
    to_date('15/05/2024','dd/mm/yyyy'),
    8,
    6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000004KB',
    'PLATSS0909',
    to_date('13/01/2024','dd/mm/yyyy'),
    8,
    6
);

insert into Possiede(codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
values(
    '00000004KB',
    'VNCDCR1013',
    to_date('12/02/2024','dd/mm/yyyy'),
    22,
    6
);
```

## Viste

Per una maggiore leggibilità dei dati, sono state create delle VIEW per controllare velocemente alcune informazioni specifiche.

### **Lista di tutte le denunce prelevate nell'ultimo mese**

```
CREATE OR REPLACE VIEW DenunceUltimoMese AS
SELECT *
FROM Denuncia
WHERE data_emissione >= ADD_MONTHS(SYSDATE, -1);
```

### **Lista di tutte le persone imputate**

```
CREATE OR REPLACE VIEW VistaImputati AS
SELECT CF, nome, cognome, data_nascita, telefono, id_imputato
FROM Persona
WHERE id_imputato IS NOT NULL;
```

### **Lista di tutte le persone vittime**

```
CREATE OR REPLACE VIEW VistaVittime AS
SELECT CF, nome, cognome, data_nascita, telefono, id_vittima
FROM Persona
WHERE id_vittima IS NOT NULL;
```

### **Lista di tutte le multe effettuate negli ultimi tre mesi**

```
CREATE OR REPLACE VIEW MulteUltimiTreMesi AS
SELECT *
FROM Multa
WHERE data_emissione_multa >= ADD_MONTHS(SYSDATE, -3);
```

## Triggers

A seguito della creazione di vincoli dinamici sono stati implementati i seguenti triggers:

### Vincolo di integrità temporale per la data di emissione della multa

```
CREATE OR REPLACE TRIGGER verifica_data_multa
BEFORE INSERT OR UPDATE ON Multa
FOR EACH ROW
BEGIN
    IF :NEW.data_emissione_multa > SYSDATE THEN
        RAISE_APPLICATION_ERROR(-20001, 'La data di emissione della multa non può
essere futura.');
    END IF;
END;
```

### Vincolo di controllo per il numero massimo di ore settimanali per un poliziotto

```
CREATE OR REPLACE TRIGGER verifica_ore_turno_settimanali
BEFORE INSERT OR UPDATE ON Turno
FOR EACH ROW
DECLARE
    ore_totali NUMBER;
BEGIN
    SELECT SUM(ore) INTO ore_totali
    FROM Turno
    WHERE codice_tesserino_turnante = :NEW.codice_tesserino_turnante
        AND data_turno BETWEEN TRUNC(:NEW.data_turno, 'IW') AND
        TRUNC(:NEW.data_turno, 'IW') + 6; -- Calcolo della settimana

    IF ore_totali > 36 THEN -- Limite settimanale di 36 ore
        RAISE_APPLICATION_ERROR(-20003, 'Il numero massimo di ore di turno
settimanali per un poliziotto è 36.');
    END IF;
END;
```

### Vincolo di integrità per la scadenza del documento

```
CREATE OR REPLACE TRIGGER verifica_scadenza_documento
BEFORE INSERT OR UPDATE ON Documento
FOR EACH ROW
BEGIN
    IF :NEW.scadenza < TRUNC(SYSDATE) THEN
        RAISE_APPLICATION_ERROR(-20004, 'Non è possibile inserire o modificare un
documento scaduto.');
    END IF;
END;
```

## Vincolo di integrità per l'utilizzo di un'attrezzatura già presa

```

CREATE OR REPLACE TRIGGER verifica_utilizzo_attrezzatura
BEFORE INSERT OR UPDATE ON Possiede
FOR EACH ROW
DECLARE
    attrezzatura_in_uso NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO attrezzatura_in_uso
    FROM Possiede
    WHERE codice_seriale_posseduto = :NEW.codice_seriale_posseduto
        AND (:NEW.data_utilizzo = data_utilizzo)
        AND (
            (:NEW.orario_inizio BETWEEN orario_inizio AND (orario_inizio +
ore_di_acquisizione - 1))
            OR ((:NEW.orario_inizio + :NEW.ore_di_acquisizione - 1) BETWEEN
orario_inizio AND (orario_inizio + ore_di_acquisizione - 1))
            OR (orario_inizio BETWEEN :NEW.orario_inizio AND (:NEW.orario_inizio +
:NEW.ore_di_acquisizione - 1)))
        );
    IF attrezzatura_in_uso > 0 THEN
        RAISE_APPLICATION_ERROR(-20008, 'L''attrezzatura è già in uso da un altro
poliziotto.');
    END IF;
END;

```

## Vincolo di integrità per l'assegnazione di un poliziotto nell'ufficio denunce

```

CREATE OR REPLACE TRIGGER verifica_poliziotto_assegnato_ufficio
BEFORE INSERT ON Denuncia
FOR EACH ROW
DECLARE
    poliziotto_assegnato NUMBER;
BEGIN
    SELECT COUNT(*) INTO poliziotto_assegnato
    FROM UfficioDenunce
    WHERE numero = :NEW.numero_ufficio
        AND codice_tesserino_ctrl IS NOT NULL;

    IF poliziotto_assegnato = 0 THEN
        RAISE_APPLICATION_ERROR(-20015, 'Non è possibile registrare una denuncia in
un ufficio senza un poliziotto assegnato.');
    END IF;
END;

```

## Vincolo di integrità per l'assegnazione di un turno ad un poliziotto già impegnato

```
CREATE OR REPLACE TRIGGER check_poliziotto_turno
BEFORE INSERT OR UPDATE ON Turno
FOR EACH ROW
DECLARE
    v_count NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO v_count
    FROM Turno
    WHERE codice_tesserino_turnante = :NEW.codice_tesserino_turnante
        AND data_turno = :NEW.data_turno
        AND (
            (:NEW.ore BETWEEN 1 AND 24) -- Assicurarsi che l'ora sia valida
            OR
            (:NEW.ore + (SELECT NVL(SUM(ore), 0) FROM Turno WHERE
codice_tesserino_turnante = :NEW.codice_tesserino_turnante AND data_turno =
:NEW.data_turno) <= 24)
        );
    IF v_count > 0 THEN
        RAISE_APPLICATION_ERROR(-20001, 'Il poliziotto ha già un turno assegnato
per queste ore nella stessa giornata.');
    END IF;
END;
```

## Procedure

Le procedure riportate di seguito sono le stesse descritte nel capitolo 1.6 Utenti, permessi e operazioni

### Assumi Poliziotto

```

CREATE OR REPLACE PROCEDURE assumiPoliziotto(
    p_codice_tesserino IN Poliziotto.codice_tesserino%TYPE,
    p_nome_poliziotto IN Poliziotto.nome_poliziotto%TYPE,
    p_cognome_poliziotto IN Poliziotto.cognome_poliziotto%TYPE,
    p_CF_poliziotto IN Poliziotto.CF_poliziotto%TYPE
) IS
BEGIN

    IF p_codice_tesserino IS NULL OR p_nome_poliziotto IS NULL OR
        p_cognome_poliziotto IS NULL OR p_CF_poliziotto IS NULL THEN
        RAISE_APPLICATION_ERROR(-20000, 'Tutti i dati devono essere forniti.');
    END IF;

    IF NOT REGEXP_LIKE(p_CF_poliziotto, '^\\d{2}[A-Z]{6}$') THEN
        RAISE_APPLICATION_ERROR(-20001, 'Codice Fiscale del Poliziotto non
valido.');
    END IF;

    IF NOT REGEXP_LIKE(p_codice_tesserino, '^[A-Z]{6}\\d{4}$') THEN
        RAISE_APPLICATION_ERROR(-20002, 'Codice Tesserino non valido.');
    END IF;

    INSERT INTO Poliziotto (codice_tesserino, nome_poliziotto, cognome_poliziotto,
    CF_poliziotto)
    VALUES (p_codice_tesserino, p_nome_poliziotto, p_cognome_poliziotto,
    p_CF_poliziotto);

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Poliziotto inserito con successo.');
EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN

        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Violazione della chiave unica o primaria: il codice
tesserino o il CF è già presente.');
        RAISE_APPLICATION_ERROR(-20003, 'Violazione della chiave unica o primaria:
il codice tesserino o il CF è già presente.');

```

```

WHEN VALUE_ERROR THEN

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Errore di conversione di tipo nei valori forniti.');
    RAISE_APPLICATION_ERROR(-20004, 'Errore di conversione di tipo nei valori
forniti.');
END assumiPoliziotto;

```

## Crea Turno

```

CREATE OR REPLACE PROCEDURE creaTurno(
    p_codice_tesserino_turnante IN Turno.codice_tesserino_turnante%TYPE,
    p_data_turno IN Turno.data_turno%TYPE,
    p_ora_inizio IN Turno.ora_inizio%TYPE,
    p_ore IN Turno.ore%TYPE,
    p_sede IN Turno.sede%TYPE
) IS
BEGIN

    IF p_codice_tesserino_turnante IS NULL OR p_data_turno IS NULL OR
        p_ora_inizio IS NULL OR p_ore IS NULL OR p_sede IS NULL THEN
        RAISE_APPLICATION_ERROR(-20000, 'Tutti i dati devono essere forniti.');
    END IF;

    IF NOT REGEXP_LIKE(p_codice_tesserino_turnante, '^[A-Z]{6}\d{4}$') THEN
        RAISE_APPLICATION_ERROR(-20001, 'Codice Tesserino non valido.');
    END IF;

    IF p_ora_inizio < 0 OR p_ora_inizio > 23 OR p_ore <= 0 OR p_ora_inizio + p_ore
    > 24 THEN
        RAISE_APPLICATION_ERROR(-20002, 'Ora di inizio o durata del turno non
valida.');
    END IF;

    INSERT INTO Turno (codice_tesserino_turnante, data_turno, ora_inizio, ore,
sede)
        VALUES (p_codice_tesserino_turnante, p_data_turno, p_ora_inizio, p_ore,
p_sede);

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Turno inserito con successo.');
EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN

```

```

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Violazione della chiave unica o primaria: il turno è
già presente.');
    RAISE_APPLICATION_ERROR(-20003, 'Violazione della chiave unica o primaria:
il turno è già presente.');
WHEN VALUE_ERROR THEN

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Errore di conversione di tipo nei valori forniti.');
    RAISE_APPLICATION_ERROR(-20004, 'Errore di conversione di tipo nei valori
forniti.');
END creaTurno;

```

## Rivendica Armi

```

CREATE OR REPLACE PROCEDURE rivendicaArmi(
    p_codice_arma IN Arma.codice_arma%TYPE,
    p_tipo IN Arma.tipo%TYPE,
    p_modello IN Arma.modello%TYPE,
    p_codice_tesserino_arma IN Arma.codice_tesserino_arma%TYPE
) IS
BEGIN

    IF p_codice_arma IS NULL OR p_tipo IS NULL OR
    p_modello IS NULL OR p_codice_tesserino_arma IS NULL THEN
        RAISE_APPLICATION_ERROR(-20000, 'Tutti i dati devono essere forniti.');
    END IF;

    IF NOT REGEXP_LIKE(p_codice_arma, '^\\d{8}[A-Z]{2}$') THEN
        RAISE_APPLICATION_ERROR(-20001, 'Codice Arma non valido.');
    END IF;

    IF NOT REGEXP_LIKE(p_codice_tesserino_arma, '^[A-Z]{6}\\d{4}$') THEN
        RAISE_APPLICATION_ERROR(-20002, 'Codice Tesserino non valido.');
    END IF;

    INSERT INTO Arma (codice_arma, tipo, modello, codice_tesserino_arma)
VALUES (p_codice_arma, p_tipo, p_modello, p_codice_tesserino_arma);

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Arma inserita con successo.');
EXCEPTION

```

```

WHEN DUP_VAL_ON_INDEX THEN

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Violazione della chiave unica o primaria: il codice
arma è già presente.');
    RAISE_APPLICATION_ERROR(-20003, 'Violazione della chiave unica o primaria:
il codice arma è già presente.');

WHEN VALUE_ERROR THEN

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Errore di conversione di tipo nei valori forniti.');
    RAISE_APPLICATION_ERROR(-20004, 'Errore di conversione di tipo nei valori
forniti.');
END rivendicaArmi;

```

## Rivendica Attrezzatura

```

CREATE OR REPLACE PROCEDURE rivendicaAttrezzatura(
    p_codice_seriale IN Attrezzatura.codice_seriale%TYPE,
    p_tipo_attrezzatura IN Attrezzatura.tipo_attrezzatura%TYPE
) IS
BEGIN

    IF p_codice_seriale IS NULL OR p_tipo_attrezzatura IS NULL THEN
        RAISE_APPLICATION_ERROR(-20000, 'Tutti i dati devono essere forniti.');
    END IF;

    IF NOT REGEXP_LIKE(p_codice_seriale, '^\\d{8}[A-Z]{2}$') THEN
        RAISE_APPLICATION_ERROR(-20001, 'Codice Seriale non valido.');
    END IF;

    IF NOT p_tipo_attrezzatura IN ('Kit base', 'Kit medio', 'Kit operazioni
speciali') THEN
        RAISE_APPLICATION_ERROR(-20002, 'Tipo di attrezzatura non valido.');
    END IF;

    INSERT INTO Attrezzatura (codice_seriale, tipo_attrezzatura)
    VALUES (p_codice_seriale, p_tipo_attrezzatura);

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Attrezzatura registrata con successo.');
EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN

```

```

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Violazione della chiave unica o primaria: il codice
seriale è già presente.');
    RAISE_APPLICATION_ERROR(-20003, 'Violazione della chiave unica o primaria:
il codice seriale è già presente.');
WHEN VALUE_ERROR THEN

    ROLLBACK;
    DBMS_OUTPUT.PUT_LINE('Errore di conversione di tipo nei valori forniti.');
    RAISE_APPLICATION_ERROR(-20004, 'Errore di conversione di tipo nei valori
forniti.');
END rivendicaAttrezzatura;

```

## Assegna Attrezzatura

```

CREATE OR REPLACE PROCEDURE assegnaAttrezzatura(
    p_codice_seriale IN Attrezzatura.codice_seriale%TYPE,
    p_codice_tesserino IN Poliziotto.codice_tesserino%TYPE,
    p_data_utilizzo IN Possiede.data_utilizzo%TYPE,
    p_orario_inizio IN Possiede.orario_inizio%TYPE,
    p_ore_di_acquisizione IN Possiede.ore_di_acquisizione%TYPE
) IS
    v_attrezzatura_count NUMBER;
    v_poliziotto_count NUMBER;
BEGIN

    IF p_codice_seriale IS NULL OR p_codice_tesserino IS NULL OR
        p_data_utilizzo IS NULL OR p_orario_inizio IS NULL OR p_ore_di_acquisizione
IS NULL THEN
        RAISE_APPLICATION_ERROR(-20000, 'Tutti i dati devono essere forniti.');
    END IF;

    SELECT COUNT(*)
    INTO v_attrezzatura_count
    FROM Attrezzatura
    WHERE codice_seriale = p_codice_seriale;

    IF v_attrezzatura_count = 0 THEN
        RAISE_APPLICATION_ERROR(-20005, 'Attrezzatura non registrata nel
sistema.');
    END IF;

    SELECT COUNT(*)
    INTO v_poliziotto_count
    FROM Poliziotto
    WHERE codice_tesserino = p_codice_tesserino;

```

```

IF v_poliziotto_count = 0 THEN
    RAISE_APPLICATION_ERROR(-20006, 'Poliziotto non trovato nel sistema.');
END IF;

INSERT INTO Possiede (codice_seriale_posseduto, codice_tesserino_possessore,
data_utilizzo, orario_inizio, ore_di_acquisizione)
VALUES (p_codice_seriale, p_codice_tesserino, p_data_utilizzo, p_orario_inizio,
p_ore_di_acquisizione);

COMMIT;

DBMS_OUTPUT.PUT_LINE('Attrezzatura assegnata con successo.');

EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Violazione della chiave unica o primaria:
l''assegnazione esiste già.');
        RAISE_APPLICATION_ERROR(-20007, 'Violazione della chiave unica o primaria:
l''assegnazione esiste già.');
    WHEN VALUE_ERROR THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore di conversione di tipo nei valori forniti.');
        RAISE_APPLICATION_ERROR(-20008, 'Errore di conversione di tipo nei valori
forniti.');
END assegnaAttrezzatura;

```

## Inserisci Persona

```

CREATE OR REPLACE PROCEDURE InserisciPersona (
    p_CF IN VARCHAR2,
    p_nome IN VARCHAR2,
    p_cognome IN VARCHAR2,
    p_data_nascita IN DATE,
    p_telefono IN VARCHAR2,
    p_id_vittima IN NUMBER,
    p_id_imputato IN NUMBER
)
IS
    v_data_nascita VARCHAR2(10);
BEGIN

    v_data_nascita := TO_CHAR(p_data_nascita, 'DD/MM/YYYY');

```

```

    INSERT INTO Persona (CF, nome, cognome, data_nascita, telefono, id_vittima,
id_imputato)
    VALUES (p_CF, p_nome, p_cognome, TO_DATE(v_data_nascita, 'DD/MM/YYYY'),
p_telefono, p_id_vittima, p_id_imputato);

```

```

    COMMIT;
    DBMS_OUTPUT.PUT_LINE('Persona inserita con successo');
EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore: Violazione di un vincolo di chiave unica.');
    WHEN OTHERS THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore generico durante l''inserimento della
persona.');
        DBMS_OUTPUT.PUT_LINE(SQLERRM);
END InserisciPersona;

```

## Preleva Denuncia

```

CREATE OR REPLACE PROCEDURE prelevaDenuncia (
    p_numero_verbale IN Denuncia.numero_verbale_denuncia%TYPE,
    p_numero_ufficio IN Denuncia.numero_ufficio%TYPE,
    p_data_emissione IN DATE,
    p_id_vittima_1 IN Persona.id_vittima%TYPE,
    p_id_vittima_2 IN Persona.id_vittima%TYPE DEFAULT NULL,
    p_id_imputato IN Persona.id_imputato%TYPE DEFAULT NULL
)
IS
    v_numero_vittima NUMBER;
    v_numero_imputato NUMBER;
    v_cf_imputato Persona.CF%TYPE;
BEGIN

    SELECT COUNT(*)
    INTO v_numero_vittima
    FROM Persona
    WHERE id_vittima = p_id_vittima_1;

    IF v_numero_vittima = 0 THEN
        INSERT INTO Persona (CF, nome, cognome, data_nascita, telefono, id_vittima)
        VALUES ('CF_' || TO_CHAR(SYSDATE, 'YYYYMMDDHH24MISS'), 'Nome Vittima 1',
        'Cognome Vittima 1', SYSDATE, '0000000000', p_id_vittima_1);
    END IF;


```

```

IF p_id_vittima_2 IS NOT NULL THEN
    SELECT COUNT(*)
    INTO v_numero_vittima
    FROM Persona
    WHERE id_vittima = p_id_vittima_2;

```

```

IF v_numero_vittima = 0 THEN
    INSERT INTO Persona (CF, nome, cognome, data_nascita, telefono,
id_vittima)
        VALUES ('CF_' || TO_CHAR(SYSDATE, 'YYYYMMDDHH24MISS'), 'Nome Vittima
2', 'Cognome Vittima 2', SYSDATE, '0000000000', p_id_vittima_2);
    END IF;
END IF;

IF p_id_imputato IS NULL THEN
    v_cf_imputato := '11AAAAAA';

    SELECT COUNT(*)
    INTO v_numero_imputato
    FROM Persona
    WHERE CF = v_cf_imputato;

    IF v_numero_imputato = 0 THEN
        INSERT INTO Persona (CF, nome, cognome, data_nascita, telefono)
        VALUES (v_cf_imputato, 'Imputato Ignoto', 'N/A', DATE '1900-01-01',
'0000000000');
    END IF;
ELSE

    SELECT COUNT(*)
    INTO v_numero_imputato
    FROM Persona
    WHERE id_imputato = p_id_imputato;

    IF v_numero_imputato = 0 THEN
        v_cf_imputato := 'CF_' || TO_CHAR(SYSDATE, 'YYYYMMDDHH24MISS');
        INSERT INTO Persona (CF, nome, cognome, data_nascita, telefono,
id_imputato)
            VALUES (v_cf_imputato, 'Nome Imputato', 'Cognome Imputato', SYSDATE,
'0000000000', p_id_imputato);
    ELSE
        SELECT CF
        INTO v_cf_imputato
        FROM Persona
        WHERE id_imputato = p_id_imputato;
    END IF;
END IF;

INSERT INTO Denuncia (numero_verbale_denuncia, numero_ufficio, data_emissione)
VALUES (p_numero_verbale, p_numero_ufficio, p_data_emissione);

INSERT INTO Depone (CF_vittima, numero_verbale_deposto)

```

```

        VALUES ((SELECT CF FROM Persona WHERE id_vittima = p_id_vittima_1),
p_numero_verbale);

        IF p_id_vittima_2 IS NOT NULL THEN
            INSERT INTO Depone (CF_vittima, numero_verbale_deposto)
            VALUES ((SELECT CF FROM Persona WHERE id_vittima = p_id_vittima_2),
p_numero_verbale);
        END IF;

        INSERT INTO Riguarda (CF_imputato, numero_verbale_a_carico)
        VALUES (v_cf_imputato, p_numero_verbale);

        COMMIT;

        DBMS_OUTPUT.PUT_LINE('Denuncia inserita con successo');

EXCEPTION
    WHEN DUP_VAL_ON_INDEX THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore: Violazione di un vincolo di chiave unica.');
    WHEN NO_DATA_FOUND THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore: Nessun dato trovato.');
    WHEN OTHERS THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore generico durante l''inserimento della
denuncia.');
        DBMS_OUTPUT.PUT_LINE(SQLERRM);
END prelevaDenuncia;

```

## Definisci Multa

```

CREATE OR REPLACE PROCEDURE definisciMultta (
    p_numero_verbale_multa IN VARCHAR2,
    p_data_emissione_multa IN VARCHAR2,
    p_costo IN NUMBER,
    p_codice_tesserino_multante IN VARCHAR2,
    p_targa_multata IN CHAR
)
AS
    v_data_emissione DATE;
BEGIN

    v_data_emissione := TO_DATE(p_data_emissione_multa, 'dd/mm/yyyy');

    INSERT INTO Multa (numero_verbale_multa, data_emissione_multa, costo,
codice_tesserino_multante, targa_multata)
    VALUES (p_numero_verbale_multa, v_data_emissione, p_costo,
p_codice_tesserino_multante, p_targa_multata);

```

```

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Multa inserita correttamente.');
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore durante l''inserimento della multa: ' ||
SQLERRM);
END definisciMulta;

```

## Definisci Veicolo Esaminato

```

CREATE OR REPLACE PROCEDURE definisciVeicoloEsaminato (
    p_targa IN CHAR,
    p_modello_veicolo IN VARCHAR2,
    p_tipo_di_alimentazione IN VARCHAR2,
    p_codice_telaio IN CHAR,
    p_CF_intestatario IN CHAR
)
AS
BEGIN
    INSERT INTO VeicoloEsaminato (targa, modello_veicolo, tipo_di_alimentazione,
codice_telaio, CF_intestatario)
    VALUES (p_targa, p_modello_veicolo, p_tipo_di_alimentazione, p_codice_telaio,
p_CF_intestatario);

    COMMIT;

    DBMS_OUTPUT.PUT_LINE('Veicolo esaminato inserito correttamente.');
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Errore durante l''inserimento del veicolo esaminato: ' ||
SQLERRM);
END definisciVeicoloEsaminato;

```

## Registra Documento

```

CREATE OR REPLACE PROCEDURE registraDocumento (
    p_numero_documento IN VARCHAR2,
    p_tipo_documento IN VARCHAR2,
    p_scadenza IN DATE,
    p_data_di_rilascio IN DATE,
    p_codice_ente_rilaschiatore IN CHAR,
    p_CF_posessore IN CHAR

```

```
)  
AS  
BEGIN  
  
    INSERT INTO Documento (numero_documento, tipo_documento, scadenza,  
data_di_rilascio, codice_ente_rilasciatario, CF_posessore)  
        VALUES (p_numero_documento, p_tipo_documento, p_scadenza,  
p_data_di_rilascio, p_codice_ente_rilasciatore, p_CF_posessore);  
  
    COMMIT;  
  
    DBMS_OUTPUT.PUT_LINE('Documento registrato correttamente.');
```

EXCEPTION

```
    WHEN OTHERS THEN  
        ROLLBACK;  
        DBMS_OUTPUT.PUT_LINE('Errore durante la registrazione del documento: '  
|| SQLERRM);  
    END registraDocumento;
```