



Università degli Studi di Napoli
“Parthenope”

Dipartimento di Scienze e Tecnologie

Corso di Laurea di Informatica

Anno Accademico: 2020/2021

RELAZIONE DEL PROGETTO DI “BASI DI DATI”

Database Nazionale Vaccinazioni Covid19

Studenti: Mariano Aponte, Francesco Calcopietro

Matricole: 0124002088, 0124002090

Professore: Antonio Maratea

Indice

1. Cenno storico del contesto.....	5
2. Requisiti.....	5
3. Glossario.....	6
4. Diagramma EE-R.....	7
5. Diagramma Relazionale.....	8
5.1 Considerazioni Diagramma Relazionale.....	9
6. Utenti.....	9
6.1 Operazioni degli utenti.....	10
7. Volumi.....	13
8. Vincoli di integrità.....	14
9. Verifica di normalità.....	15
10. Implementazione.....	16
10.1 Creazione degli utenti.....	16
10.2 Data Definition Language.....	18
10.2.1 Vaccinando.....	18
10.2.2 Prenotazione.....	19
10.2.3 Fragile.....	20
10.2.4 Disabile.....	20
10.2.5 Disabilità.....	20
10.2.6 Patologia.....	21
10.2.7 Telefono.....	21
10.2.8 Operatore Sanitario Responsabile.....	22
10.2.9 Anamnesi.....	22
10.2.10 Non Idoneo Temporaneo.....	23
10.2.11 Ideoneo.....	23

10.2.12	Non Idoneo Permanente.....	24
10.2.13	Card Vaccinato.....	24
10.2.14	Stabilimento.....	24
10.2.15	Casa Produttrice.....	25
10.2.16	Luogo.....	25
10.2.17	Lotto Vaccino.....	26
10.2.18	Produttore di.....	26
10.2.19	Vaccinazione.....	27
10.2.20	Consegnato a.....	28
10.3	Data Manipulation Language.....	29
10.3.1	Prenotazione.....	29
10.3.2	Vaccinando.....	33
10.3.3	Fragile.....	40
10.3.4	Patologia.....	41
10.3.5	Disabile.....	43
10.3.6	Disabilità.....	43
10.3.7	Telefono.....	44
10.3.8	Operatore Sanitario Responsabile.....	45
10.3.9	Anamnesi.....	46
10.3.10	Idoneo.....	50
10.3.11	Non Ideoneo Temporaneo.....	51
10.3.12	Non Idoneo Permanente.....	53
10.3.13	Card Vaccinato.....	53
10.3.14	Luogo.....	55
10.3.15	Stabilimento.....	56
10.3.16	Casa Produttrice.....	57

10.3.17	Lotto Vaccino.....	58
10.3.18	Produttore di.....	60
10.3.19	Consegnato a.....	62
10.3.20	Vaccinazione.....	63
10.4	Trigger.....	66
10.4.1	Check_Age_over18.....	67
10.4.2	Check_Prenotazione_Anamnesi_Date.....	67
10.4.3	Check_Vaccinazione_Anamnesi_Date.....	68
10.4.4	CheckNonIdoneoPerm_Cant_Be_Vaccinated.....	70
10.4.5	Check_NonIdeoneoTemp_Cant_Be_Vaccinated.....	71
10.4.6	Check_Seconda_Dose_Date.....	72
10.4.7	Check_Consegna_Vaccini.....	74
10.4.8	Vaccino_Scaduto.....	75
10.5	Procedure.....	76
10.5.1	create_Vaccinando.....	76
10.5.2	createAnamnesi.....	81
10.5.3	insertVaccino.....	86
10.5.4	consegnaVaccino.....	88
10.5.5	createVaccinazione.....	94

1. Cenno storico del contesto

La Covid-19, conosciuta anche come malattia respiratoria acuta da SARS-CoV-2 o Malattia da coronavirus 2019, è una malattia infettiva respiratoria causata dal virus denominato SARS-CoV-2 appartenente alla famiglia dei coronavirus. I primi casi sono stati registrati in Cina alla fine del 2019. A seguito di migliaia di casi accertati e decessi, tale virus è stato denominato dall'OMS (Organizzazione Mondiale della Sanità) come pandemia il 30 gennaio del 2020.

2. Requisiti

Si vuole progettare un database per la campagna vaccinale contro la Covid-19. In particolare si vuole gestire tutto il processo della vaccinazione, a partire dalla prenotazione di un vaccinando, passando per l'anamnesi per poi, a seguito dell'esito di quest'ultima, arrivare all'operazione di vaccinazione.

Del vaccinando si mantengono tutte le sue informazioni anagrafiche, eventuale positività pregressa e la categoria speciale a cui potrebbe appartenere (disabile o fragile). In particolare, se il vaccinando è disabile si vuole sapere anche il tipo di disabilità e il numero di protocollo dell'attestato certificante la Legge 104. Dei vaccinandoli fragili si vuole conoscere solo il tipo di patologia.

Come detto, il vaccinando per potersi vaccinare deve prima di tutto prenotarsi connettendosi al database. Della prenotazione si vuole sapere la data e ora prevista della vaccinazione, il tipo di prenotazione (singola o doppia dose) e il luogo di vaccinazione. Siccome possono esserci delle rimodulazioni della prenotazione, la singola prenotazione è identificata univocamente dal numero di protocollo e dal numero di tentativo.

Fatta la prenotazione, un'altra operazione preliminare che deve effettuare il vaccinando è l'anamnesi. Essa è fondamentale per poter stabilire se il vaccinando è idoneo o meno per effettuare la vaccinazione. In particolare, gli esiti di tale anamnesi sono: idoneo, non idoneo temporaneo cioè il paziente al momento non può sostenere la vaccinazione ma in futuro, rifacendo l'anamnesi in seguito alla rimodulazione automatica della prenotazione fatta da un medico, potrebbe risultare idoneo e sottoporsi alla vaccinazione. Un altro esito è non idoneo permanente. In questo caso il paziente non potrà mai vaccinarsi. Dell'anamnesi si vuole conoscere quando è stata effettuata e l'operatore sanitario responsabile che l'ha sostenuta.

Quest'ultimo è identificato dal numero Iscrizione Ordine dei Medici.

Il vaccinando idoneo può quindi sostenere la vaccinazione. Di tale operazione si vuole conoscere data e ora effettiva in cui viene sostenuta, braccio di inoculazione, luogo dove è effettuata, il vaccino usato.

Del luogo di vaccinazione si vuole sapere: nome del centro, indirizzo, distretto e asl di appartenenza. Il luogo può essere un PVO o un PVP. Dei PVO si vuole conoscere, oltre a quanto già detto del luogo generico, anche il codice presidio ospedaliero.

Del vaccino utilizzato si vuole conoscere: il nome del farmaco, numero lotto, la scadenza, numero dosi del lotto, il tipo di vaccino (singola o doppia dose), la casa produttrice che l'ha ideata e lo stabilimento specifico che ha prodotto quel vaccino.

Della casa produttrice si memorizza nome, sito web e nazione, mentre di stabilimento: nome, indirizzo e nazione.

Un paziente vaccinato riceve una card che attesta la vaccinazione. Di essa si tiene conto del numero di carta, che la identifica, data di attivazione e data di scadenza.

3. Glossario

Termine	Definizione	Sinonimi
Anamnesi	Controllo medico completo effettuato precedentemente alla vaccinazione per verificare l'idoneità di un paziente.	Diagnosi, prognosi
PVP	Presidio Vaccinale Popolare. Struttura adibita temporaneamente per la vaccinazione.	Hub vaccinale
PVO	Presidio Vaccinale Ospedaliero. Ospedale adibito per la vaccinazione.	Ospedale

Tabella 1: Glossario

4. Diagramma EE/R

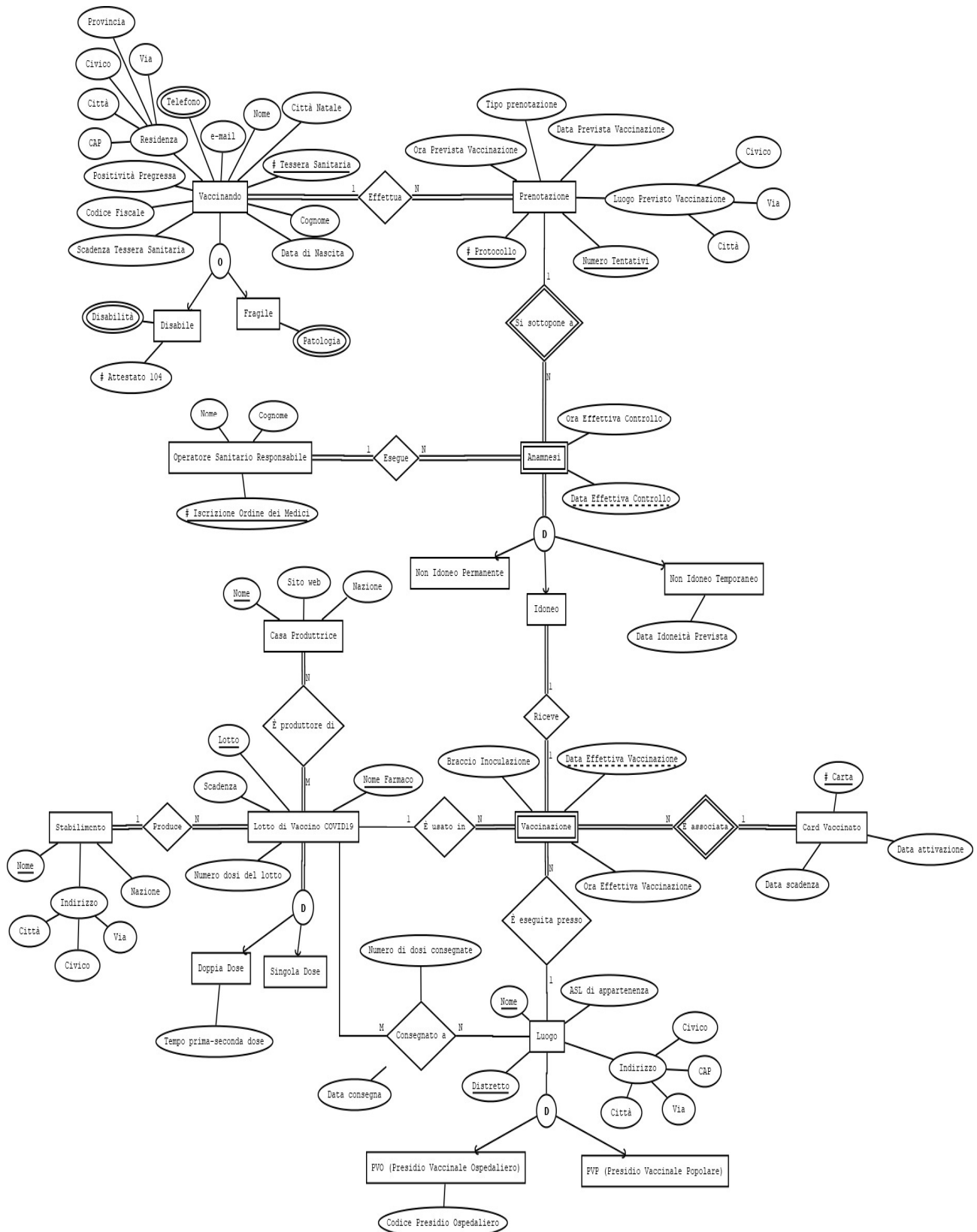


Figura 1: Diagramma EE-R

5. Diagramma relazionale

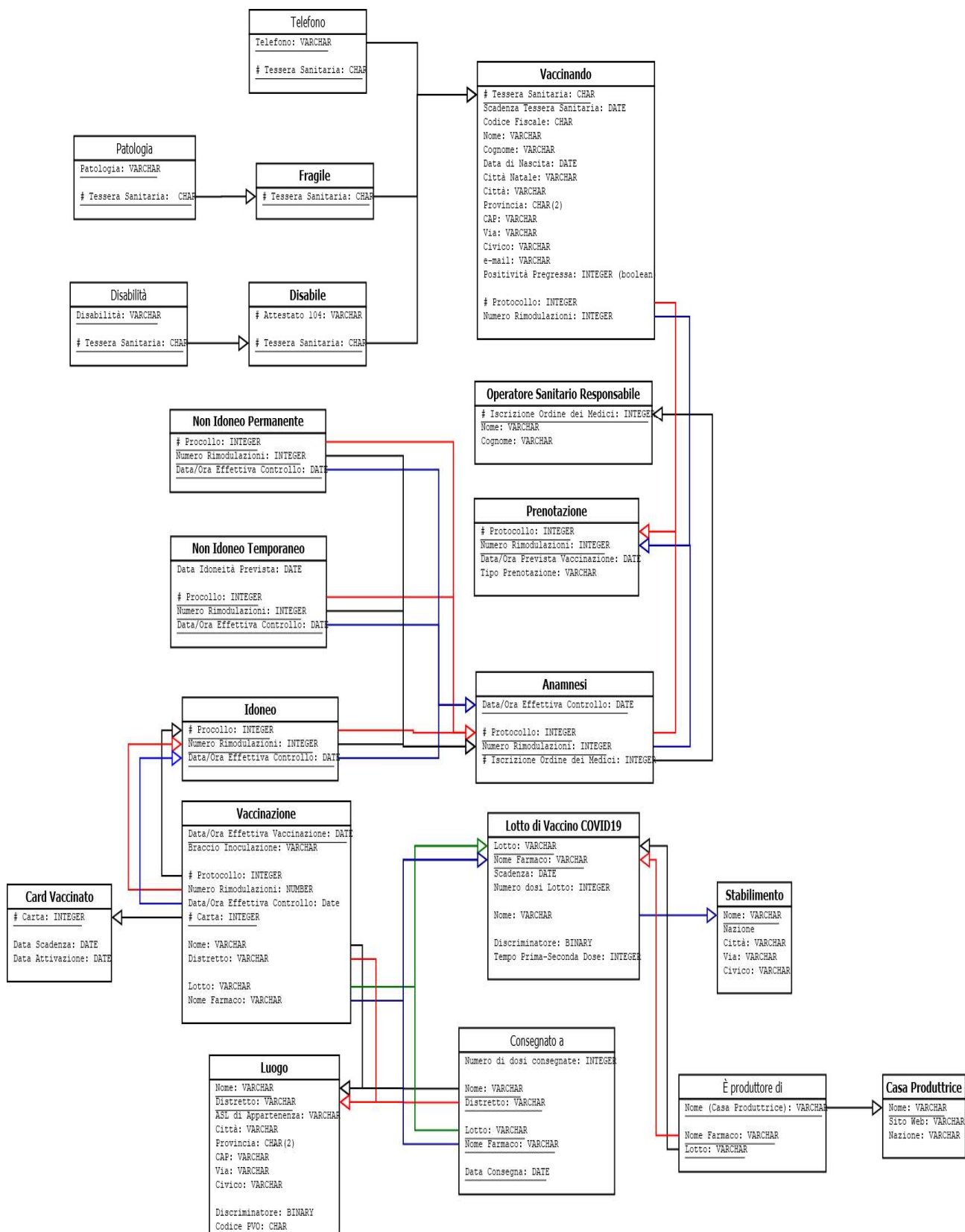


Figura 2: Modello Relazionale

5.1 Considerazioni Diagramma relazione

Dall'analisi del modello relazionale si nota come nella rappresentazione della specializzazione di vaccinando si usufruisce della tecnica del partizionamento verticale per il fatto che le classi figlie possiedono degli ulteriori attributi non in comune e anche per il fatto che la classe madre ha un'associazione con un'ulteriore entità. Stesso discorso per la specializzazione di anamnesi. In questo caso la motivazione è dovuta dal fatto che una delle classi figlie (idoneo) ha un'associazione con un'ulteriore entità. Utilizzare una tabella unica con discriminatore avrebbe portato grandi incongruenze in quanto la foreign key di vaccinazione avrebbe coinvolto tutte le t-uple di anamnesi, cosa inaccettabile in quanto, come già detto, non tutti i pazienti possono vaccinarsi.

Discorso diverso per la specializzazione di luogo e di lotto. In questi casi le classi figlie hanno pochi o 0 attributi e non sono coinvolti in ulteriori associazioni. Solo le loro classi madri lo sono. Per tale motivo si decide di usufruire di tabelle uniche con discriminatori per distinguere le varie istanze delle varie entità.

Sono presenti delle entità deboli: anamnesi e vaccinazione. Lo sono in quanto vengono rispettate le 3 caratteristiche di tale entità:

- Relazione di dipendenza
- Molteplicità 1:N (multivalore)
- Presenza di chiave debole sull'entità che ha molteplicità=N. Tale chiave, concatenata con la chiave dell'attributo forte, forma una chiave multi attributo.

Tutto ciò lo si può notare anche nella realtà: un'anamnesi non esiste se non esiste una prenotazione. L'azione della vaccinazione non è identificata se non con la card che viene rilasciata al vaccinando.

6. Utenti

Tale database potrà essere usufruito da 3 diverse tipologie di utenti, i quali sono definiti dalla seguente tabella:

Utente	Tipo	Permessi
Admin	Amministratore	All

Autorità sanitario	Comune	EXECUTE ON CONSEGNAVACCINO; EXECUTE ON INSERTVACCINO; SELECT, DELETE, UPDATE ON CONSEGNATO_A; SELECT, INSERT, DELETE, UPDATE ON LUOGO; SELECT, DELETE, UPDATE ON LOTTO_VACCINO; SELECT, INSERT, DELETE, UPDATE ON STABILIMENTO; SELECT, INSERT, DELETE, UPDATE ON CASA_PRODUTTRICE; SELECT, DELETE, UPDATE ON PRODUTTORE_DI;
Medico	Comune	EXECUTE ON CREATEANAMNESI; EXECUTE ON CREATEVACCINAZIONE; SELECT ON CONSEGNATO_A; SELECT ON LOTTO_VACCINO; SELECT ON VACCINANDO; SELECT ON ANAMNESI; SELECT ON IDONEO; SELECT ON NON_IDONEO_PERMANENTE; SELECT ON NON_IDONEO_TEMPORANEO; SELECT ON OPERATORE_SANITARIO_RESPONSABILE;
Ospite	Comune	EXECUTE ON CREATE_VACCINANDO;

Tabella 2: Utenti

6.1 Operazioni degli Utenti

Seguono le schede descrittore operazione per le operazioni degli utenti

Operazione	consegnaVaccino
Scopo	Registrare la consegna di un vaccino in un luogo
Argomenti	Nome e distretto del luogo, nome e lotto del farmaco, data di consegna e numero dosi da consegnare
Risultato	Aggiunta della consegna o errore
Errori	Numero dosi negativo, vaccino non esistente, luogo non esistente, lotto già consegnato, dosi eccessive

Usa	Lotto_vaccino, luogo, consegnato_a
Modifica	Consegnato_a
Prima	Non è consegnato un lotto in un luogo o è consegnato parzialmente
Dopo	Lotto consegnato parzialmente o totalmente in un luogo

Operazione	create_Vaccinando
Scopo	Registrare un vaccinando
Argomenti	Numero e scadenza tessera sanitaria, codice fiscale, nome, cognome, data di nascita, città natale, città, provincia, cap, via, civico, e-mail, positività pregressa, discriminatore per disabili, numero attestato 104, lista di disabilità, discriminatore per fragili, lista di patologie, lista di numeri di telefono
Risultato	Aggiunta di un vaccino o errore
Errori	Vaccinando già esistente, positività pregressa non valida
Usa	Vaccinando, prenotazione,
Modifica	Vaccinando, prenotazione, telefono, disabile, disabilità, fragile, patologia
Prima	Non è registrato uno specifico vaccinando
Dopo	Registrazione corretta di uno specifico vaccinando

Operazione	createAnamnesi
Scopo	Registrare un'anamnesi di un vaccinando
Argomenti	Codice fiscale, numero iscrizione dei medici, data controllo, esito controllo, data idoneità prevista
Risultato	Aggiunta di un'anamnesi di un paziente
Errori	Vaccinando non esistente, operatore sanitario non esistente, data idoneità assente, esito sconosciuto, anamnesi già esistente

Usa	Vaccinando, operatore sanitario responsabile, prenotazione, anamnesi
Modifica	Anamnesi, idoneo, non idoneo temporaneo, non idoneo permanente
Prima	Anamnesi non registrata
Dopo	Registrazione corretta dell'anamnesi di un paziente

Operazione	createVaccinazione
Scopo	Registrazione di una vaccinazione di un vaccinando
Argomenti	Codice fiscale, data vaccinazione, braccio di inoculazione, luogo di vaccinazione, distretto del luogo, lotto vaccino, nome del farmaco
Risultato	Aggiunta della vaccinazione nel database
Errori	Vaccinando non esistente, operatore sanitario non esistente, data idoneità assente, esito sconosciuto, anamnesi già esistente
Usa	Vaccinando, operatore sanitario responsabile, prenotazione, anamnesi
Modifica	Anamnesi, idoneo, non idoneo temporaneo, non idoneo permanente
Prima	Anamnesi non registrata
Dopo	Registrazione corretta dell'anamnesi di un paziente

Operazione	insertVaccino
Scopo	Registrare un nuovo lotto di vaccino
Argomenti	Nome del farmaco, numero del lotto, scadenza del lotto, numero dosi del vaccino
Risultato	Aggiunta di un nuovo lotto di vaccino
Errori	Casa produttrice non presente nel database
Usa	Stabilimento, casa_produttrice,
Modifica	Stabilimento, lotto_vaccino, produttore_di

Prima	Nuovo lotto non inserito nel database
Dopo	Registrazione corretta del nuovo lotto

Come si può notare dalle varie tabelle, le operazioni che gli utenti possono effettuare (da tenere in considerazione i permessi loro assegnati), nel dettaglio eseguono dei comandi DML (Data Manipulation Language) cioè inseriscono o aggiornano nuove t-uple in tabelle specifiche, a seconda del loro scopo. Si vuole riassumere il tutto attraverso il seguente Diagramma UML dei casi d'uso visualizzando anche chi esegue tali operazioni.

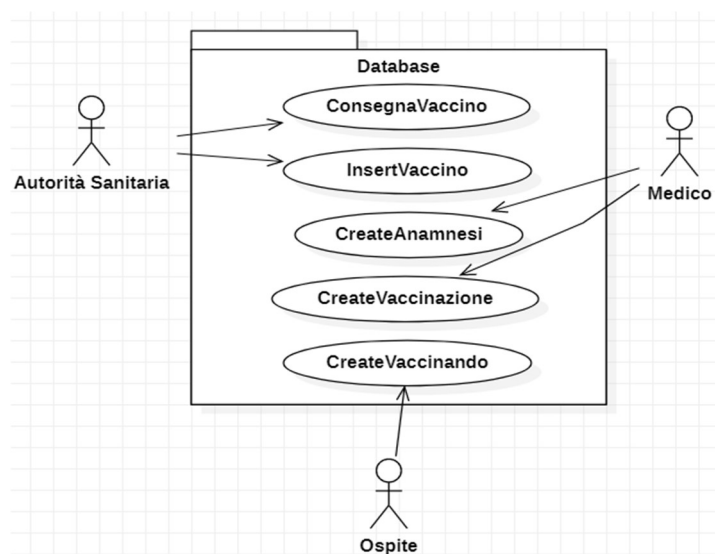


Figura 3: Diagramma UML dei casi d'uso

7. Volumi

Si vuole illustrare nella seguente tabella dei volumi quante t-uple sono presenti per ogni tabella. Non è presente nessuna politica di aggiornamento delle t-uple in quanto, essendo questo un database utilizzato a livello nazionale, si può decidere anche di verificare periodicamente quanti vaccinandi ci sono luogo per luogo, quali vaccini sono stati somministrati di più... Tutto questo quindi per questioni statistiche.

Tabella	Tipo	Volume	Incremento	Periodo
Prenotazione	E	15 000 000	1 000 000	Giorno
Vaccinando	E	5 000 000	500 000	Giorno
Fragile	E	250 000	25 000	Giorno

Patologia	E	250 000	25 000	Giorno
Disabile	E	250 000	25 000	Giorno
Disabilità	E	250 000	25 000	Giorno
Telefono	E	5 000 000	500 000	Giorno
Operatore Sanitario Responsabile	E	50 000	0	
Anamnesi	ED	10 000 000	1 000 000	Giorno
Idoneo	E	8 000 000	800 000	Giorno
Non idoneo temporaneo	E	1 900 000	190 000	Giorno
Non idoneo permanente	E	100 000	10 000	Giorno
Card vaccinato	E	3 000 000	300 000	Giorno
Luogo	E	8 000	0	
Stabilimento	E	200	0	
Casa produttrice	E	20	0	
Lotto vaccino	E	1 000 000	10 000	Giorno
Produttore di	A	1 000 000	10 000	Giorno
Vaccinazione	ED	8 000 000	500 000	Giorno
Consegnato a	A	900 000	10 000	Giorno

Tabella 3: Volumi

8. Vincoli di integrità

Di seguito sono elencati i seguenti vincoli definiti mediante trigger:

- Check_NonIdoneoTemp_Cant_Be_Vaccinated: un vaccinando risultato non idoneo temporaneo non può essere vaccinato. Vincolo dinamico per entità.
- Check_Consegna_Vaccini: un farmaco deve essere consegnato in un luogo per essere utilizzato. Vincolo dinamico per entità.

- Check_Prenotazione_Anamnesi_Date: l'anamnesi deve avvenire solo dopo la prenotazione. Vincolo dinamico per entità.
- Check_Vaccinazione_Anamnesi_Date: la vaccinazione deve avvenire solo dopo l'anamnesi. Vincolo dinamico per entità.
- Check_NonIdoneoPerm_Cant_Be_Vaccinated: un vaccinando risultato non idoneo permanente non potrà mai vaccinarsi. Vincolo statico per entità.
- Check_Seconda_Dose_Date: la seconda dose di una vaccinazione deve avvenire solo dopo la prima dose. Vincolo dinamico per entità.
- Check_Age_over18: il vaccinando deve essere maggiorenne per vaccinarsi. Vincolo dinamico per entità.
- vaccinoScaduto: se il vaccino è scaduto non può essere utilizzato per la vaccinazione. Vincolo dinamico per entità.

9. Verifica di normalità

Prima forma normale

Questo database rispetta la prima forma di normalizzazione in quanto tutti gli attributi delle varie tabelle sono atomici; non ci sono attributi che presentano come valori testi liberi; gli attributi strutturati, come indirizzo, non sono presenti nelle tabelle, bensì sono convertiti nelle loro componenti, come via, CAP, città...

Tale forma viene rispettata anche per i campi data/ora in quanto anche quando non viene espressa l'ora, per esempio nel campo data di nascita, questo non provocherà delle anomalie di inserimento, in quanto tali attributi sono definiti in un solo campo.

Seconda forma normale

Tale forma viene rispettata in quanto in tutte le tabelle non sono presenti delle dipendenze parziali da chiave. Si osserva la tabella Luogo. In tale tabella è presente l'attributo Asl di appartenenza, la quale, si potrebbe pensare che dipenda da una parte della chiave, cioè da Distretto. Distretto, però, è definito come numero, o come una coppia di caratteri. Uno stesso valore di Distretto potrebbe trovarsi in più regioni e quindi l'Asl di appartenenza di un luogo sarà determinato solo con la concatenazione di: nome del luogo e distretto, quindi con l'intera chiave.

Terza forma normale e Boyce & Codd

Anche queste due forme sono rispettate in questa base dati in quanto in nessuna tabella è presente una forma di dipendenza funzionale anomala tra attributi non chiave. Si osserva la tabella Vaccinando. E' presente il campo codice fiscale il quale è una chiave candidata. Secondo tali forme di normalizzazione, le chiavi candidate sono trattate come Primary key. Per tale motivo, quindi, l'attributo data di nascita (che dipende da codice fiscale), non dipenderà da un attributo non chiave ma da una chiave a tutti gli effetti. Questo impedisce l'andare contro tali forme di normalizzazione. Inoltre sono presenti gli attributi Città e Città Natale che fanno riferimento a nomi di comuni. Siccome in Italia sono presenti molti comuni con lo stesso nome ma situati in province e regioni differenti, l'attributo Provincia non avrà una dipendenza funzionale anomala con questi due attributi.

10. Implementazione

In questo capitolo vengono illustrate le specifiche convertite in codice eseguibile. Il codice si riferisce al DBMS Oracle 18c XE e il linguaggio adottato è il PL/SQL.

10.1 Creazione degli utenti

Si crea un utente amministratore e si definiscono i suoi privilegi:

-- Utente admin

```
CREATE USER c19v_admin IDENTIFIED BY admin;  
GRANT ALL PRIVILEGES TO c19v_admin;
```

Si decide, per la creazione degli utenti e per la gestione dei permessi, di creare dei ruoli per ogni categoria di utente:

-- Ruolo ospite

```
CREATE ROLE role_ospite IDENTIFIED BY ospiterpsw;  
GRANT CONNECT, CREATE SESSION to role_ospite;  
GRANT EXECUTE ON CREATE_VACCINANDO TO role_ospite;
```

-- Ruolo medico

```
CREATE ROLE role_medico IDENTIFIED BY medicorpsw;  
GRANT CONNECT, CREATE SESSION to role_medico;
```



```
GRANT EXECUTE ON CREATEANAMNESI TO role_medico;
GRANT EXECUTE ON CREATEVACCINAZIONE TO role_medico;
GRANT SELECT ON CONSEGNA_TO_A TO role_medico;
GRANT SELECT ON LOTTO_VACCINO TO role_medico;
GRANT SELECT ON VACCINANDO TO role_medico;
GRANT SELECT ON ANAMNESI TO role_medico;
GRANT SELECT ON IDONEO TO role_medico;
GRANT SELECT ON NON_IDONEO_PERMANENTE TO role_medico;
GRANT SELECT ON NON_IDONEO_TEMPORANEO TO role_medico;
GRANT SELECT ON OPERATORE_SANITARIO_RESPONSABILE TO role_medico;
```

-- Ruolo autorità sanitaria

```
CREATE ROLE role_autoritasanitaria IDENTIFIED BY autsanrpsw;
GRANT CONNECT, CREATE SESSION to role_autoritasanitaria;
GRANT EXECUTE ON CONSEGNAVACCINO TO role_autoritasanitaria;
GRANT EXECUTE ON INSERTVACCINO TO role_autoritasanitaria;
GRANT SELECT, DELETE, UPDATE ON CONSEGNA_TO_A TO role_autoritasanitaria;
GRANT SELECT, INSERT, DELETE, UPDATE ON LUOGO TO role_autoritasanitaria;
GRANT SELECT, DELETE, UPDATE ON LOTTO_VACCINO TO role_autoritasanitaria;
GRANT SELECT, INSERT, DELETE, UPDATE ON STABILIMENTO TO
role_autoritasanitaria;
GRANT SELECT, INSERT, DELETE, UPDATE ON CASA_PRODUTTRICE TO
role_autoritasanitaria;
GRANT SELECT, DELETE, UPDATE ON PRODUTTORE_DI TO role_autoritasanitaria;
```

Una volta creati i ruoli si creano i 3 utenti NON amministratori e gli si garantiscono i permessi tramite ruoli:

-- Utente guest1, ruolo ospite

```
CREATE USER guest1 IDENTIFIED BY guest1psw;
GRANT role_ospite TO guest1;
```

-- Utente medico1, ruolo medico

```
CREATE USER medico1 IDENTIFIED BY medico1psw;
GRANT role_medico TO medico1;
```

-- Utente authsan1, ruolo autorità sanitaria

CREATE USER authsan1 IDENTIFIED BY authsan1psw;

GRANT role_autoritasanitaria TO authsan1;

10.2 Data Definition Language

Il DDL riflette esattamente ciò che è presente nel modello relazione (vedere sezione 5). Tutte le 20 tabelle sono state create mediante l'istruzione CREATE TABLE. Sono stati anche definiti tutti i possibili vincoli esprimibili nel modello relazionale come l'obbligatorietà (NOT NULL), unicità (UNIQUE), vincoli di dominio statico (CHECK)... Si osserva che in Oracle il tipo Date include anche l'ora. Per tale motivo, anche se nel diagramma EE/R sono presenti due attributi distinti data e ora, nel codice sono stati tradotti mediante un unico tipo di dato DATE.

10.2.1 Vaccinando

Come specificato dai requisiti, si vuole tenere traccia di tutte le informazioni di un paziente che vuole vaccinarsi. Tale tabella ha questo obiettivo.

```
CREATE TABLE Vaccinando (  
  Num_Tessera_Sanitaria      CHAR(20) PRIMARY KEY,  
  Scadenza_Tessera_Sanitaria  DATE,  
  Codice_Fiscale             CHAR(16) UNIQUE NOT NULL,  
  Nome                       VARCHAR2(25) NOT NULL,  
  Cognome                    VARCHAR2(25) NOT NULL,  
  Data_Nascita               DATE NOT NULL,  
  Citta_Natale               VARCHAR2(50) NOT NULL,  
  Citta                      VARCHAR2(50) NOT NULL,  
  Provincia                  CHAR(2) NOT NULL,  
  CAP                        CHAR(5),  
  Via                        VARCHAR2(30),  
  Civico                     VARCHAR2(10),  
  e_mail                     VARCHAR2(40),  
  Positivita_Pregressa        CHAR(1) DEFAULT 'F' CHECK (Positivita_Pregressa IN ('T',  
  'F')) NOT NULL  
);
```

10.2.2 Prenotazione

Siccome ogni vaccinando si deve prenotare, si tiene traccia anche delle informazioni di ogni prenotazione fatta da un vaccinando. Ogni prenotazione è identificata dalla concatenazione di un numero protocollo, il quale cambia per ogni vaccinando, e numero tentativo. Quest'ultimo attributo è utilizzato in quanto, ogni vaccinando, effettuando l'anamnesi, possono risultare non idonei temporaneamente. Ciò significa che dovranno rifare il controllo fin quando non saranno idonei a tutti gli effetti. Per ogni controllo fatto il numero di prenotazione non cambierà mai ma il numero tentativo sarà incrementato di 1.

```
CREATE TABLE Prenotazione(  
  Num_Protocollo      NUMBER(8, 0),  
  Num_Tentativi       NUMBER(2, 0),  
  Data_Ora_Previste   DATE NOT NULL,  
  
  Tipo_Prenotazione   VARCHAR2(7) CHECK (Tipo_Prenotazione IN  
('SINGOLA','DOPPIA')) NOT NULL,  
  
  PRIMARY KEY (Num_Protocollo, Num_Tentativi)  
);
```

Tramite il seguente ALTER TABLE si aggiunge il legame tra vaccinando e prenotazione. In particolare si aggiunge un constraint per definire la foreign key sulla tabella Vaccinando da Prenotazione.

```
ALTER TABLE Vaccinando ADD (  
  Num_Protocollo   NUMBER(8, 0),  
  Num_Tentativi    NUMBER(2, 0),  
  
  CONSTRAINT Foreign_Key_Vaccinando FOREIGN KEY (Num_Protocollo,  
  Num_Tentativi) REFERENCES Prenotazione(Num_Protocollo, Num_Tentativi)  
);
```

10.2.3 Fragile

Si definisce la tabella Fragile tenendo conto del suo legame con Vaccinando essendo essa una classe figlia della specializzazione con Vaccinando. Si aggiunge, per tale motivo, la foreign key su Fragile da Vaccinando.

```
CREATE TABLE Fragile (  
    Num_Tessera_Sanitaria      CHAR(20) PRIMARY KEY,  
  
    CONSTRAINT Foreign_Key_Fragile FOREIGN KEY (Num_Tessera_Sanitaria)  
    REFERENCES Vaccinando(Num_Tessera_Sanitaria)  
);
```

10.2.4 Disabile

Il discorso fatto per la tabella Fragile vale anche per la tabella Disabile creata con il seguente codice:

```
CREATE TABLE Disabile (  
    Num_Attestato_104          VARCHAR2(10) NOT NULL,  
    Num_Tessera_Sanitaria      CHAR(20) PRIMARY KEY,  
  
    CONSTRAINT Foreign_Key_Disabile FOREIGN KEY (Num_Tessera_Sanitaria)  
    REFERENCES Vaccinando(Num_Tessera_Sanitaria)  
);
```

10.2.5 Disabilità

Disabilità, essendo un attributo multivalore, si decide di definirlo come entità a parte. Conterrà come attributi il tipo specifico di disabilità e la foreign key Num_Tessera_Sanitaria discendente da Vaccinando e Disabile. La primary key sarà la concatenazione di entrambi gli attributi.

```
CREATE TABLE Disabilita (  
    Num_Tessera_Sanitaria      CHAR(20),  
    Disabilita                 VARCHAR2(40),  
  
    PRIMARY KEY (Num_Tessera_Sanitaria, Disabilita),
```

```
CONSTRAINT Foreign_Key_Disabilita FOREIGN KEY (Num_Tessera_Sanitaria)
REFERENCES Disabile(Num_Tessera_Sanitaria)
);
```

10.2.6 Patologia

Ciò che è stato definito nel paragrafo dedicato alla tabella Disabilità vale anche per la tabella Patologia.

```
CREATE TABLE Patologia (
Num_Tessera_Sanitaria      CHAR(20),
Patologia                  VARCHAR2(40),

PRIMARY KEY (Num_Tessera_Sanitaria, Patologia),

CONSTRAINT Foreign_Key_Patologia FOREIGN KEY (Num_Tessera_Sanitaria)
REFERENCES Fragile(Num_Tessera_Sanitaria)
);
```

10.2.7 Telefono

Anche Telefono, essendo un attributo multivalore si decide di definirlo come una tabella separata da Vaccinando. Avrà come PK la concatenazione di Telefono e Num_Tessera_Sanitaria (FK).

```
CREATE TABLE Telefono (
Num_Tessera_Sanitaria      CHAR(20),
Telefono                  VARCHAR2(20),

PRIMARY KEY (Num_Tessera_Sanitaria, Telefono),

CONSTRAINT Foreign_Key_Telefono FOREIGN KEY (Num_Tessera_Sanitaria)
REFERENCES Vaccinando(Num_Tessera_Sanitaria)
);
```

10.2.8 Operatore Sanitario Responsabile

Tale Tabella definisce le informazioni del medico responsabile dell'anamnesi come specificato nei requisiti.

```
CREATE TABLE Operatore_Sanitario_Responsabile (  
  Num_Iscrizione_Ordine_Medici      NUMBER(10, 0) PRIMARY KEY,  
  Nome                               VARCHAR2(25) NOT NULL,  
  Cognome                           VARCHAR2(25) NOT NULL  
);
```

10.2.9 Anamnesi

Tale tabella contiene tutte le informazioni sul controllo medico che il paziente sostiene. Come si può notare, essendo un'entità debole, secondo le regole di traduzione, essa avrà come PK la concatenazione della sua chiave debole (Data_Ora_Controllo) e la chiave forte (concatenazione di Num_Protocollo e Num_Tentativi), in modo tale da avere una chiave multiattributo.

Gli attributi che compongono la chiave forte sono anche foreign key, dovuti alla molteplicità dell'associazione 1:N. Essendo Anamnesi associata anche con l'operatore sanitario responsabile ci sono anche le foreign key da quest'ultima tabella.

```
CREATE TABLE Anamnesi (  
  Num_Protocollo                     NUMBER(8, 0) NOT NULL,  
  Num_Tentativi                     NUMBER(2, 0) NOT NULL,  
  Data_Ora_Controllo                 DATE NOT NULL,  
  Num_Iscrizione_Ordine_Medici       NUMBER(10, 0) NOT NULL,  
  
  PRIMARY KEY (Num_Protocollo, Num_Tentativi, Data_Ora_Controllo),  
  
  CONSTRAINT Foreign_Key_Anamnesi1 FOREIGN KEY (Num_Protocollo,  
  Num_Tentativi) REFERENCES Prenotazione (Num_Protocollo, Num_Tentativi),  
  CONSTRAINT Foreign_Key_Anamnesi2 FOREIGN KEY  
  (Num_Iscrizione_Ordine_Medici) REFERENCES  
  Operatore_Sanitario_Responsabile (Num_Iscrizione_Ordine_Medici)  
);
```

10.2.10 Non Idoneo Temporaneo

Essa è una delle 3 classi figlie di anamnesi. Avendo utilizzato la tecnica del partizionamento verticale (vedere sezione 5.1), tutte le classi figlie di questa specializzazione saranno tabelle distinte. Avranno come PK e FK la chiave i Anamnesi. In più, tale tabella presenta un attributo in più rispetto alle altre 2 tabelle, Data_Ideoneità_Prevista che indica dopo quanto tempo il paziente potrà rifare il controllo per verificare la sua idoneità e sostenere la vaccinazione.

```
CREATE TABLE Non_Idoneo_Temporaneo(  
  Num_Procollo          NUMBER(8, 0),  
  Num_Tentativi         NUMBER(2, 0),  
  Data_Ora_Controllo    DATE,  
  Data_Idoneita_Prevista DATE NOT NULL,
```

```
PRIMARY KEY (Num_Procollo, Num_Tentativi, Data_Ora_Controllo),
```

```
CONSTRAINT Foreign_Key_Non_Idoneo_Temporaneo1 FOREIGN KEY  
(Num_Procollo, Num_Tentativi, Data_Ora_Controllo) REFERENCES  
Anamnesi(Num_Procollo, Num_Tentativi, Data_Ora_Controllo)  
);
```

10.2.11 Idoneo

Anche tale tabella è un'entità figlia della specializzazione di Anamnesi. Tutto ciò che è stato scritto per la tabella Non_Idoneo_Temporaneo vale anche per questa tabella.

```
CREATE TABLE Idoneo(  
  Num_Procollo          NUMBER(8, 0),  
  Num_Tentativi         NUMBER(2, 0),  
  Data_Ora_Controllo    DATE,
```

```
PRIMARY KEY (Num_Procollo, Num_Tentativi, Data_Ora_Controllo),
```

```
CONSTRAINT Foreign_Key_Idoneo FOREIGN KEY (Num_Procollo, Num_Tentativi,  
Data_Ora_Controllo) REFERENCES Anamnesi(Num_Procollo, Num_Tentativi,  
Data_Ora_Controllo)
```

);

10.2.12 Non Idoneo Permanente

E' la terza e ultima entità figlia della specializzazione di Anamnesi. Anche qui valgono le stesse caratteristiche dette per le altre entità figlie.

```
CREATE TABLE Non_Idoneo_Permanente (  
  Num_Protocollo      NUMBER(8, 0),  
  Num_Tentativi       NUMBER(2, 0),  
  Data_Ora_Controllo  DATE,
```

```
PRIMARY KEY (Num_Protocollo, Num_Tentativi, Data_Ora_Controllo),
```

```
CONSTRAINT Foreign_Key_Non_Idoneo_Permanente FOREIGN KEY  
(Num_Protocollo, Num_Tentativi, Data_Ora_Controllo) REFERENCES  
Anamnesi(Num_Protocollo, Num_Tentativi, Data_Ora_Controllo));
```

10.2.13 Card Vaccinato

Tale tabella contiene tutte le informazioni riguardanti l'attestato di vaccinazione di un paziente.

```
CREATE TABLE Card_Vaccinato (  
  Num_Carta          NUMBER(8, 0) PRIMARY KEY,  
  Data_Attivazione   DATE NOT NULL,  
  Data_Scadenza      DATE NOT NULL  
);
```

10.2.14 Stabilimento

Tale tabella contiene tutte le informazioni necessarie a distinguere i vari stabilimenti in cui è prodotto un lotto di vaccino.

```
CREATE TABLE Stabilimento (  
  Nome      VARCHAR2(30) PRIMARY KEY,  
  Nazione   VARCHAR2(20),  
  Citta     VARCHAR2(50),  
  Via       VARCHAR2(50),  
  Civico    VARCHAR2(10)
```


);

10.2.15 Casa Produttrice

Come si capisce già dal nome si vuole conoscere la casa produttrice di un particolare vaccino.

```
CREATE TABLE Casa_Produttrice (  
Nome    VARCHAR2(30) PRIMARY KEY,  
Nazione VARCHAR2(20),  
Sito_Web VARCHAR2(30)  
);
```

10.2.16 Luogo

Questa tabella indica la località dove è avvenuta la vaccinazione. Siccome nel modello relazionale è stata utilizzata la tecnica della tabella unica, si usufruisce anche di un discriminatore per distinguere le varie t-uple. La PK è la concatenazione di nome e distretto.

```
CREATE TABLE Luogo (  
Nome          VARCHAR2(30),  
Distretto     VARCHAR2(20),  
ASL_Appartenenza VARCHAR2(20) NOT NULL,  
Provincia     CHAR(2) NOT NULL,  
Citta         VARCHAR2(50) NOT NULL,  
CAP           CHAR(5) NOT NULL,  
Via           VARCHAR2(30) NOT NULL,  
Civico        VARCHAR2(10),  
  
Discriminatore_PVO CHAR(1) DEFAULT '1' CHECK (Discriminatore_PVO IN ('1', '0')) NOT NULL,  
Codice_Presidio_Ospedaliero VARCHAR2(20) UNIQUE,  
  
PRIMARY KEY (Nome, Distretto)  
);
```

10.2.17 Lotto Vaccino

Tale tabella ha l'obiettivo di definire le caratteristiche di un lotto di vaccino. In particolare si fa la distinzione tra vaccino a singola dose (tipo Janssen) o doppia (Pfizer, Astrazeneca...). Tale distinzione la si definisce mediante un discriminatore in quanto lotto è specializzato proprio in queste 2 categorie. Se è a doppia dose c'è anche l'attributo che stabilisce il periodo di tempo tra prima e seconda dose. La PK è la concatenazione tra il numero lotto e il nome del farmaco in quanto diversi farmaci possono usufruire della stessa numerazione per classificare i lotti. Per tale motivo usufruire solo del numero lotto per distinguere i lotti non basterebbe.

```
CREATE TABLE Lotto_Vaccino (  
  Num_Lotto          VARCHAR2(10),  
  Nome_Farmaco       VARCHAR2(30),  
  Scadenza           DATE NOT NULL,  
  Num_Dosi_Lotto     NUMBER(6, 0) NOT NULL,  
  
  Nome_Stabilimento  VARCHAR2(30) NOT NULL,  
  
  Discriminatore     CHAR(1) CHECK (Discriminatore IN ('T', 'F')) NOT NULL,  
  Tempo_Prima_Seconda_Dose  NUMBER(3, 0),  
  
  PRIMARY KEY (Num_Lotto, Nome_Farmaco),  
  
  CONSTRAINT Foreign_Key_Lotto_Vaccino FOREIGN KEY (Nome_Stabilimento)  
  REFERENCES Stabilimento(Nome) ON DELETE CASCADE  
);
```

10.2.18 Produttore Di

Tale tabella è frutto dell'associazione tra Casa Produttrice e Lotto. Avendo una molteplicità M:N si definisce tale tabella che, secondo le regole di traduzione avrà come PK la concatenazione delle PK delle entità coinvolte nell'associazione. Tali PK saranno anche FK.

```
CREATE TABLE Produttore_Di (  
  Nome_Produttore   VARCHAR2(30),
```

Nome_Farmaco VARCHAR2(30),
Num_Lotto VARCHAR2(10),

PRIMARY KEY (Nome_Produttore, Nome_Farmaco, Num_Lotto),

CONSTRAINT Foreign_Key_Produttore_Di1 FOREIGN KEY (Nome_Produttore)
REFERENCES Casa_Produttrice(Nome),

CONSTRAINT Foreign_Key_Produttore_Di2 FOREIGN KEY (Nome_Farmaco,
Num_Lotto) REFERENCES Lotto_Vaccino(Nome_Farmaco, Num_Lotto) ON DELETE
CASCADE

);

10.2.19 Vaccinazione

Tale tabella contiene le informazioni della vaccinazione effettuata dal paziente. In particolare, essa essendo entità debole prevede come PK la concatenazione della sua chiave debole (Data_Ora_Effettiva) e la chiave dell'entità forte (Card Vaccinato), cioè Num_Carta. Inoltre, dalle varie associazioni che coinvolgono tale tabella, ci sono le FK che puntano alle PK di Lotto, Luogo, Idoneo. Dal legame con quest'ultima tabella si comprende il fatto che solo chi è risultato idoneo dall'anamnesi può sostenere la vaccinazione.

CREATE TABLE Vaccinazione (

Data_Ora_Effettiva DATE,

Braccio_Inoculazione CHAR(1) CHECK (Braccio_Inoculazione IN ('L', 'R')) NOT
NULL,

Num_Carta NUMBER(8, 0),

Nome_Luogo VARCHAR2(30) NOT NULL,

Distretto_Luogo VARCHAR2(20) NOT NULL,

Num_Lotto VARCHAR2(10) NOT NULL,

Nome_Farmaco VARCHAR2(30) NOT NULL,

Num_Protocollo NUMBER(8, 0) NOT NULL,

Num_Tentativi NUMBER(2, 0) NOT NULL,

Data_Ora_Controllo DATE NOT NULL,

PRIMARY KEY (Num_Carta, Data_Ora_Effettiva),

```
CONSTRAINT Foreign_Key_Vaccinazione1 FOREIGN KEY (Num_Protocollo,  
Num_Tentativi, Data_Ora_Controllo) REFERENCES Idoneo(Num_Protocollo,  
Num_Tentativi, Data_Ora_Controllo),  
CONSTRAINT Foreign_Key_Vaccinazione2 FOREIGN KEY (Num_Carta) REFERENCES  
Card_Vaccinato(Num_Carta),  
CONSTRAINT Foreign_Key_Vaccinazione3 FOREIGN KEY (Nome_Luogo,  
Distretto_Luogo) REFERENCES Luogo(Nome, Distretto),  
CONSTRAINT Foreign_Key_Vaccinazione4 FOREIGN KEY (Num_Lotto,  
Nome_Farmaco) REFERENCES Lotto_Vaccino(Num_Lotto, Nome_Farmaco)  
);
```

10.2.20 Consegnato_a

Anche tale tabella è frutto dell'associazione tra Luogo e Lotto. Definisce le consegne di lotti in specifici luoghi. Per tale motivo, secondo le regole di traduzione, tale tabella avrà come PK la concatenazione delle PK di Lotto e Luogo con anche la data di consegna. In più è presente anche un ulteriore attributo che stabilisce il numero di dosi consegnate.

```
CREATE TABLE Consegnato_a(  
Nome_Luogo          VARCHAR2(30),  
Distretto_Luogo     VARCHAR2(20),  
Nome_Farmaco        VARCHAR2(30),  
Num_Lotto           VARCHAR2(10),  
Data_Consegna       DATE NOT NULL,  
Num_Dosi_Consegnate NUMBER(6, 0) NOT NULL,  
  
PRIMARY KEY (Nome_Luogo, Distretto_Luogo, Nome_Farmaco, Num_Lotto,  
Data_Consegna),  
  
CONSTRAINT Foreign_Key_Consegnato1 FOREIGN KEY (Nome_Luogo,  
Distretto_Luogo) REFERENCES Luogo(Nome, Distretto),  
CONSTRAINT Foreign_Key_Consegnato2 FOREIGN KEY (Nome_Farmaco, Num_Lotto)  
REFERENCES Lotto_Vaccino(Nome_Farmaco, Num_Lotto)  
);
```

10.3 Data Manipulation Language

Le DML sono una serie di comandi appositamente utilizzati per effettuare il popolamento delle tabelle create con le DDL. Le tabelle sono state popolate sia in maniera diretta attraverso le operazioni di Insert sia, come nei successivi paragrafi è stato spiegato, attraverso apposite procedure. Questo perché le procedure, per raggiungere il loro obiettivo effettuano degli inserimenti in tabelle specifiche.

10.3.1 Prenotazione

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (1, 1, TO_DATE('2021-07-14
11:05:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (2, 1, TO_DATE('2021-07-14
11:15:54', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (3, 1, TO_DATE('2021-07-14
11:25:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (3, 2, TO_DATE('2021-08-16
11:01:15', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (4, 1, TO_DATE('2021-07-14
11:35:54', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (5, 1, TO_DATE('2021-07-14
11:45:54', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (6, 1, TO_DATE('2021-07-14
11:55:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (6, 2, TO_DATE('2021-08-17
11:02:12', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (7, 1, TO_DATE('2021-07-14
12:05:54', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (8, 1, TO_DATE('2021-07-14  
12:15:54', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (9, 1, TO_DATE('2021-07-14  
12:25:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (9, 2, TO_DATE('2021-09-18  
11:03:20', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (10, 1, TO_DATE('2021-07-14  
12:35:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (10, 2, TO_DATE('2021-09-20  
12:44:19', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (11, 1, TO_DATE('2021-07-14  
12:45:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (11, 2, TO_DATE('2021-08-31  
11:03:36', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (12, 1, TO_DATE('2021-07-14  
12:55:54', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (13, 1, TO_DATE('2021-07-14  
13:05:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (14, 1, TO_DATE('2021-07-14  
13:15:54', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (15, 1, TO_DATE('2021-08-31  
11:13:36', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (16, 1, TO_DATE('2021-08-31  
11:23:36', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (17, 1, TO_DATE('2021-08-31  
11:33:36', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (18, 1, TO_DATE('2021-08-31  
11:43:36', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (19, 1, TO_DATE('2021-08-31  
11:53:36', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (20, 1, TO_DATE('2021-08-31  
12:03:36', 'YYYY-MM-DD HH24:MI:SS'), 'SINGOLA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (21, 1, TO_DATE('2021-08-31  
12:13:36', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (22, 1, TO_DATE('2021-09-20  
12:54:19', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (22, 2, TO_DATE('2021-11-19  
12:59:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (23, 1, TO_DATE('2021-11-19  
13:09:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (23, 2, TO_DATE('2021-11-29  
13:09:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (24, 1, TO_DATE('2021-11-19  
13:19:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (24, 2, TO_DATE('2021-11-29  
13:19:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (25, 1, TO_DATE('2021-11-19  
13:29:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```



```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (25, 2, TO_DATE('2021-11-29  
13:39:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (26, 1, TO_DATE('2021-11-19  
13:39:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (26, 2, TO_DATE('2021-11-29  
13:39:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (27, 1, TO_DATE('2021-11-19  
13:49:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (27, 3, TO_DATE('2022-01-20  
13:50:00', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (27, 2, TO_DATE('2021-11-21  
13:50:00', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (28, 1, TO_DATE('2021-11-19  
13:59:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (28, 3, TO_DATE('2022-01-20  
14:14:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (28, 2, TO_DATE('2021-11-21  
14:14:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (29, 1, TO_DATE('2021-11-19  
14:09:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (29, 3, TO_DATE('2022-01-20  
14:24:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');  
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (29, 2, TO_DATE('2021-11-21  
14:24:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```



```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (30, 1, TO_DATE('2021-11-19  
14:19:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (30, 3, TO_DATE('2022-01-20  
14:34:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (30, 2, TO_DATE('2021-11-21  
14:34:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (31, 1, TO_DATE('2021-11-19  
14:29:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (31, 3, TO_DATE('2022-01-20  
14:44:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (31, 2, TO_DATE('2021-11-21  
14:44:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (32, 1, TO_DATE('2021-11-19  
14:39:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (32, 2, TO_DATE('2021-11-21  
14:54:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

```
INSERT INTO PRENOTAZIONE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_PREVISTE, TIPO_PRENOTAZIONE) VALUES (32, 3, TO_DATE('2022-01-20  
14:54:59', 'YYYY-MM-DD HH24:MI:SS'), 'DOPPIA');
```

10.3.2 Vaccinando

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
(('80380001532234567878', TO_DATE('2027-07-19 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'RZLGNR79S08L845C', 'Gennaro', 'Arzilli', TO_DATE('1979-11-08  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Vico Equense', 'Milano', 'MI', null, null, null,  
'arzgen@gmail.com', 'F', 1, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001501111111111', null, 'BBRFNC98R06H919I', 'Franco', 'Barbaroni',  
TO_DATE('1998-06-10 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'San Giovanni In  
Fiore', 'San Giovanni In Fiore', 'CS', null, null, null, 'barbfrank@yahoo.it', 'T', 2, 1);  
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001501111111118', null, 'FLLMRC66R06H061X', 'Marco', 'Fellini',  
TO_DATE('1966-10-06 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Prignano Sulla  
Secchia', 'Prignano Sulla Secchia', 'MO', null, null, null, 'marcofellini66@icloud.it', 'F',  
3, 1);  
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001512345678985', TO_DATE('2025-03-27 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'ZZZFNC91A51D086X', 'Francesca', 'Zazza', TO_DATE('1991-01-11  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Cosenza', 'Cosenza', 'CO', '80053', 'Via  
Collocaputo', '93', 'francescazazza@gmail.com', 'T', 4, 1);  
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001515978456321', TO_DATE('2024-05-15 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'CROPPE37E41G070I', 'Giuseppe', 'Crosta', TO_DATE('1937-05-01  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Onifai', 'Onifai', 'NU', '8020', 'Via Bella', '55',  
null, 'T', 5, 1);  
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001532125469846', TO_DATE('2024-05-15 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'DMLDNL01E41G070I', 'Danilo', 'Di Meglio', TO_DATE('2001-12-27
```

00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Benevento', 'Benevento', 'BE', '80090', 'Via Chiara', null, null, 'F', 6, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001532154698458', TO_DATE('2024-05-15 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'CRSFBA80A01F205E', 'Fabio', 'Caruso', TO_DATE('1976-07-21 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Milano', 'Milano', 'MI', '80100', 'Via Azzurra', '2', 'CarusFabio76@gmail.com', 'T', 7, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001532125469865', TO_DATE('2022-11-16 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'DIRVCG77E41G070I', 'Vincenzo', 'Di Iorio', TO_DATE('1981-01-03 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Aosta', 'Aosta', 'AO', '80105', 'Via Gialla', '8', 'DilorioVoncenzo@gmail.com', 'T', 8, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001532125469877', TO_DATE('2023-08-29 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'BRGNTF85E41G070I', 'Antonio', 'Bargiggia', TO_DATE('1985-02-03 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Genoa', 'Genoa', 'GE', '80106', 'Via Rossa', null, null, 'F', 9, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001500319201049', TO_DATE('2026-04-20 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'CLCFNC00L10E396M', 'Francesco', 'Calcopietro', TO_DATE('2000-07-10 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Lacco Ameno', 'Serrara Fontana', 'NA', '80081', 'Via Aito di Fuori', '2', 'francescocalopietro00@gmail.com', 'F', 10, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,

```
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001509877110988', null, 'MLVLGU85P45L259P', 'Luigi', 'Malvone',  
TO_DATE('1990-09-05 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Torre del Greco',  
'Torre del Greco', 'NA', null, null, null, null, 'F', 11, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001508765434580', null, 'CRSDNY74D12B905U', 'Denny', 'Caruso',  
TO_DATE('1974-04-12 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Casalnuovo di  
Napoli', 'Casalnuovo di Napoli', 'NA', null, null, null, null, 'T', 12, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001595412365479', TO_DATE('2023-11-30 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'NTLCLM55E41G070I', 'Colomba', 'Natale', TO_DATE('1955-12-15  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Ischia', 'Ischia', 'NA', '80077', 'Via Aito di  
Fuori', '1', 'natalecolomba55@gmail.com', 'F', 13, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500161870297', TO_DATE('2021-07-17 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'PNTMRN01C24D129E', 'Mariano', 'Aponte', TO_DATE('2001-04-24  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Castellammare di Stabia', 'Castellammare di  
Stabia', 'NA', '80053', 'Via Roma', '93', 'aponte2001@gmail.com', 'F', 14, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001532234567879', TO_DATE('2027-07-20 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'MRRLRD85B05F205V', 'Alfredo', 'Murari', TO_DATE('1985-02-05  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Milano', 'Milano', 'MI', '80000', 'Via  
celstiale', '10', null, 'F', 15, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,
```

```
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
(  
'80380001532234567870', TO_DATE('2027-07-21 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'RCMGNN94L54L219S', 'Giovanna', 'Arcamone', TO_DATE('1985-02-05  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Torino', 'Genoa', 'GE', '80001', 'Via brutta',  
null, null, 'T', 16, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
(  
'80380001532234567871', TO_DATE('2027-07-22 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'MLFSFN90S18L781T', 'Stefano', 'Amalfitano', TO_DATE('1990-11-18  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Verona', 'Verona', 'VR', '80002', null, null,  
'stefanoerbello@live.com', 'T', 17, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
(  
'80380001532234567872', TO_DATE('2028-07-22 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'GLDMNN77E59G791N', 'Marianna', 'Geladas', TO_DATE('1977-05-19  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Polistena', 'Polistena', 'RC', '80003', 'Via di  
dentro', '2', 'gelido@gmail.com', 'F', 18, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
(  
'80380001532234567873', TO_DATE('2029-07-22 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'BLLVCN20A41B963V', 'Vincenza', 'Bellobuono', TO_DATE('1920-01-01  
00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Caserta', 'Caserta', 'CE', '80004', null, null,  
null, 'F', 19, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
(  
'80380001532234567874', TO_DATE('2022-07-22 00:00:00', 'YYYY-MM-DD  
HH24:MI:SS'), 'MTTGRD67H46F839S', 'Gerarda', 'Mattera', TO_DATE('1967-06-06
```

00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Napoli', 'Napoli', 'NA', '80005', null, null, null, 'T', 20, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('8038000153221111787', TO_DATE('2023-07-19 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'NTNNNL69A01L845T', 'Antonello', 'Antonelli', TO_DATE('1969-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Vico Equense', 'Napoli', 'NA', null, null, null, 'antoantonelli1969@gmail.com', 'F', 21, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001500161874596', TO_DATE('2021-07-13 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'PNTRFL64P22C129V', 'Raffaele', 'Aponte', TO_DATE('1964-09-22 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Castellammare di Stabia', 'Castellammare di Stabia', 'NA', '80053', null, null, 'raffaele.aponte@gmail.com', 'F', 22, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001500116783562', null, 'MRARSS80A01F839V', 'Mario', 'Rossi', TO_DATE('1964-07-22 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Napoli', 'Napoli', 'NA', '80053', null, null, null, 'F', 23, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001500116999962', null, 'GNLBCH80A02F839O', 'Gianluca', 'Bianchi', TO_DATE('1964-07-22 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Napoli', 'Napoli', 'NA', '80053', null, null, null, 'F', 24, 1);

INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA, SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME, DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL, POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) **VALUES** ('80380001500116990002', null, 'MCHVLI80A01F839I', 'Michele', 'Viola',


```
TO_DATE('1960-02-09 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Napoli', 'Napoli', 'NA',  
'80053', null, null, null, 'F', 25, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500110980002', null, 'GNVBRB80A41F839Q', 'Genoveffa', 'Barba',  
TO_DATE('1979-01-04 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Napoli', 'Napoli', 'NA',  
'80053', null, null, null, 'F', 26, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500118888802', null, 'SLVGTN80A01H501C', 'Salvatore Antonio',  
'Gaetano', TO_DATE('1973-10-29 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Crotone',  
'Napoli', 'NA', '80053', null, null, null, 'F', 27, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500122288802', null, 'DNLGPP80A01F839M', 'Giuseppe', 'Daniele',  
TO_DATE('1955-03-19 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Napoli', 'Napoli', 'NA',  
'80053', null, null, null, 'F', 28, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500109088802', null, 'DNDFRZ80A01H501B', 'Fabrizio Cristiano', 'De  
Andrè', TO_DATE('1940-02-18 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Genova',  
'Milano', 'MI', '80053', null, null, null, 'F', 29, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500009088802', null, 'FRNGCN80A01H501K', 'Francesco', 'Guccini',  
TO_DATE('1940-06-14 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Modena', 'Milano',  
'MI', '80053', null, null, null, 'F', 30, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500009444802', null, 'PLRFL80A41H501Z', 'Raffaella', 'Pelloni',  
TO_DATE('1943-06-18 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Bologna', 'Roma',  
'RM', '80053', null, null, null, 'F', 31, 1);
```

```
INSERT INTO VACCINANDO (NUM_TESSERA_SANITARIA,  
SCADENZA_TESSERA_SANITARIA, CODICE_FISCALE, NOME, COGNOME,  
DATA_NASCITA, CITTA_NATALE, CITTA, PROVINCIA, CAP, VIA, CIVICO, E_MAIL,  
POSITIVITA_PREGRESSA, NUM_PROTOCOLLO, NUM_TENTATIVI) VALUES  
( '80380001500001144802', null, 'SLVMHL80A01F284Q', 'Salvemini', 'Michele',  
TO_DATE('1943-06-18 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Molfetta', 'Roma',  
'RM', null, null, null, null, 'F', 32, 1);
```

10.3.3 Fragile

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001501111111118');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001512345678985');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001515978456321');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532125469865');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532125469877');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532154698458');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567870');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567871');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567872');
```

```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567873');
```



```
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567874' );  
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567878' );  
INSERT INTO FRAGILE (NUM_TESSERA_SANITARIA) VALUES  
( '80380001532234567879' );
```

10.3.4 Patologia

```
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001501111111118', 'Diabete Mellito' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001501111111118', 'Gastrite e ulcera peptica' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001512345678985', 'Meningite' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001512345678985', 'Sterilità' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001515978456321', 'Tumore alla prostata' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532125469865', 'Cefalea' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532125469877', 'Artrosi' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532125469877', 'Meningite' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532154698458', 'Dermatiti' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532154698458', 'Incontinenza Urinaria' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567870', 'Osteoporosi' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567871', 'Scoliosi' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567871', 'Tumore alla prostata' );  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567872', 'Sinusite' );
```

```
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567873', 'Ulcera peptica');  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567874', 'Malattia di Parkinson');  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567878', 'Allergia Api');  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567878', 'Allergia Graminacee');  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567878', 'Asma');  
INSERT INTO PATOLOGIA (NUM_TESSERA_SANITARIA, PATOLOGIA) VALUES  
( '80380001532234567879', 'Osteoporosi');
```

10.3.5 Disabile

```
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345346', '80380001532234567879');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345347', '80380001532234567870');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345348', '80380001532234567871');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345349', '80380001532234567872');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345350', '80380001532234567873');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345351', '80380001532234567874');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345345', '80380001532234567878');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345345', '80380001501111111111');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345346', '80380001512345678985');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345346', '80380001515978456321');  
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345346', '80380001532125469846');
```

```
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345347', '80380001532125469865');
```

```
INSERT INTO DISABILE (NUM_ATTESTATO_104, NUM_TESSERA_SANITARIA) VALUES  
( '0012345348', '80380001532125469877');
```

10.3.6 Disabilità

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001501111111111', 'Distrofia muscolare');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001512345678985', 'Ictus Cerebrale');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001515978456321', 'Mutismo');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532125469846', 'Sindrome di Down');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532125469865', 'Psicosi');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532125469877', 'Epilessia');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567870', 'Paraplegia');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567871', 'Disabilità cerebrale');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567872', 'Distrofia muscolare');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567873', 'Sindrome di Williams');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567874', 'Sindrome di Angelman');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567878', 'Alzheimer');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567878', 'SLA');
```

```
INSERT INTO DISABILITA (NUM_TESSERA_SANITARIA, DISABILITA) VALUES  
( '80380001532234567879', 'SLA');
```

10.3.7 Telefono

```
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001500161870297', '3801886739');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001500319201049', '3801886739');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001501111111111', '3205469870');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001501111111118', '3461578923');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001508765434580', '3801886732');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001509877110988', '3801886730');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001512345678985', '0815060938');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001515978456321', '0815060945');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532125469846', '0815060955');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532125469846', '3102546982');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532125469865', '0815060963');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532125469865', '3102546988');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532125469877', '0815060971');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532125469877', '3102546989');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('8038000153221111787 ', '0812345432');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('8038000153221111787 ', '3333455623');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532234567870', '0266787632');
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES
('80380001532234567871', '0266787634');
```

```
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567872', '0266787635');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567873', '0266787636');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567874', '0266787637');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567878', '0266201530');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567878', '0266787630');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567879', '0266201531');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001532234567879', '0266787631');  
INSERT INTO TELEFONO (NUM_TESSERA_SANITARIA, TELEFONO) VALUES  
( '80380001595412365479', '3397298181');
```

10.3.8 Operatore Sanitario Responsabile

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (1111111111,  
'Mario', 'Rossi');  
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (1234567890,  
'Giorgia', 'Di Scala');  
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (2222222222,  
'Luca', 'Verdi');  
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (3333333333,  
'Marco', 'Bianchi');  
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (4444444444,  
'Gianluca', 'Monti');  
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (5555555555,  
'Giovanni', 'Montella');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (6666666666,  
'Francesco', 'Arcamone');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (7410852963,  
'Martina', 'Cianciarelli');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (7777777777,  
'Ciro', 'Mattera');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (7839267343,  
'Gianfranco', 'Gianfranchi');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (8887775321,  
'Antonella', 'Antonelli');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (8888888888,  
'Matteo', 'Iacono');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (9874563210,  
'Federica', 'Amalfitano');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (9876543210,  
'Isabella', 'Di Iorio');
```

```
INSERT INTO OPERATORE_SANITARIO_RESPONSABILE  
(NUM_ISCRIZIONE_ORDINE_MEDICI, NOME, COGNOME) VALUES (9999999999,  
'Fabio', 'Natale');
```

10.3.9 Anamnesi

```
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (1, 1,  
TO_DATE('2021-07-22 11:05:17', 'YYYY-MM-DD HH24:MI:SS'), 8888888888);
```

```
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (2, 1,  
TO_DATE('2021-07-17 11:00:56', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);
```



```
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (3, 1,  
TO_DATE('2021-07-17 11:01:15', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (4, 1,  
TO_DATE('2021-07-17 11:01:38', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (5, 1,  
TO_DATE('2021-07-18 11:01:56', 'YYYY-MM-DD HH24:MI:SS'), 2222222222);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (6, 1,  
TO_DATE('2021-07-18 11:02:12', 'YYYY-MM-DD HH24:MI:SS'), 3333333333);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (6, 2,  
TO_DATE('2021-08-18 11:02:12', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (7, 1,  
TO_DATE('2021-07-19 11:02:28', 'YYYY-MM-DD HH24:MI:SS'), 4444444444);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (8, 1,  
TO_DATE('2021-07-20 11:02:43', 'YYYY-MM-DD HH24:MI:SS'), 5555555555);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (9, 1,  
TO_DATE('2021-07-20 11:03:00', 'YYYY-MM-DD HH24:MI:SS'), 6666666666);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (9, 2,  
TO_DATE('2021-09-18 11:05:20', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (10, 1,  
TO_DATE('2021-07-22 11:03:19', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (10, 2,  
TO_DATE('2021-09-20 12:47:19', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (11, 1,  
TO_DATE('2021-07-22 11:03:36', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);
```

```
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (12, 1,  
TO_DATE('2021-07-22 11:03:52', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (13, 1,  
TO_DATE('2021-07-22 11:04:05', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (14, 1,  
TO_DATE('2021-07-22 11:04:15', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (16, 1,  
TO_DATE('2021-08-31 11:23:36', 'YYYY-MM-DD HH24:MI:SS'), 7777777777);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (22, 1,  
TO_DATE('2021-09-20 12:59:21', 'YYYY-MM-DD HH24:MI:SS'), 5555555555);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (23, 1,  
TO_DATE('2021-11-19 13:10:00', 'YYYY-MM-DD HH24:MI:SS'), 5555555555);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (23, 2,  
TO_DATE('2021-11-29 13:14:59', 'YYYY-MM-DD HH24:MI:SS'), 5555555555);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (24, 1,  
TO_DATE('2021-11-19 13:20:00', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (24, 2,  
TO_DATE('2021-11-29 13:24:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (25, 1,  
TO_DATE('2021-11-19 13:30:00', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (25, 2,  
TO_DATE('2021-11-29 13:45:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (26, 1,  
TO_DATE('2021-11-19 13:45:00', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);
```



```
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (26, 2,  
TO_DATE('2021-11-29 13:45:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (27, 1,  
TO_DATE('2021-11-19 13:50:00', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (27, 2,  
TO_DATE('2021-11-21 13:50:00', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (28, 1,  
TO_DATE('2021-11-19 14:14:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (28, 2,  
TO_DATE('2021-11-21 14:14:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (29, 1,  
TO_DATE('2021-11-19 14:24:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (29, 2,  
TO_DATE('2021-11-21 14:24:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (30, 1,  
TO_DATE('2021-11-19 14:34:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (30, 2,  
TO_DATE('2021-11-21 14:34:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (31, 1,  
TO_DATE('2021-11-19 14:44:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (31, 2,  
TO_DATE('2021-11-21 14:44:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);  
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (32, 1,  
TO_DATE('2021-11-19 14:54:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);
```

```
INSERT INTO ANAMNESI (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, NUM_ISCRIZIONE_ORDINE_MEDICI) VALUES (32, 2,  
TO_DATE('2021-11-21 14:54:59', 'YYYY-MM-DD HH24:MI:SS'), 1111111111);
```

10.3.10 Idoneo

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (2, 1, TO_DATE('2021-07-17 11:00:56', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (4, 1, TO_DATE('2021-07-17 11:01:38', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (5, 1, TO_DATE('2021-07-18 11:01:56', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (6, 1, TO_DATE('2021-07-18 11:02:12', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (6, 2, TO_DATE('2021-08-18 11:02:12', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (7, 1, TO_DATE('2021-07-19 11:02:28', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (8, 1, TO_DATE('2021-07-20 11:02:43', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (9, 1, TO_DATE('2021-07-20 11:03:00', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (9, 2, TO_DATE('2021-09-18 11:05:20', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (10, 1, TO_DATE('2021-07-22 11:03:19', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (10, 2, TO_DATE('2021-09-20 12:47:19', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (12, 1, TO_DATE('2021-07-22 11:03:52', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (14, 1, TO_DATE('2021-07-22 11:04:15', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (16, 1, TO_DATE('2021-08-31 11:23:36', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (22, 1, TO_DATE('2021-09-20 12:59:21', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (27, 2, TO_DATE('2021-11-21 13:50:00', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (28, 2, TO_DATE('2021-11-21 14:14:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (29, 2, TO_DATE('2021-11-21 14:24:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (30, 2, TO_DATE('2021-11-21 14:34:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (31, 2, TO_DATE('2021-11-21 14:44:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO IDONEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (32, 2, TO_DATE('2021-11-21 14:54:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

10.3.11 Non Ideoneo temporaneo

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (3, 1,
```

```
TO_DATE('2021-07-17 11:01:15', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-08-16  
11:01:15', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (11, 1,  
TO_DATE('2021-07-22 11:03:36', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-08-31  
11:03:36', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (23, 1,  
TO_DATE('2021-11-19 13:10:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-29  
13:09:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (24, 1,  
TO_DATE('2021-11-19 13:20:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-29  
13:19:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (25, 1,  
TO_DATE('2021-11-19 13:30:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-29  
13:39:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (26, 1,  
TO_DATE('2021-11-19 13:45:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-29  
13:39:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (27, 1,  
TO_DATE('2021-11-19 13:50:00', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-21  
13:50:00', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (28, 1,  
TO_DATE('2021-11-19 14:14:59', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-21  
14:14:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (29, 1,  
TO_DATE('2021-11-19 14:24:59', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-21  
14:24:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (30, 1,
```

```
TO_DATE('2021-11-19 14:34:59', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-21  
14:34:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (31, 1,  
TO_DATE('2021-11-19 14:44:59', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-21  
14:44:59', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_TEMPORANEO (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO, DATA_IDONEITA_PREVISTA) VALUES (32, 1,  
TO_DATE('2021-11-19 14:54:59', 'YYYY-MM-DD HH24:MI:SS'), TO_DATE('2021-11-21  
14:54:59', 'YYYY-MM-DD HH24:MI:SS'));
```

10.3.12 Non Idoneo Permanente

```
INSERT INTO NON_IDONEO_PERMANENTE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (1, 1, TO_DATE('2021-07-22 11:05:17', 'YYYY-MM-  
DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_PERMANENTE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (13, 1, TO_DATE('2021-07-22 11:04:05', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_PERMANENTE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (23, 2, TO_DATE('2021-11-29 13:14:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_PERMANENTE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (24, 2, TO_DATE('2021-11-29 13:24:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_PERMANENTE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (25, 2, TO_DATE('2021-11-29 13:45:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO NON_IDONEO_PERMANENTE (NUM_PROTOCOLLO, NUM_TENTATIVI,  
DATA_ORA_CONTROLLO) VALUES (26, 2, TO_DATE('2021-11-29 13:45:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

10.3.13 Card Vaccinato

```
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (1, TO_DATE('2021-08-06 11:06:15', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-06 11:06:15', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (2, TO_DATE('2021-08-06 12:44:19', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-06 12:44:19', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (3, TO_DATE('2021-09-15 11:25:36', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-09-15 11:25:36', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (4, TO_DATE('2021-08-01 11:00:56', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-01 11:00:56', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (5, TO_DATE('2021-08-02 11:05:56', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-02 11:05:56', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (6, TO_DATE('2021-08-02 11:02:12', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-02 11:02:12', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (7, TO_DATE('2021-08-03 11:12:28', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-03 11:12:28', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (8, TO_DATE('2021-08-04 11:02:43', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-04 11:02:43', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (9, TO_DATE('2021-08-04 11:03:20', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-08-04 11:03:20', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (10, TO_DATE('2021-10-05 12:59:59', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-10-05 12:59:59', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (11, TO_DATE('2021-12-06 13:50:00', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-12-06 13:50:00', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (12, TO_DATE('2021-12-06 14:14:59', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-12-06 14:14:59', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (13, TO_DATE('2021-12-06 14:24:59', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-12-06 14:24:59', 'YYYY-MM-DD HH24:MI:SS'));
```



```
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (14, TO_DATE('2021-12-06 14:34:59', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-12-06 14:34:59', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (15, TO_DATE('2021-12-06 14:44:59', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-12-06 14:44:59', 'YYYY-MM-DD HH24:MI:SS'));  
INSERT INTO CARD_VACCINATO (NUM_CARTA, DATA_ATTIVAZIONE,  
DATA_SCADENZA) VALUES (16, TO_DATE('2021-12-06 14:54:59', 'YYYY-MM-DD  
HH24:MI:SS'), TO_DATE('2022-12-06 14:54:59', 'YYYY-MM-DD HH24:MI:SS'));
```

10.3.14 Luogo

L'autorità sanitaria ha il permesso di popolare tale tabella.

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,  
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO OSPEDALIERO) VALUES  
( 'Ospedale di Pozzuoli', '35', 'Napoli 2 Nord', 'NA', 'Pozzuoli', '80078', 'Via Domitiana',  
'12', '1', '231');
```

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,  
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO OSPEDALIERO) VALUES  
( 'San Leonardo', '53', 'Napoli 3 Sud', 'NA', 'Castellammare di Stabia', '80053', 'Viale  
Europa', '73', '1', '416');
```

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,  
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO OSPEDALIERO) VALUES  
( 'Termini', '6', 'Roma 1', 'RM', 'Roma', '00185', 'Piazza dei 500', '1', '0', null);
```

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,  
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO OSPEDALIERO) VALUES  
( 'San Camillo', '4', 'Roma 3', 'RM', 'Roma', '00152', 'Circonvallazione Gianicolense',  
'87', '1', '342');
```

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,  
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO OSPEDALIERO) VALUES  
( 'Fatebenefratelli Sacco', '1', 'Milano 1', 'MI', 'Milano', '20157', 'Corso di Porta  
Nuova', '23', '1', '233');
```

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,  
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO OSPEDALIERO) VALUES  
( 'Palazzetto dello Sport Ischia', '36', 'Napoli 2 Nord', 'NA', 'Ischia', '80077', 'Via Fondo  
Bosso', '8', '0', null);
```

```
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO_OSPEDALIERO) VALUES
('Palazzetto dello Sport Forio', '36', 'Napoli 2 Nord', 'NA', 'Forio', '80075', 'Via Casale',
'29', '0', null);
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO_OSPEDALIERO) VALUES
('Caserma Ferrari Orsi', '12', 'Caserta 1', 'CE', 'Caserta ', '81100', 'Via Capitano Luigi
Laviano', '8', '0', null);
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO_OSPEDALIERO) VALUES
('Pala Ud"a', '22', 'Chieti 2', 'CH', 'Chieti', '66100', 'Via dei Vestini', '31', '0', null);
INSERT INTO LUOGO (NOME, DISTRETTO, ASL_APPARTENENZA, PROVINCIA, CITTA,
CAP, VIA, CIVICO, DISCRIMINATORE_PVO, CODICE_PRESIDIO_OSPEDALIERO) VALUES
('Hub vaccinale Lingotto', '11', 'Torino 1', 'TO', 'Torino', '10126', 'Via Nizza', '1', '0',
null);
```

10.3.15 Stabilimento

L'autorità sanitaria ha il permesso di popolare tale tabella.

```
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('Vaccinemaker', 'Belgio', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('Vaccinedoers', 'Francia', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('WeMakeVaccines', 'Inghilterra', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('Vaccinators', 'Germania', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('LetPharma', 'Germania', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('Producers', 'Cina', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('PharmaMake', 'USA', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('PharmaVaccine', 'Italia', null, null, null);
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES
('Vaccinemaders', 'Italia', null, null, null);
```



```
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES  
( 'SuperVaccine', 'Svizzera', null, null, null);  
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES  
( 'PharmaMaker Inc', 'Austria', null, null, null);  
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES  
( 'RussiaVaccine', 'Russia', null, null, null);  
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES  
( 'VaccinePharmaceutics', 'Serbia', null, null, null);  
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES  
( 'USAPharma', 'USA', null, null, null);  
INSERT INTO STABILIMENTO (NOME, NAZIONE, CITTA, VIA, CIVICO) VALUES  
( 'Novartis', 'Italy', null, null, null);
```

10.3.16 Casa Produttrice

L'autorità sanitaria ha il permesso di popolare tale tabella.

```
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES  
( 'Astrazeneca', 'Svezia', 'www.astrazeneca.com');  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES ( 'Oxford',  
'Inghilterra', 'www.oxforduniversity.com');  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES ( 'Pfizer',  
'USA', 'www.pfizer.com');  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES  
( 'Biontech', 'Germania', 'www.biontech.com');  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES ( 'Janssen-  
Cilag International', 'Belgio', 'www.janssen.com');  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES  
( 'Moderna', 'USA', 'www.moderna.com');  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES  
( 'Gamaleja', 'Russia', null);  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES ( 'Novavax',  
'USA', null);  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES ( 'CEPI',  
'USA', null);  
INSERT INTO CASA_PRODUTTRICE (NOME, NAZIONE, SITO_WEB) VALUES  
( 'Universita Statale di Milano', 'Italia', 'www.unimi.it');
```

10.3.17 Lotto Vaccino

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('023342', 'Vaxzevira', TO_DATE('2021-12-  
10 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 500, 'Vaccinemaker', 'T', 60);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('098765', 'Vaxzevira', TO_DATE('2021-12-  
10 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 500, 'Vaccinedoers', 'T', 60);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('CMT123', 'Comirnaty', TO_DATE('2021-  
12-15 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100, 'WeMakeVaccines', 'T', 30);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('12345', 'Comirnaty', TO_DATE('2021-12-  
17 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 300, 'Vaccinators', 'T', 30);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('CMT6666', 'Comirnaty', TO_DATE('2021-  
09-29 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 110, 'Producers', 'T', 30);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('CMT1115', 'Comirnaty', TO_DATE('2021-  
10-25 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 230, 'Producers', 'T', 30);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('01299', 'Vaxzevira', TO_DATE('2021-12-  
24 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 50, 'Vaccinemaker', 'T', 60);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('01210', 'Vaxzevira', TO_DATE('2021-12-  
23 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100, 'PharmaMake', 'T', 60);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('JNS1111', 'Janssen', TO_DATE('2021-11-  
30 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 200, 'LetPharma', 'F', 0);
```

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('JNS1112', 'Janssen', TO_DATE('2021-08-  
01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100, 'LetPharma', 'F', 0);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('JNS1332', 'Janssen', TO_DATE('2021-10-  
12 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 50, 'PharmaVaccine', 'F', 0);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('MOD44431', 'Moderna COVID-19  
Vaccine', TO_DATE('2021-10-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 200,  
'Vaccinemaders', 'T', 40);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('MOD44441', 'Moderna COVID-19  
Vaccine', TO_DATE('2021-11-12 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 50,  
'SuperVaccine', 'T', 40);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('MOD44221', 'Moderna COVID-19  
Vaccine', TO_DATE('2021-10-18 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 120,  
'PharmaMaker Inc', 'T', 40);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('SPTN12345', 'Sputnik V',  
TO_DATE('2021-11-18 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 120, 'RussiaVaccine',  
'T', 21);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('SPTN12355', 'Sputnik V',  
TO_DATE('2021-12-21 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100,  
'VaccinePharmaceutics', 'T', 21);  
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA,  
NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE,  
TEMPO_PRIMA_SECONDA_DOSE) VALUES ('NV99999', 'NVX-CoV2373',
```

```
TO_DATE('2021-09-29 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 200, 'USAPharma', 'T', 30);
```

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA, NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE, TEMPO_PRIMA_SECONDA_DOSE) VALUES ('NV99998', 'NVX-CoV2373', TO_DATE('2021-10-28 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 200, 'USAPharma', 'T', 30);
```

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA, NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE, TEMPO_PRIMA_SECONDA_DOSE) VALUES ('NV99997', 'NVX-CoV2373', TO_DATE('2021-09-25 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 100, 'USAPharma', 'T', 30);
```

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA, NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE, TEMPO_PRIMA_SECONDA_DOSE) VALUES ('LCV12345', 'LeCoVax2', TO_DATE('2021-10-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 300, 'Novartis', 'T', 60);
```

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA, NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE, TEMPO_PRIMA_SECONDA_DOSE) VALUES ('LCV12346', 'LeCoVax2', TO_DATE('2021-11-02 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 300, 'Novartis', 'T', 60);
```

```
INSERT INTO LOTTO_VACCINO (NUM_LOTTO, NOME_FARMACO, SCADENZA, NUM_DOSI_LOTTO, NOME_STABILIMENTO, DISCRIMINATORE, TEMPO_PRIMA_SECONDA_DOSE) VALUES ('LCV12347', 'LeCoVax2', TO_DATE('2021-12-03 00:00:00', 'YYYY-MM-DD HH24:MI:SS'), 200, 'Novartis', 'T', 60);
```

10.3.18 Produttore di

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO, NUM_LOTTO) VALUES ('Astrazeneca', 'Vaxzevira', '01210');
```

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO, NUM_LOTTO) VALUES ('Astrazeneca', 'Vaxzevira', '01299');
```

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO, NUM_LOTTO) VALUES ('Astrazeneca', 'Vaxzevira', '098765');
```

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO, NUM_LOTTO) VALUES ('Biontech', 'Comirnaty', '12345');
```

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO, NUM_LOTTO) VALUES ('Biontech', 'Comirnaty', 'CMT1115');
```

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Biontech', 'Comirnaty', 'CMT123');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Biontech', 'Comirnaty', 'CMT6666');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('CEPI', 'NVX-CoV2373', 'NV99997');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('CEPI', 'NVX-CoV2373', 'NV99998');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('CEPI', 'NVX-CoV2373', 'NV99999');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Gamaleja', 'Sputnik V', 'SPTN12345');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Gamaleja', 'Sputnik V', 'SPTN12355');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Janssen-Cilag International', 'Janssen', 'JNS1111');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Janssen-Cilag International', 'Janssen', 'JNS1112');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Janssen-Cilag International', 'Janssen', 'JNS1332');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Moderna', 'Moderna COVID-19 Vaccine', 'MOD44221');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Moderna', 'Moderna COVID-19 Vaccine', 'MOD44431');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Moderna', 'Moderna COVID-19 Vaccine', 'MOD44441');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Novavax', 'NVX-CoV2373', 'NV99997');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Novavax', 'NVX-CoV2373', 'NV99998');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Novavax', 'NVX-CoV2373', 'NV99999');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Oxford', 'Vaxzevira', '01210');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Oxford', 'Vaxzevira', '01299');
```

```
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Oxford', 'Vaxzevira', '098765');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Pfizer', 'Comirnaty', '12345');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Pfizer', 'Comirnaty', 'CMT1115');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Pfizer', 'Comirnaty', 'CMT123');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Pfizer', 'Comirnaty', 'CMT6666');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Universita Statale di Milano', 'LeCoVax2', 'LCV12345');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Universita Statale di Milano', 'LeCoVax2', 'LCV12346');  
INSERT INTO PRODUTTORE_DI (NOME_PRODUTTORE, NOME_FARMACO,  
NUM_LOTTO) VALUES ('Universita Statale di Milano', 'LeCoVax2', 'LCV12347');
```

10.3.19 Consegnato a

```
INSERT INTO CONSEGNA_A (NOME_LUOGO, DISTRETTO_LUOGO,  
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNATE)  
VALUES ('San Leonardo', '53', 'Vaxzevira', '023342', TO_DATE('2021-07-02 12:21:15',  
'YYYY-MM-DD HH24:MI:SS'), 300);  
INSERT INTO CONSEGNA_A (NOME_LUOGO, DISTRETTO_LUOGO,  
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNATE)  
VALUES ('San Leonardo', '53', 'Vaxzevira', '023342', TO_DATE('2021-07-01 00:00:00',  
'YYYY-MM-DD HH24:MI:SS'), 200);  
INSERT INTO CONSEGNA_A (NOME_LUOGO, DISTRETTO_LUOGO,  
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNATE)  
VALUES ('San Leonardo', '53', 'Vaxzevira', '098765', TO_DATE('2021-07-01 00:00:00',  
'YYYY-MM-DD HH24:MI:SS'), 211);  
INSERT INTO CONSEGNA_A (NOME_LUOGO, DISTRETTO_LUOGO,  
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNATE)  
VALUES ('San Leonardo', '53', 'Vaxzevira', '098765', TO_DATE('2021-07-02 12:50:26',  
'YYYY-MM-DD HH24:MI:SS'), 29);  
INSERT INTO CONSEGNA_A (NOME_LUOGO, DISTRETTO_LUOGO,  
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNATE)
```



```
VALUES ('Ospedale di Pozzuoli', '35', 'Vaxzevira', '098765', TO_DATE('2021-07-02
12:52:49', 'YYYY-MM-DD HH24:MI:SS'), 100);
INSERT INTO CONSEGNAТО_A (NOME_LUOGO, DISTRETTO_LUOGO,
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNA)
VALUES ('Termini', '6', 'LeCoVax2', 'LCV12345', TO_DATE('2021-07-02 12:53:46',
'YYYY-MM-DD HH24:MI:SS'), 200);
INSERT INTO CONSEGNAТО_A (NOME_LUOGO, DISTRETTO_LUOGO,
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNA)
VALUES ('Fatebenefratelli Sacco', '1', 'LeCoVax2', 'LCV12345', TO_DATE('2021-07-02
12:54:12', 'YYYY-MM-DD HH24:MI:SS'), 100);
INSERT INTO CONSEGNAТО_A (NOME_LUOGO, DISTRETTO_LUOGO,
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNA)
VALUES ('Fatebenefratelli Sacco', '1', 'Comirnaty', 'CMT123', TO_DATE('2021-07-02
12:54:36', 'YYYY-MM-DD HH24:MI:SS'), 100);
INSERT INTO CONSEGNAТО_A (NOME_LUOGO, DISTRETTO_LUOGO,
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNA)
VALUES ('San Leonardo', '53', 'Janssen', 'JNS1332', TO_DATE('2021-07-02 12:55:36',
'YYYY-MM-DD HH24:MI:SS'), 50);
INSERT INTO CONSEGNAТО_A (NOME_LUOGO, DISTRETTO_LUOGO,
NOME_FARMACO, NUM_LOTTO, DATA_CONSEGNA, NUM_DOSI_CONSEGNA)
VALUES ('Hub vaccinale Lingotto', '11', 'Moderna COVID-19 Vaccine', 'MOD44431',
TO_DATE('2021-07-02 12:56:25', 'YYYY-MM-DD HH24:MI:SS'), 200);
```

10.3.20 Vaccinazione

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES
(TO_DATE('2021-07-22 11:06:15', 'YYYY-MM-DD HH24:MI:SS'), 'L', 1, 'San Leonardo',
'53', 'JNS1332', 'Janssen', 14, 1, TO_DATE('2021-07-22 11:04:15', 'YYYY-MM-DD
HH24:MI:SS'));
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES
(TO_DATE('2021-07-22 12:44:19', 'YYYY-MM-DD HH24:MI:SS'), 'L', 2, 'San Leonardo',
'53', '098765', 'Vaxzevira', 10, 1, TO_DATE('2021-07-22 11:03:19', 'YYYY-MM-DD
HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-24 12:49:19', 'YYYY-MM-DD HH24:MI:SS'), 'L', 2, 'San Leonardo',  
'53', '098765', 'Vaxzevira', 10, 2, TO_DATE('2021-09-20 12:47:19', 'YYYY-MM-DD  
HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-08-31 11:25:36', 'YYYY-MM-DD HH24:MI:SS'), 'L', 3, 'San Leonardo',  
'53', '098765', 'Vaxzevira', 16, 1, TO_DATE('2021-08-31 11:23:36', 'YYYY-MM-DD  
HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-07-17 11:00:56', 'YYYY-MM-DD HH24:MI:SS'), 'L', 4, 'Termini', '6',  
'LCV12345', 'LeCoVax2', 2, 1, TO_DATE('2021-07-17 11:00:56', 'YYYY-MM-DD  
HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-07-18 11:05:56', 'YYYY-MM-DD HH24:MI:SS'), 'L', 5, 'Termini', '6',  
'LCV12345', 'LeCoVax2', 5, 1, TO_DATE('2021-07-18 11:01:56', 'YYYY-MM-DD  
HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-07-18 11:02:12', 'YYYY-MM-DD HH24:MI:SS'), 'R', 6,  
'Fatebenefratelli Sacco', '1', 'CMT123', 'Comirnaty', 6, 1, TO_DATE('2021-07-18  
11:02:12', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-08-18 11:02:12', 'YYYY-MM-DD HH24:MI:SS'), 'R', 6,  
'Fatebenefratelli Sacco', '1', 'CMT123', 'Comirnaty', 6, 2, TO_DATE('2021-08-18  
11:02:12', 'YYYY-MM-DD HH24:MI:SS'));
```



```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-07-19 11:12:28', 'YYYY-MM-DD HH24:MI:SS'), 'R', 7,  
'Fatebenefratelli Sacco', '1', 'CMT123', 'Comirnaty', 7, 1, TO_DATE('2021-07-19  
11:02:28', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-07-20 11:02:43', 'YYYY-MM-DD HH24:MI:SS'), 'R', 8, 'Hub vaccinale  
Lingotto', '11', 'MOD44431', 'Moderna COVID-19 Vaccine', 8, 1, TO_DATE('2021-07-  
20 11:02:43', 'YYYY-MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-07-20 11:03:20', 'YYYY-MM-DD HH24:MI:SS'), 'R', 9, 'Ospedale di  
Pozzuoli', '35', '098765', 'Vaxzevira', 9, 1, TO_DATE('2021-07-20 11:03:00', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-09-18 11:07:20', 'YYYY-MM-DD HH24:MI:SS'), 'R', 9, 'Ospedale di  
Pozzuoli', '35', '098765', 'Vaxzevira', 9, 2, TO_DATE('2021-09-18 11:05:20', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-09-20 12:59:59', 'YYYY-MM-DD HH24:MI:SS'), 'L', 10, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 22, 1, TO_DATE('2021-09-20 12:59:21', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-21 13:50:00', 'YYYY-MM-DD HH24:MI:SS'), 'L', 11, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 27, 2, TO_DATE('2021-11-21 13:50:00', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-21 14:14:59', 'YYYY-MM-DD HH24:MI:SS'), 'L', 12, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 28, 2, TO_DATE('2021-11-21 14:14:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-21 14:24:59', 'YYYY-MM-DD HH24:MI:SS'), 'L', 13, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 29, 2, TO_DATE('2021-11-21 14:24:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-21 14:34:59', 'YYYY-MM-DD HH24:MI:SS'), 'L', 14, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 30, 2, TO_DATE('2021-11-21 14:34:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-21 14:44:59', 'YYYY-MM-DD HH24:MI:SS'), 'L', 15, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 31, 2, TO_DATE('2021-11-21 14:44:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

```
INSERT INTO VACCINAZIONE (DATA_ORA_EFFETTIVA, BRACCIO_INOCULAZIONE,  
NUM_CARTA, NOME_LUOGO, DISTRETTO_LUOGO, NUM_LOTTO, NOME_FARMACO,  
NUM_PROTOCOLLO, NUM_TENTATIVI, DATA_ORA_CONTROLLO) VALUES  
(TO_DATE('2021-11-21 14:54:59', 'YYYY-MM-DD HH24:MI:SS'), 'L', 16, 'San  
Leonardo', '53', '098765', 'Vaxzevira', 32, 2, TO_DATE('2021-11-21 14:54:59', 'YYYY-  
MM-DD HH24:MI:SS'));
```

10.4 Trigger

I trigger vengono utilizzati per controllare che vengano rispettati i vincoli dinamici (come ad esempio i vincoli temporali). Possono anche essere utilizzati per

implementare delle regole di business o per automatizzare operazioni, quali l'inserimento e l'aggiornamento di alcuni campi.

10.4.1 Check Age over18

Uno dei requisiti della base dati ci dice che il vaccinando deve essere maggiorenne per potersi vaccinare. Questo trigger controlla l'età del vaccinando in inserimento calcolando la sua età in base alla data di nascita ed alla data odierna, e se è minorenne alza un'eccezione.

```
CREATE OR REPLACE TRIGGER Check_Age_over18
  BEFORE INSERT OR UPDATE OF DATA_NASCITA ON VACCINANDO
  FOR EACH ROW

  BEGIN
    -- Se il vaccinando ha meno di 18 anni al giorno di inserimento, errore.
    IF trunc(months_between(sysdate,:NEW.DATA_NASCITA) / 12) < 18 THEN
      RAISE_APPLICATION_ERROR(-20000, 'Vaccinando troppo giovane!');
    end if;
  end;
```

10.4.2 Check Prenotazione Anamnesi Date

Uno dei molti vincoli di tempo presenti in questa base dati prevede che l'anamnesi, ovvero il controllo che certifica l'idoneità alla vaccinazione di un vaccinando, avvenga dopo la data prevista sulla prenotazione. Ciò significa che non ci si può presentare alla vaccinazione in momenti antecedenti la data prevista. Se la data in inserimento è errata o se non esistono prenotazioni, il trigger alza un'opportuna eccezione.

```
CREATE OR REPLACE TRIGGER Check_Prenotazione_Anamnesi_Date
  BEFORE INSERT OR UPDATE OF DATA_ORA_CONTROLLO ON ANAMNESI
  FOR EACH ROW
  DECLARE
    Data_Prenotazione          DATE;
    Data_Controllo             DATE;
    Data_non_valida_inserimento EXCEPTION;
```

```
Data_non_valida_aggiornamento EXCEPTION;

BEGIN
-- Si estrae (se esiste) la prenotazione
SELECT DATA_ORA_PREVISTE
INTO Data_Prenotazione
FROM PRENOTAZIONE
WHERE NUM_PROTOCOLLO = :NEW.NUM_PROTOCOLLO AND
NUM_TENTATIVI = :NEW.NUM_TENTATIVI;

-- Si controlla che la data dell'anamnesi sia dopo la data della prenotazione
IF (:NEW.DATA_ORA_CONTROLLO < Data_Prenotazione ) THEN
IF INSERTING THEN
RAISE Data_non_valida_inserimento;
end if;
IF UPDATING THEN
RAISE Data_non_valida_aggiornamento;
end if;
END IF;

EXCEPTION
WHEN no_data_found THEN
RAISE_APPLICATION_ERROR(-20021 , 'La prenotazione non esiste!');
WHEN Data_non_valida_inserimento THEN
RAISE_APPLICATION_ERROR(-20001, 'Data od ora del controllo invalida! (in
inserimento)');
WHEN Data_non_valida_aggiornamento THEN
RAISE_APPLICATION_ERROR(-20002, 'Data od ora del controllo invalida! (in
aggiornamento)');
end;
```

10.4.2 Check Vaccinazione Anamnesi Date

La vaccinazione non avviene allo stesso preciso istante dell'anamnesi, ma in momenti successivi. Per tale ragione, si hanno date separate per l'anamnesi e per la vaccinazione, dato che in Oracle DBMS la data e l'ora sono racchiuse nello stesso attributo. Si effettua pertanto lo stesso controllo di data effettuata tra anamnesi e

prenotazione, ma stavolta gli attori saranno vaccinazione ed anamnesi, con la data della vaccinazione che deve essere obbligatoriamente successiva alla data dell'anamnesi. Nel caso il vaccinando non risulti essere presente nella tabella idoneo o nel caso la data non sia valida, si alza un'eccezione.

```
CREATE OR REPLACE TRIGGER Check_Vaccinazione_Anamnesi_Date
  BEFORE INSERT OR UPDATE OF DATA_ORA_EFFETTIVA ON VACCINAZIONE
  FOR EACH ROW
  DECLARE
    data_controllo          IDONEO.DATA_ORA_CONTROLLO%TYPE;
    vaccinando_idoneo        NUMBER;
    vaccinando_non_e_idoneo  EXCEPTION;
    data_non_valida          EXCEPTION;

  BEGIN

    -- Si controlla se esiste l'anamnesi con risultato idoneo prima dell'inserimento
    in vaccinazione
      SELECT count(*)
      INTO vaccinando_idoneo
      FROM IDONEO
           WHERE IDONEO.NUM_PROTOCOLLO = :NEW.NUM_PROTOCOLLO AND
IDONEO.NUM_TENTATIVI = :NEW.NUM_TENTATIVI;

    -- Se non esiste, eccezione
    IF (vaccinando_idoneo = 0) THEN
      raise vaccinando_non_e_idoneo;
    end if;

    -- Si estrae la data dell'anamnesi
    SELECT DATA_ORA_CONTROLLO
    INTO data_controllo
      FROM IDONEO
           WHERE IDONEO.NUM_PROTOCOLLO = :NEW.NUM_PROTOCOLLO AND
IDONEO.NUM_TENTATIVI = :NEW.NUM_TENTATIVI;

    -- Si controlla che la vaccinazione avvenga dopo l'anamnesi
    IF (data_controllo > :NEW.DATA_ORA_EFFETTIVA) THEN
```

```
RAISE data_non_valida;  
end if;
```

EXCEPTION

```
WHEN vaccinando_non_e_idoneo THEN  
RAISE_APPLICATION_ERROR(-20050, 'Il vaccinando non risulta idoneo.');
```



```
WHEN data_non_valida THEN  
RAISE_APPLICATION_ERROR(-20051, 'La data che hai inserito non è valida.');
```



```
end;
```

10.4.2 Check NonIdoneoPerm Cant Be Vaccinated

Un vaccinando può avere tre risultati diversi dall'anamnesi: Idoneo, Non idoneo temporaneo e Non idoneo permanente. Se un vaccinando risulta Non idoneo permanente, non potrà mai vaccinarsi. Il seguente trigger gestisce questa politica, facendo sì che un vaccinando presente nella tabella non idoneo permanente non potrà mai vaccinarsi, alzando un'eccezione.

```
CREATE OR REPLACE TRIGGER Check_NonIdoneoPerm_Cant_Be_Vaccinated  
BEFORE INSERT OR UPDATE ON VACCINAZIONE  
FOR EACH ROW
```

```
DECLARE
```

```
Numero_Protocollo      NUMBER(8, 0);
```

```
Vaccinazione_Non_Valida  EXCEPTION;
```

```
BEGIN
```

```
-- Si estrae (se esiste) un non idoneo permanente
```

```
SELECT NUM_PROTOCOLLO
```

```
INTO Numero_Protocollo
```

```
FROM NON_IDONEO_PERMANENTE
```

```
where :NEW.NUM_PROTOCOLLO = NUM_PROTOCOLLO;
```

```
-- Se il numero protocollo in inserimento su vaccinazione è uguale al numero  
protocollo del non idoneo permanente, allora eccezione
```

```
IF (:NEW.NUM_PROTOCOLLO = Numero_Protocollo) THEN
```

```
RAISE Vaccinazione_Non_Valida;
```

```
end if;

EXCEPTION
WHEN no_data_found THEN
DBMS_OUTPUT.PUT_LINE('no_data_found. Il vaccinando non fa parte di Non
Idoneo Permanente.');
```

WHEN Vaccinazione_Non_Valida THEN
RAISE_APPLICATION_ERROR(-20007, 'Un non idoneo permanente non deve essere vaccinato!');

```
end;
```

10.4.3 Check NonIdoneoTemp Cant Be Vaccinated

Come già detto, uno dei tre risultati dell'anamnesi può essere Non idoneo temporaneo. Un vaccinando risultato tale potrebbe però risultare idoneo in un secondo momento. Si gestisce quindi tramite trigger il caso in cui si provi a vaccinare un vaccinando la cui anamnesi in quel momento lo definisce Non idoneo temporaneo. Se risulta tale, si alza un'eccezione.

```
CREATE OR REPLACE TRIGGER Check_NonIdoneoTemp_Cant_Be_Vaccinated
BEFORE INSERT OR UPDATE ON VACCINAZIONE
FOR EACH ROW
DECLARE
exists_in_nit          NUMBER;
Numero_Protocollo     NUMBER(8, 0);
Numero_Tentativi      NUMBER(2, 0);
Vaccinazione_Non_Valida EXCEPTION;

BEGIN
-- Si controlla se esiste il vaccinando in inserimento nella tabella
NON_IDONEO_TEMPORANEO.
SELECT count(*)
INTO exists_in_nit
FROM NON_IDONEO_TEMPORANEO
WHERE :NEW.NUM_PROTOCOLLO = NUM_PROTOCOLLO AND
:NEW.NUM_TENTATIVI = NUM_TENTATIVI;
```

-- Se esiste, errore.

IF (exists_in_nit <> 0) THEN

RAISE Vaccinazione_Non_Valida;

end if;

EXCEPTION

WHEN Vaccinazione_Non_Valida THEN

RAISE_APPLICATION_ERROR(-20007, 'Un non idoneo temporaneo non deve essere vaccinato!');

end;

10.4.5 Check Seconda Dose Date

Secondo le indicazioni di EMA ed AIFA, la seconda dose di vaccino deve essere somministrata dopo un periodo di tempo prefissato. Si gestisce tramite trigger tale politica. Se si provasse a vaccinare un vaccinando prima della data prevista, il trigger alza un'eccezione.

CREATE OR REPLACE TRIGGER Check_Seconda_Dose_Date

BEFORE INSERT OR UPDATE OF DATA_ORA_EFFETTIVA ON VACCINAZIONE
FOR EACH ROW

DECLARE

Tempo_Prima_Seconda

LOTTO_VACCINO.TEMPO_PRIMA_SECONDA_DOSE%TYPE;

Data_Effettiva DATE;

NumCartaExists NUMBER;

NumCarta CARD_VACCINATO.NUM_CARTA%TYPE;

Data_non_valida_inserimento EXCEPTION;

Data_non_valida_aggiornamento EXCEPTION;

BEGIN

-- Si controlla se esiste la card vaccinato (in caso di esistenza, il vaccinando ha fatto una dose di vaccino)

SELECT count(*)

INTO NumCartaExists

FROM CARD_VACCINATO


```
WHERE :NEW.NUM_CARTA = NUM_CARTA;

-- Se la carta esiste, la si estrae
IF (NumCartaExists <> 0) THEN
SELECT NUM_CARTA
    INTO NumCarta
    FROM CARD_VACCINATO
    WHERE :NEW.NUM_CARTA = NUM_CARTA;

-- Si estrae il tempo che decorre tra la prima e la seconda dose
SELECT TEMPO_PRIMA_SECONDA_DOSE, DATA_ORA_EFFETTIVA
    INTO Tempo_Prima_Seconda, Data_Effettiva
    FROM LOTTO_VACCINO
    JOIN VACCINAZIONE V on LOTTO_VACCINO.NUM_LOTTO =
V.NUM_LOTTO and LOTTO_VACCINO.NOME_FARMACO = V.NOME_FARMACO
    WHERE DISCRIMINATORE = 'T' AND NUM_CARTA = NumCarta;

-- Se la data in inserimento su vaccinazione è minore del decorso
prima/seconda dose, allora eccezione
IF (:NEW.DATA_ORA_EFFETTIVA < Tempo_Prima_Seconda + Data_Effettiva)
THEN
    IF INSERTING THEN
        RAISE Data_non_valida_inserimento;
    end if;
    IF UPDATING THEN
        RAISE Data_non_valida_aggiornamento;
    end if;
end if;
end if;

EXCEPTION
WHEN no_data_found THEN
DBMS_OUTPUT.PUT_LINE( 'Il vaccino non risulta essere a doppia dose. ');
WHEN Data_non_valida_inserimento THEN
    RAISE_APPLICATION_ERROR(-20005, 'Data od ora della vaccinazione invalida!
(in inserimento)');
```

```
    WHEN Data_non_valida_aggiornamento THEN  
    RAISE_APPLICATION_ERROR(-20006, 'Data od ora della vaccinazione invalida!  
(in aggiornamento)');  
end;
```

10.4.6 Check Consegna Vaccini

All'inserimento di una vaccinazione, è richiesto indicare anche quale vaccino è stato utilizzato ed in quale luogo si è tenuta la seduta di vaccinazione. È facile intuire che se un dato vaccino non è stato consegnato in un luogo, esso non può essere utilizzato. Questo trigger gestisce tale politica, controllando se il vaccino in inserimento su vaccinazione è stato consegnato nel luogo identificato. Se così non fosse, il trigger alza un'eccezione.

```
CREATE OR REPLACE TRIGGER Check_Consegna_Vaccini  
    BEFORE INSERT OR UPDATE OF NOME_FARMACO, NUM_LOTTO ON  
VACCINAZIONE  
    FOR EACH ROW  
    DECLARE  
    farmaco_exists    NUMBER;  
    farmaco_nonesiste    EXCEPTION;  
    BEGIN  
    -- Si controlla se il farmaco è stato consegnato in un dato luogo  
    SELECT count(*)  
    INTO farmaco_exists  
        FROM CONSEGNATO_A CONS  
        WHERE :NEW.NOME_LUOGO = CONS.NOME_LUOGO AND  
:NEW.DISTRETTO_LUOGO = CONS.DISTRETTO_LUOGO AND :NEW.NOME_FARMACO  
= CONS.NOME_FARMACO AND :NEW.NUM_LOTTO = CONS.NUM_LOTTO;  
  
    -- Se non esiste, errore  
    IF (farmaco_exists = 0) THEN  
    raise farmaco_nonesiste;  
    end if;  
  
    EXCEPTION
```

```
    WHEN farmaco_nonesiste THEN  
        RAISE_APPLICATION_ERROR(-20008, 'Il lotto di vaccino che si sta cercando di  
usare non esiste nel luogo desiderato.');
```

end;

10.4.6 Vaccino Scaduto

Ovviamente un vaccino, come ogni altro farmaco, ha una data di scadenza, e come ogni altro farmaco non può essere somministrato se quest'ultimo è scaduto. Il seguente trigger gestisce questa situazione, evitando che venga registrata una vaccinazione con un vaccino scaduto. Si controlla se la data in inserimento è successiva alla data di scadenza, ed in caso affermativo si alza un'eccezione.

```
CREATE OR REPLACE TRIGGER vaccinoScaduto  
    BEFORE INSERT OR UPDATE OF DATA_ORA_EFFETTIVA ON VACCINAZIONE  
    FOR EACH ROW  
    DECLARE  
  
    check_data          LOTTO_VACCINO.SCADENZA%TYPE;  
  
    BEGIN  
        -- Si estrae la scadenza del lotto di vaccino che si sta utilizzando  
        SELECT SCADENZA  
        INTO CHECK_DATA  
            FROM LOTTO_VACCINO LV  
            WHERE :NEW.NOME_FARMACO = LV.NOME_FARMACO AND  
:NEW.NUM_LOTTO = LV.NUM_LOTTO;  
  
        -- Se il vaccino è scaduto, errore  
        IF (:NEW.DATA_ORA_EFFETTIVA > check_data) then  
            RAISE_APPLICATION_ERROR(-20080, 'Non puoi usare un vaccino scaduto!!!');  
        end if;  
  
    end;
```

10.5 Procedure

Le procedure vengono utilizzate per implementare le logiche di business, semplificare le operazioni degli utenti ed automatizzare la base dati.

10.5.1 create Vaccinando

Essendo questo un database per una campagna vaccinale (nel caso specifico, la campagna vaccinale contro la COVID19), tutte le operazioni ruotano attorno al vaccinando, alle vaccinazioni ed ai vaccini utilizzati.

Senza il vaccinando, non possiamo effettuare vaccinazioni, dato che non si ha nessuno a cui somministrare il vaccino.

La seguente procedura automatizza l'inserimento nelle tabelle Vaccinando, Telefono, Disabile, Disabilità, Fragile, Patologia e Prenotazione.

La procedura riceve in input:

- I dati relativi al Vaccinando, ovvero: tessera sanitaria e scadenza, codice fiscale, nome e cognome, data di nascita e città di nascita, residenza (città, provincia, cap, via, civico), informazioni di contatto (email e numeri di telefono, questi ultimi passati tramite array), eventuale positività pregressa alla COVID19;
- I dati relativi alle eventuali patologie o disabilità del vaccinando (numero attestato 104, disabilità, patologie, questi due ultimi passati tramite array);

Con questi dati, si genera il vaccinando e si popolano le tabelle figlie di quest'ultimo e la tabella per l'attributo multivalore Telefono.

Viene inoltre generata la prenotazione in base alla positività pregressa, secondo la regola di business che una persona risultata positiva in passato deve ricevere una sola dose di vaccino.

```
create or replace type disArr is varray(10) of varchar2(40);  
create or replace type patArr is varray(10) of varchar2(40);  
create or replace type NumTelArr is varray(5) of varchar2(20);
```

```
create or replace procedure create_Vaccinando(  
  num_Tessera_p  vaccinando.num_tessera_sanitaria%type,  
  scad_Tessera_p vaccinando.scadenza_tessera_sanitaria%type,  
  cod_fiscale_p   vaccinando.codice_fiscale%type,
```

```
nome_p          vaccinando.nome%type,
cognome_p       vaccinando.cognome%type,
data_nascita_p  vaccinando.data_nascita%type,
citta_natale_p  vaccinando.citta_natale%type,
citta_p         vaccinando.citta%type,
provincia_p     vaccinando.provincia%type,
cap_p          vaccinando.cap%type,
via_p          vaccinando.via%type,
civico_p        vaccinando.civico%type,
email_p        vaccinando.e_mail%type,
pos_preg_p      vaccinando.positivita_pregressa%type,

disc_Dis_p      number,
num_Attes_104_p disabile.num_attestato_104%type,
disabilita_p    disArr,

disc_Pat_p      number,
patologia_p     patArr,

numero_telefono_p NumTelArr
)
is
prenotazioni_esistenti number;
vaccinando_exists      number;
vaccinando_gia_esiste  exception;

max_protocollo      PRENOTAZIONE.NUM_PROTOCOLLO%TYPE;
max_data            DATE;
pos_preg_errata     EXCEPTION;

begin
    -- Si controlla se il vaccinando esiste
    select count(*)
    into vaccinando_exists
    from vaccinando
        where num_tessera_sanitaria = num_Tessera_p;
```

```
if (vaccinando_exists <> 0) then  
raise vaccinando_gia_esiste;  
end if;
```

-- Si estrae il numero delle prenotazioni esistenti. Servirà per creare il nuovo numero di prenotazione.

```
SELECT count(*)  
into prenotazioni_esistenti  
from PRENOTAZIONE;
```

-- Se non esistono prenotazioni, si crea la prima prenotazione con data una settimana successiva alla data di sistema.

```
IF (prenotazioni_esistenti = 0) THEN  
max_protocollo := 0;  
max_data := sysdate + 7;
```

-- Se esistono prenotazioni, si procede incrementando di 1 l'ultimo valore di prenotazione.

```
ELSE  
SELECT MAX(NUM_PROTOCOLLO)  
INTO max_protocollo  
FROM PRENOTAZIONE;
```

```
SELECT MAX(DATA_ORA_PREVISTE)  
INTO max_data  
FROM PRENOTAZIONE;
```

```
end if;
```

-- Se si ha positività pregressa la prenotazione è a singola dose, altrimenti, a doppia dose.

```
IF (pos_preg_p = 'T') THEN  
INSERT INTO PRENOTAZIONE VALUES (max_protocollo+1, 1, max_data +  
(1/1440*10), 'SINGOLA');  
ELSIF (pos_preg_p = 'F') THEN
```

```
INSERT INTO PRENOTAZIONE VALUES (max_protocollo+1, 1, max_data +  
(1/1440*10), 'DOPPIA');
```

```
ELSE
```

```
RAISE pos_preg_errata;
```

```
end if;
```

-- Si inserisce il vaccinando.

```
insert into vaccinando values(  
    num_Tessera_p,  
    scad_Tessera_p,  
    cod_fiscale_p,  
    nome_p,  
    cognome_p,  
    data_nascita_p,  
    citta_natale_p,  
    citta_p,  
    provincia_p,  
    cap_p,  
    via_p,  
    civico_p,  
    email_p,  
    pos_preg_p,  
    max_protocollo+1,  
    1  
);
```

-- Si controlla se è stato passato un vaccinando disabile come parametro della procedura. Se sì, si procede ad inserirlo in DISABILE e ad inserire tutte le sue disabilità.

```
if(disc_Dis_p <> 0) then  
insert into disabile values (  
    num_Attes_104_p,  
    num_Tessera_p  
);
```

```
for i in 1..disabilita_p.count loop
```

```
insert into disabilita values (  
    num_Tessera_p,  
    disabilita_p(i)  
);  
end loop;  
end if;
```

-- Si controlla se è stato inserito un vaccinando fragile come parametro della procedura. Se sì, si procede ad inserirlo in FRAGILE e ad inserire tutte le sue patologie.

```
if(disc_Pat_p <> 0) then  
insert into fragile values (  
    num_Tessera_p  
);
```

```
for i in 1..patologia_p.count loop  
insert into patologia values (  
    num_Tessera_p,  
    patologia_p(i)  
);  
end loop;  
end if;
```

-- Se sono stati passati dei numeri di telefono in input alla procedura, si procede ad inserirli nella tabella TELEFONO.

```
if(numero_telefono_p is not NULL) then  
for i in 1.. numero_telefono_p.count loop  
insert into telefono values (  
    num_Tessera_p,  
    numero_telefono_p(i)  
);  
  
end loop;  
end if;
```

```
commit;
```



```
exception  
when vaccinando_gia_esiste then  
raise_application_error(-20015, 'Il vaccinando esiste già nel database!');  
rollback;  
  
WHEN pos_preg_errata then  
raise_application_error(-20060, 'Inserisci una positività pregressa valida!');  
rollback;
```

```
end;
```

10.5.2 createAnamnesi

Ogni vaccinando prima di potersi vaccinare deve sottoporsi all'anamnesi, ovvero un controllo per attestare l'idoneità del candidato alla vaccinazione. La procedura seguente automatizza il processo di creazione dell'anamnesi e di inserimento dei dati nelle tabelle figlie, le quali vengono usate per identificare i risultati del controllo.

La procedura riceve in input:

- Il codice fiscale del vaccinando;
- Il numero di iscrizione ordine dei medici del medico responsabile della vaccinazione;
- la data effettiva del controllo;
- l'esito dell'anamnesi;
- eventuale data di idoneità prevista, nel caso in cui il vaccinando dovesse risultare Non Idoneo Temporaneo.

Si procede quindi a generare la t-upla per la tabella Anamnesi, e si genera inoltre una t-upla per l'esito nell'apposita tabella, che può essere Idoneo, Non Idoneo Temporaneo o Non Idoneo Permanente.

Nel caso in cui un vaccinando risulti Non Idoneo Temporaneo, si genera inoltre una nuova prenotazione in base alla data di idoneità prevista, data dal medico.

```
CREATE OR REPLACE PROCEDURE createAnamnesi(  
cod_fisc_p          VACCINANDO.CODICE_FISCALE%TYPE,  
num_isc_medici_p    ANAMNESI.NUM_ISCRIZIONE_ORDINE_MEDICI%TYPE,
```

```
data_controllo_p      ANAMNESI.DATA_ORA_CONTROLLO%TYPE,  
esito_p              NUMBER,  
idoneita_prevista_p   DATE  
)  
IS  
    vaccinando_exists    NUMBER;  
    operatore_exists     NUMBER;  
    prenotazione_exists  NUMBER;  
    vaccinando_non_esiste EXCEPTION;  
    operatore_non_esiste EXCEPTION;  
    prenotazione_non_esiste EXCEPTION;  
    num_protocollo_v     PRENOTAZIONE.NUM_PROTOCOLLO%TYPE;  
    num_tentativi_v      PRENOTAZIONE.NUM_TENTATIVI%TYPE;  
    tipo_prenotazione_v  PRENOTAZIONE.TIPO_PRENOTAZIONE%TYPE;  
    data_idoneita_assente EXCEPTION;  
    esito_sconosciuto    EXCEPTION;  
    data_controllo_valida DATE;  
    anamnesi_check       NUMBER;  
    anamnesi_gia_esiste  EXCEPTION;  
  
BEGIN  
    -- Si controlla se esiste il vaccinando  
    SELECT count(*)  
    INTO vaccinando_exists  
    FROM VACCINANDO  
        WHERE CODICE_FISCALE = cod_fisc_p;  
  
    IF (vaccinando_exists = 0) THEN  
        raise vaccinando_non_esiste;  
    end if;  
  
    -- Si controlla se esiste l'operatore sanitario responsabile  
    SELECT count(*)  
    INTO operatore_exists  
    FROM OPERATORE_SANITARIO_RESPONSABILE  
        WHERE NUM_ISCRIZIONE_ORDINE_MEDICI = num_isc_medici_p;
```

```
IF (operatore_exists = 0) THEN  
raise operatore_non_esiste;  
end if;
```

-- Si ricavano il numero di prenotazione ed il numero di rimodulazioni
massimo

```
SELECT PRENOTAZIONE.NUM_PROTOCOLLO,  
MAX(PRENOTAZIONE.NUM_TENTATIVI), TIPO_PRENOTAZIONE  
INTO num_protocollo_v, num_tentativi_v, tipo_prenotazione_v  
FROM PRENOTAZIONE JOIN VACCINANDO  
ON PRENOTAZIONE.NUM_PROTOCOLLO =  
VACCINANDO.NUM_PROTOCOLLO  
WHERE CODICE_FISCALE = cod_fisc_p  
group by PRENOTAZIONE.NUM_PROTOCOLLO,  
PRENOTAZIONE.TIPO_PRENOTAZIONE;
```

-- Si controlla se è stata fornita una data di anamnesi

```
IF (data_controllo_p IS NULL) THEN  
data_controllo_valida := sysdate;  
ELSE  
data_controllo_valida := data_controllo_p;  
end if;
```

-- Si controlla se è stata effettuata l'anamnesi per l'ultima prenotazione

```
SELECT count(*)  
into anamnesi_check  
FROM ANAMNESI  
WHERE NUM_PROTOCOLLO = num_protocollo_v and NUM_TENTATIVI =  
num_tentativi_v;
```

```
if (anamnesi_check <> 0) then  
raise anamnesi_gia_esiste;  
end if;
```

-- Si inserisce nell'anamnesi

```
INSERT INTO ANAMNESI VALUES (
```

```
        num_protocollo_v,  
        num_tentativi_v,  
        data_controllo_valida,  
        num_isc_medici_p  
    );  
  
-- Si controlla l'esito  
-- Idoneo  
IF (esito_p = 0) THEN  
INSERT INTO IDONEO VALUES (  
    num_protocollo_v,  
    num_tentativi_v,  
    data_controllo_valida  
);  
  
-- Non Idoneo Permanente  
ELSIF (esito_p = 2) THEN  
INSERT INTO NON_IDONEO_PERMANENTE VALUES (  
    num_protocollo_v,  
    num_tentativi_v,  
    data_controllo_valida  
);  
  
-- Non Idoneo Temporaneo  
ELSIF (esito_p = 1) THEN  
IF (idoneita_prevista_p IS NULL) THEN  
raise data_idoneita_assente;  
end if;  
  
INSERT INTO NON_IDONEO_TEMPORANEO VALUES (  
    num_protocollo_v,  
    num_tentativi_v,  
    data_controllo_valida,  
    idoneita_prevista_p  
);  
  
-- Si genera la nuova prenotazione  
INSERT INTO PRENOTAZIONE VALUES (  
    num_protocollo_v,  
    num_tentativi_v+1,
```

```
        idoneita_prevista_p,  
        tipo_prenotazione_v  
    );  
  
    -- Se viene passato un valore errato  
    ELSE  
        raise esito_sconosciuto;  
    end if;  
  
    EXCEPTION  
    WHEN vaccinando_non_esiste THEN  
        RAISE_APPLICATION_ERROR(-20022, 'Il vaccinando non esiste!');  
        rollback;  
  
    WHEN operatore_non_esiste THEN  
        RAISE_APPLICATION_ERROR(-20023, 'Operatore Sanitario non esiste!');  
        rollback;  
  
    WHEN data_idoneita_assente THEN  
        RAISE_APPLICATION_ERROR(-20024, 'Devi fornire una data di idoneità!');  
        rollback;  
  
    WHEN esito_sconosciuto THEN  
        RAISE_APPLICATION_ERROR(-20025, 'Esito sconosciuto! Inserisci un valore di  
esito valido: 0 Idoneo, 1 Non idoneo temporaneo, 2 Non idoneo permanente.');
```

```
        rollback;  
  
    WHEN anamnesi_gia_esiste THEN  
        RAISE_APPLICATION_ERROR(-20062, 'Questa anamnesi è già stata  
effettuata.');
```

```
        rollback;  
    end;
```

10.5.3 insertVaccino

Senza vaccini non possono essere effettuate vaccinazioni, per cui si ha ovviamente interesse nel dare la possibilità di registrare i vaccini presenti in Italia, e consegnarli successivamente (tramite un'altra procedura apposita).

Questa procedura automatizza l'inserimento sulle tabelle Vaccino, Stabilimento e Produttore_di.

È necessario che le case produttrici del vaccino siano già presenti nel database.

La procedura prende in input:

- Informazioni sul vaccino (Nome, lotto, scadenza e numero di dosi del lotto);
- discriminatore per determinare se è un vaccino one-shot (ad esempio il vaccino Janssen) o un vaccino a doppia dose (ad esempio il Comirnaty);
- eventuale periodo di tempo (in giorni) che deve intercorrere tra la prima e la seconda dose (0 nel caso dei vaccini one shot);
- dati relativi allo stabilimento produttore (nome, nazione, indirizzo);
- array contenente i nomi delle case produttrici.

La procedura inserisce le t-uple su Vaccino, genera la t.upla dello stabilimento produttore se non esiste già, e controlla se le case produttrici sono esistenti e procede ad associare il vaccino inserito a queste ultime.

```
CREATE OR REPLACE TYPE caseprodarr IS VARRAY(5) OF VARCHAR(30);
```

```
CREATE OR REPLACE PROCEDURE insertVaccino(  
-- Parametri per l'inserimento su Lotto_Vaccino  
nome_farmaco_p LOTTO_VACCINO.NOME_FARMACO%TYPE,  
num_lotto_p LOTTO_VACCINO.NUM_LOTTO%TYPE,  
scadenza_p LOTTO_VACCINO.SCADENZA%TYPE,  
num_dosi_p LOTTO_VACCINO.NUM_DOSI_LOTTO%TYPE,  
  
discriminatore_p LOTTO_VACCINO.DISCRIMINATORE%TYPE,  
tempo_prima_sec_p LOTTO_VACCINO.TEMPO_PRIMA_SECONDA_DOSE%TYPE  
default 0,  
  
-- Parametri per l'inserimento dello stabilimento produttore  
nome_stab_p STABILIMENTO.NOME%TYPE,  
nazione_stab_p STABILIMENTO.NAZIONE%TYPE,
```

```
citta_stab_p STABILIMENTO.CITTA%TYPE,  
via_stab_p STABILIMENTO.VIA%TYPE,  
civico_stab_p STABILIMENTO.CIVICO%TYPE,  
  
-- Parametri per l'inserimento della casa produttrice  
nome_casa_prod_p caseprodarr  
)  
IS  
    nome_casa_prod_i CASA_PRODUTTRICE.NOME%TYPE;  
    stab_exists number;  
BEGIN  
  
    -- Si controlla se lo stabilimento passato in input alla procedura esiste già a  
    database. Se non esiste, lo si inserisce.  
    SELECT COUNT(*)  
    INTO stab_exists  
    FROM STABILIMENTO  
        WHERE NOME = nome_stab_p;  
  
    IF (stab_exists = 0) THEN  
        INSERT INTO STABILIMENTO VALUES (  
            nome_stab_p,  
            nazione_stab_p,  
            citta_stab_p,  
            via_stab_p,  
            civico_stab_p  
        );  
    end if;  
  
    -- Si inserisce il lotto di vaccino nella tabella dedicata.  
    INSERT INTO LOTTO_VACCINO VALUES (num_lotto_p,  
        nome_farmaco_p,  
        scadenza_p,  
        num_dosi_p,  
        nome_stab_p,  
        discriminatore_p,  
        tempo_prima_sec_p);
```

-- Se vengono passate delle case produttrici in input, si controlla se esistono effettuando la select sottostante. Se non esistono, errore.

```
IF (nome_casa_prod_p IS NOT NULL) THEN
for i in 1 .. nome_casa_prod_p.count loop
    SELECT NOME
    INTO nome_casa_prod_i
    FROM CASA_PRODUTTRICE
    WHERE NOME = nome_casa_prod_p(i);
```

-- Se la select non alza no_data_found, allora si associa il vacicno con le case produttrici.

```
INSERT INTO PRODUTTORE_DI VALUES (
    nome_casa_prod_p(i),
    nome_farmaco_p,
    num_lotto_p
);

end loop;
end if;
COMMIT;

EXCEPTION
WHEN no_data_found THEN
RAISE_APPLICATION_ERROR(-20009, 'Una o più case produttrici scelte non
sono presenti nel database.');
```

rollback;

end;

10.5.4 consegnaVaccino

Come già accennato in precedenza, un vaccino non può essere utilizzato in una vaccinazione se non è stato consegnato nel luogo in cui si svolge quest'ultima. Per tale ragione, è di forte importanza la procedura di consegna di un vaccino in un luogo.

Tale procedura prende in input:

- nome e distretto del luogo in cui si vogliono consegnare delle dosi di vaccino;

- nome e lotto del vaccino consegnato;
- data della consegna;
- numero di dosi da consegnare;

Si controlla che il numero di dosi da consegnare sia un valore valido e se il lotto che si sta provando a consegnare sia esistente nel database. Si controlla inoltre se esiste il luogo di consegna.

Se l'attributo data di consegna viene lasciato nullo, si utilizza la data di inserimento della t-upla come data di consegna.

Se sono già state consegnate delle dosi di questo vaccino, si controlla quante altre dosi sono consegnabili e si alza un'eccezione se il lotto è stato completamente consegnato o se le dosi che si provano a consegnare sono maggiori delle dosi consegnabili. Se non sono state consegnate delle dosi in precedenza, si controlla solo che il numero di dosi in inserimento sia valido.

Se esiste una consegna con le stesse caratteristiche (eccetto il numero di dosi), si va ad aggiornare il numero di dosi di quella consegna, altrimenti si crea la nuova consegna.

```
CREATE OR REPLACE PROCEDURE consegnaVaccino(  
  nome_luogo_p          LUOGO.NOME%TYPE,  
  distretto_p          LUOGO.DISTRETTO%TYPE,  
  farmaco_p            LOTTO_VACCINO.NOME_FARMACO%TYPE,  
  lotto_p              LOTTO_VACCINO.NUM_LOTTO%TYPE,  
  data_consegna_p      DATE,  
  num_dosi_da_consegnare_p  NUMBER  
)  
IS  
  negative_doses        EXCEPTION;  
  vaccino_exists        NUMBER;  
  vaccino_non_esiste    EXCEPTION;  
  luogo_exists          NUMBER;  
  luogo_non_esiste     EXCEPTION;  
  dosi_exists           NUMBER;  
  dosi_gia_consegnate   NUMBER;  
  dosi_consegnabili     NUMBER;  
  dosi_consegnabili_real  NUMBER;  
  lotto_consegnato      EXCEPTION;
```

```
dosi_eccessive          EXCEPTION;  
data_consegna_real      DATE;  
consegna_exists         NUMBER;  
num_dosi_esistenti_specifiche  NUMBER;
```

```
BEGIN
```

```
-- Se si provano a consegnare 0 o meno dosi, errore.
```

```
IF (num_dosi_da_consegnare_p < 1) THEN
```

```
RAISE negative_doses;
```

```
end if;
```

```
-- Si controlla se il lotto di vaccino esiste nel database
```

```
SELECT count(*)
```

```
INTO vaccino_exists
```

```
FROM LOTTO_VACCINO
```

```
WHERE NOME_FARMACO = farmaco_p AND NUM_LOTTO = lotto_p;
```

```
if (vaccino_exists = 0) THEN
```

```
RAISE vaccino_non_esiste;
```

```
end if;
```

```
-- Si controlla se il luogo esiste nel database
```

```
SELECT count(*)
```

```
INTO luogo_exists
```

```
FROM LUOGO
```

```
WHERE NOME = nome_luogo_p AND DISTRETTO = distretto_p;
```

```
IF (luogo_exists = 0) THEN
```

```
raise luogo_non_esiste;
```

```
end if;
```

```
-- Si controlla il numero di dosi del lotto
```

```
SELECT NUM_DOSI_LOTTO
```

```
INTO dosi_consegnabili
```

```
FROM LOTTO_VACCINO
```

```
WHERE NOME_FARMACO = farmaco_p AND NUM_LOTTO = lotto_p;
```

-- Si controlla quale data di consegna usare

IF (data_consegna_p IS NULL) THEN

data_consegna_real := sysdate;

ELSE

data_consegna_real := data_consegna_p;

end if;

-- Si controlla se sono già state consegnate delle dosi di vaccino e si agisce di conseguenza.

SELECT count(*)

INTO dosi_exists

FROM CONSEGNATO_A

WHERE NUM_LOTTO = lotto_p AND NOME_FARMACO = farmaco_p;

-- Se sono state già consegnate delle dosi in database:

IF (dosi_exists <> 0) THEN

SELECT sum(ALL NUM_DOSI_CONSEGNATE)

INTO dosi_gia_consegnate

FROM CONSEGNATO_A

WHERE NOME_FARMACO = farmaco_p AND NUM_LOTTO = lotto_p;

-- Se il lotto è stato già consegnato completamente

IF (dosi_consegnabili = dosi_gia_consegnate) THEN

RAISE lotto_consegnato;

end if;

-- Si definisce il numero di dosi effettivamente consegnabili

dosi_consegnabili_real := dosi_consegnabili - dosi_gia_consegnate;

-- Se si provano a consegnare troppe dosi

IF (dosi_consegnabili_real < num_dosi_da_consegnare_p) THEN

RAISE dosi_eccessive;

end if;

-- Se non sono state già consegnate delle dosi

ELSE

-- Se si provano a consegnare troppe dosi

```
IF (dosi_consegnabili < num_dosi_da_consegnare_p) THEN
RAISE dosi_eccessive;
end if;
end if;

-- Controlla se esiste già una consegna di quel lotto specifico
SELECT count(*)
INTO consegna_exists
FROM CONSEGNATO_A
    WHERE nome_luogo_p = NOME_LUOGO AND distretto_p =
DISTRETTO_LUOGO AND farmaco_p = NOME_FARMACO AND lotto_p =
NUM_LOTTO AND data_consegna_p = DATA_CONSEGNA;

-- Se esiste la consegna va incrementato il numero di dosi della consegna già
esistente
IF (consegna_exists = 1) THEN
SELECT NUM_DOSI_CONSEGNATE
INTO num_dosi_esistenti_specifiche
    FROM CONSEGNATO_A
    WHERE nome_luogo_p = NOME_LUOGO AND distretto_p =
DISTRETTO_LUOGO AND farmaco_p = NOME_FARMACO AND lotto_p =
NUM_LOTTO AND data_consegna_p = DATA_CONSEGNA;

UPDATE CONSEGNATO_A
SET NUM_DOSI_CONSEGNATE = num_dosi_esistenti_specifiche +
num_dosi_da_consegnare_p
    WHERE nome_luogo_p = NOME_LUOGO AND distretto_p =
DISTRETTO_LUOGO AND farmaco_p = NOME_FARMACO AND lotto_p =
NUM_LOTTO AND data_consegna_p = DATA_CONSEGNA;

ELSE
-- Consegna effettiva nel caso le dosi esistono ma non esiste la stessa
consegna
INSERT INTO CONSEGNATO_A VALUES (
    nome_luogo_p,
    distretto_p,
    farmaco_p,
```

```
        lotto_p,  
        data_consegna_real,  
        num_dosi_da_consegnare_p  
    );  
  
end if;  
  
COMMIT;  
  
-- Gestione delle eccezioni  
EXCEPTION  
WHEN negative_doses THEN  
    RAISE_APPLICATION_ERROR(-20014, 'Devi consegnare almeno una dose!');  
    rollback;  
  
    WHEN vaccino_non_esiste THEN  
        RAISE_APPLICATION_ERROR(-20010, 'Il vaccino che hai selezionato non esiste  
nel database.');
```

```
        rollback;  
  
    WHEN luogo_non_esiste THEN  
        RAISE_APPLICATION_ERROR(-20011, 'Il luogo che hai selezionato non esiste  
nel database.');
```

```
        rollback;  
  
    WHEN lotto_consegnato THEN  
        RAISE_APPLICATION_ERROR(-20012, 'Il lotto è stato già consegnato  
completamente.');
```

```
        rollback;  
  
    WHEN dosi_eccessive THEN  
        RAISE_APPLICATION_ERROR(-20013, 'Il numero di dosi che stai provando a  
consegnare è troppo alto.');
```

```
        rollback;  
end;
```

10.5.5 createVaccinazione

L'ultima procedura (e probabilmente la più importante) riguarda l'inserimento della vaccinazione nella base dati.

La procedura prende in input:

- Codice fiscale del vaccinando;
- Data effettiva della vaccinazione;
- Braccio di inoculazione;
- Nome e distretto del luogo in cui si sostiene la vaccinazione;
- Numero di lotto e nome del vaccino utilizzato.

Si controlla se il vaccinando ed il luogo esistono. Nel caso non viene passato un parametro per la data, viene utilizzata la data di inserimento della t-upla.

Si controlla se il vaccinando è idoneo, ed in caso affermativo si seleziona l'ultima idoneità che verrà utilizzata per generare l'associazione con Idoneo.

Se la vaccinazione per l'ultima anamnesi sostenuta è già stata inserita, si alza un'eccezione.

Nel caso sia la prima volta che il vaccinando si sottopone alla vaccinazione, si genera un numero di card vaccinato e si generano anche la data di attivazione e di scadenza. La card sarà valida a partire da 2 settimane dopo la prima dose, e la validità dura un anno da quella data.

Si controlla il tipo della prenotazione, per capire se si deve generare una nuova prenotazione per la seconda dose. Nel caso affermativo, si genera una nuova prenotazione incrementando il Num_tentativi; in caso negativo il ciclo di vaccinazione termina. Il ciclo di vaccinazione termina altresì se si utilizza un vaccino one-shot, dato che una singola somministrazione vale doppio.

```
CREATE OR REPLACE PROCEDURE createVaccinazione(  
cod_fisc_p          VACCINANDO.CODICE_FISCALE%TYPE,  
  
data_vaccinazione_p  VACCINAZIONE.DATA_ORA_EFFETTIVA%TYPE,  
braccio_p           VACCINAZIONE.BRACCIO_INOCULAZIONE%TYPE,  
  
luogo_p             VACCINAZIONE.NOME_LUOGO%TYPE,  
distretto_p         VACCINAZIONE.DISTRETTO_LUOGO%TYPE,
```

```
lotto_p          VACCINAZIONE.NUM_LOTTO%TYPE,
farmaco_p        VACCINAZIONE.NOME_FARMACO%TYPE
)
IS
    vaccinando_exists      NUMBER;
    num_prot_vaccinando    VACCINANDO.NUM_PROTOCOLLO%TYPE;
    vaccinando_non_esiste  EXCEPTION;
    luogo_exists          NUMBER;
    luogo_non_esiste      EXCEPTION;
    data_usata            DATE;
    max_carta            CARD_VACCINATO.NUM_CARTA%TYPE;
    card_exists          NUMBER;
    num_idoneita          NUMBER;
    data_ora_controllo_v  IDONEO.DATA_ORA_CONTROLLO%TYPE;
    num_tentativi_v      PRENOTAZIONE.NUM_TENTATIVI%TYPE;
    non_idoneo           EXCEPTION;
    tipo_p_v            PRENOTAZIONE.TIPO_PRENOTAZIONE%TYPE;
    check_doppia_dose    NUMBER;
    tempo_prima_seconda  LOTTO_VACCINO.TEMPO_PRIMA_SECONDA_DOSE%TYPE;
    vaccino_one_shot     LOTTO_VACCINO.DISCRIMINATORE%TYPE;
    vaccinazione_check   NUMBER;
    vaccinazione_gia_esiste EXCEPTION;

BEGIN
    -- Si controlla se esiste il vaccinando
    SELECT count(*)
    into vaccinando_exists
    FROM VACCINANDO
        WHERE CODICE_FISCALE = cod_fisc_p;

    -- Se non esiste, errore
    IF (vaccinando_exists = 0) THEN
        raise vaccinando_non_esiste;
    end if;

    -- Si controlla se esiste il luogo
```

```
SELECT count(*)
into luogo_exists
from LUOGO
    where DISTRETTO = distretto_p and NOME = luogo_p;

-- Se non esiste, errore
IF (luogo_exists = 0) THEN
raise luogo_non_esiste;
end if;

-- Se non viene passato un parametro per la data, si usa la data attuale.
IF (data_vaccinazione_p IS NULL) THEN
data_usata := sysdate;
ELSE
data_usata := data_vaccinazione_p;
end if;

-- Si ottiene il numero protocollo del vaccinando
SELECT NUM_PROTOCOLLO
INTO num_prot_vaccinando
FROM VACCINANDO
    WHERE cod_fisc_p = CODICE_FISCALE;

-- Si controlla se il vaccinando è risultato idoneo
SELECT count(*)
INTO num_idoneita
FROM IDONEO
    WHERE num_prot_vaccinando = NUM_PROTOCOLLO;

-- Se il vaccinando è idoneo, si seleziona l'ultima idoneità.
IF (num_idoneita <> 0) THEN
SELECT MAX(NUM_TENTATIVI), MAX(DATA_ORA_CONTROLLO)
INTO num_tentativi_v, data_ora_controllo_v
    FROM IDONEO
    WHERE NUM_PROTOCOLLO = num_prot_vaccinando;
ELSE
```



```
raise non_idoneo;  
end if;
```

-- Si controlla se è stata già inserita la vaccinazione che si sta provando ad inserire

```
SELECT count(*)  
into vaccinazione_check  
FROM VACCINAZIONE  
WHERE NUM_PROTOCOLLO = num_prot_vaccinando and NUM_TENTATIVI =  
num_tentativi_v;
```

```
if (vaccinazione_check <> 0) then  
raise vaccinazione_gia_esiste;  
end if;
```

-- Se c'è una sola idoneità, si deve generare il numero di carta

```
IF (num_idoneita > 0 and num_idoneita < 2) then
```

-- Si controlla se esistono tuple in card_vaccinato per generare il numero carta

```
SELECT count(*)  
into card_exists  
FROM CARD_VACCINATO;
```

```
IF (card_exists = 0) then
```

```
max_carta := 0;
```

```
ELSE
```

```
SELECT MAX(NUM_CARTA)  
INTO max_carta  
FROM CARD_VACCINATO;
```

```
end if;
```

-- Creazione della card vaccinato

```
INSERT INTO CARD_VACCINATO VALUES (  
    max_carta+1,  
    data_usata + 15,  
    data_usata + 15 + 365  
);
```

```
end if;
```

```
SELECT MAX(NUM_CARTA)
      INTO max_carta
      FROM CARD_VACCINATO;
```

```
-- Inserimento in vaccinazione
```

```
INSERT INTO VACCINAZIONE VALUES (
      data_usata,
      braccio_p,
      max_carta,
      luogo_p,
      distretto_p,
      lotto_p,
      farmaco_p,
      num_prot_vaccinando,
      num_tentativi_v,
      data_ora_controllo_v
);
```

```
-- Si controlla se la vaccinazione è a singola o doppia dose, e si crea la
prenotazione nuova nel secondo caso
```

```
select TIPO_PRENOTAZIONE
into tipo_p_v
from PRENOTAZIONE
      where NUM_PROTOCOLLO = num_prot_vaccinando AND
```

```
NUM_TENTATIVI = num_tentativi_v;
```

```
select DISCRIMINATORE
into vaccino_one_shot
FROM LOTTO_VACCINO
      WHERE NUM_LOTTO = lotto_p AND NOME_FARMACO = farmaco_p;
```

```
if (tipo_p_v = 'DOPPIA' AND vaccino_one_shot = 'T') then
```

```
-- Si controlla che effettivamente si inserisca solo una seconda dose, e non una
terza, quarta...
```

```
-- si conta quante vaccinazioni sono state effettuate su quel vaccinando
```

```
SELECT count(*)
into check_doppia_dose
    from VACCINAZIONE
    where NUM_PROTOCOLLO = num_prot_vaccinando;

-- Se esiste effettivamente solo una vaccinazione
if (check_doppia_dose = 1) then
    -- Si controlla quale vaccino è stato usato e si controlla dopo quanto deve
    essere effettuata la seconda dose
    SELECT TEMPO_PRIMA_SECONDA_DOSE
    into tempo_prima_seconda
        FROM LOTTO_VACCINO
        WHERE NOME_FARMACO = farmaco_p and NUM_LOTTO = lotto_p;

    -- Si crea la nuova prenotazione
    INSERT INTO PRENOTAZIONE VALUES (
        num_prot_vaccinando,
        num_tentativi_v+1,
        data_usata + tempo_prima_seconda,
        tipo_p_v
    );

end if;
end if;
COMMIT;

EXCEPTION
WHEN vaccinando_non_esiste then
    raise_application_error(-20030, 'Il vaccinando non esiste!');
rollback;

when luogo_non_esiste then
    raise_application_error(-20031, 'Il luogo scelto non esiste!');
rollback;

when non_idoneo then
    raise_application_error(-20032, 'Il vaccinando non appare nella tabella
idoneo!');
```

```
rollback;  
  
when vaccinazione_gia_esiste then  
raise_application_error(-20077, 'La vaccinazione in inserimento esiste già.');
```

rollback;
end;