

Ditta Di Trasporti

Gennaro Panico
0124002115

Michele Fiorentino
0124002085



Anno accademico: 2020/2021

Data di Consegna: 22/06/2021 Ore: 00:00

UNIVERSITÀ DEGLI STUDI DI NAPOLI PARTHENOPE

DIPARTIMENTO DI SCIENZE E TECNOLOGIE

CORSO DI BASI DI DATI E LABORATORIO



Indice

1	Progettazione	3
1.1	Sintesi dei requisiti	3
1.2	Glossario	4
1.3	Diagramma EE/R	4
1.4	Diagramma Relazionale	8
1.5	Utenti e loro categorie	9
1.6	Utenti	10
1.7	Verifica di normalità	10
2	Implementazione	10
2.1	Creazione Utenti	11
2.2	Data Definition Language(DDL)	11
2.2.1	Descrizione tabelle:	12
2.3	Data Manipulation Language(DML)	16
2.4	Trigger	21
2.5	Procedure	30
2.5.1	Descrizione procedura Inserimento Autista	30
2.5.2	Descrizione procedura Inserimento Mezzo	31
2.5.3	Descrizione procedura Inserimento Fattura	32
2.5.4	Descrizione procedura Inserimento Viaggio	33
2.6	Chiamate a procedure	35
2.7	Data Control Language	39

1 Progettazione

1.1 Sintesi dei requisiti

Il Database deve gestire in maniera ottimale le spedizioni e la tracciabilità delle stesse per una ditta di trasporti. La ditta chiede, inoltre, di possedere nel Database le informazioni necessarie riguardanti gli autisti e i mezzi posseduti, per eventuali controlli su entrambi. Riguardo gli autisti, infatti, la ditta chiede di registrare dati anagrafici e patenti (CQC e ADR) per visualizzarne scadenze e competenze ed agire di conseguenza. La ditta precisa di dare in dotazione un cellulare all' autista sia per contatti da essa stessa che dalle sedi riceventi, pertanto si decide di registrare anche questo tipo di contatto oltre a quello privato dell'autista. Riguardo i mezzi la ditta chiede di registrarne tutti i documenti, quindi contratti d'acquisto, assicurazioni e revisione, ma anche la tipologia dei mezzi e dei rimorchi posseduti (In possesso della ditta: autocarro, rimorchio e tipo, container, trattore stradale). Vuole sapere, inoltre, l'esito di una revisione e l'annessa riparazione in caso di esito negativo dalla revisione o rientro da incidente. Si è deciso di registrare ciò dopo avvenuto confronto con la ditta, in quanto essa ha bisogno di conoscere costantemente i suoi mezzi a disposizione, per tanto c'è bisogno di conoscere tutti gli annessi problemi che incombono sui mezzi posseduti dall'azienda e sui suoi dipendenti. Poiché lo scopo del Database è quello di gestire spedizioni e tracciabilità, si è scelto di centrarlo sul viaggio effettuato da ogni autista. L'azienda che sceglie il servizio di spedizione offerto dalla ditta, deve poter prenotare il viaggio o eventualmente disdire, ricevendo tempi di consegna e fattura dell'avvenuto pagamento, nel momento in cui conclude la procedura di prenotazione viaggio. La ditta ha stabilito inoltre, di voler offrire il servizio unicamente dopo aver ricevuto il pagamento dall' azienda richiedente. La spedizione può partire sia dalla ditta offerente che da una sede indicata dal richiedente. Il Database deve registrare pertanto i dati delle sedi (P.IVA, indirizzi e contatti) e i dati delle fatture emanate. La ditta decide, inoltre, di registrare il DDT nel momento in cui viene rilasciato, il numero di ore, i km effettuati e il tipo di viaggio (se nazionale o internazionale) in maniera antecedente al viaggio, e in maniera approssimativa, eventuali variazioni verranno segnalate mediante dispositivo installato sul trattore stradale.

1.2 Glossario

Tabella 1: Glossario

TERMINE	SIGNIFICATO
CQC	Carta di qualificazione per trasporto di persone e di cose
ADR	Carta di qualificazione per trasporto di merci pericolose: carburanti, ecc..
DDT	Documento di trasporto, certifica un trasferimento di merci
AUTOCARRO	mezzo singolo per il trasporto merci in maniera autonoma
RIMORCHIO	E' un veicolo sprovvisto di motore, destinato ad essere trainato da autoveicoli
CONTAINER	E' una attrezzatura specifica usata per i trasporti intermodale
TRATTORE	E' un veicolo adibito esclusivamente al traino su strada di semirimorchi
REVISIONE	controllo periodico della funzionalità di un veicolo, in rispetto delle norme stradali
P.IVA	Identifica un soggetto che esercita un'attività, di impresa e non

1.3 Diagramma EE/R

In Figura 5 viene mostrato il diagramma EE/R, si noti che molti vincoli, cardinalità massime e minime delle associazioni, non sono esprimibili mediante questa rappresentazione. Il DBMS e' centrato interamente sull'entità vaiggio.

La parte in alto a sinistra dell' EE/R(Figure 1) rappresenta gli autisti, i contratti legati agli autisti, e le patenti. Se un'autista viene licenziato prima della scadenza del contratto, anticipiamo la data di fine nel DB. (e' una semplificazione, invece di tener conto di lettere di licenziamo ecc.). Se e' a tempo indeterminato, la data fine e' sconosciuta, quindi e' NULL.

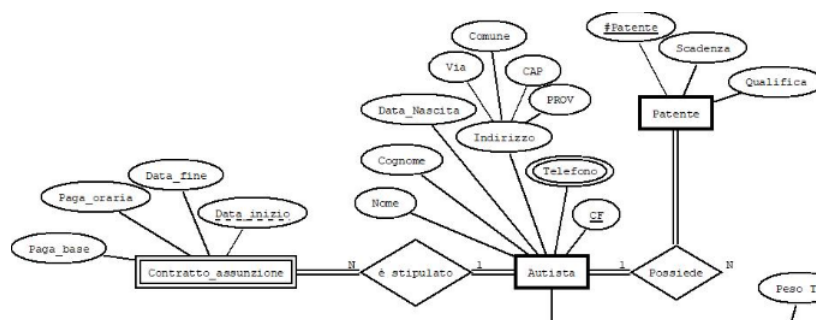


Figura 1: Diagramma EE/R

Nella parte in alto a destra dell'EE/R(Figure 2) vengono riportate le entità inerenti all'azienda esterna che decide di prenotare un viaggio presso la suddetta ditta, si ritiene quindi opportuno progettare le entità fattura e sede permettendo in questo modo al DBMS di salvare tutti i dati necessari (sia quelli necessari alla spedizione che quelli necessari legalmente, ai pagamenti). Si noti, inoltre,

che il numero di fattura e' un numero progressivo, definito come chiave primaria dell'entità fattura.

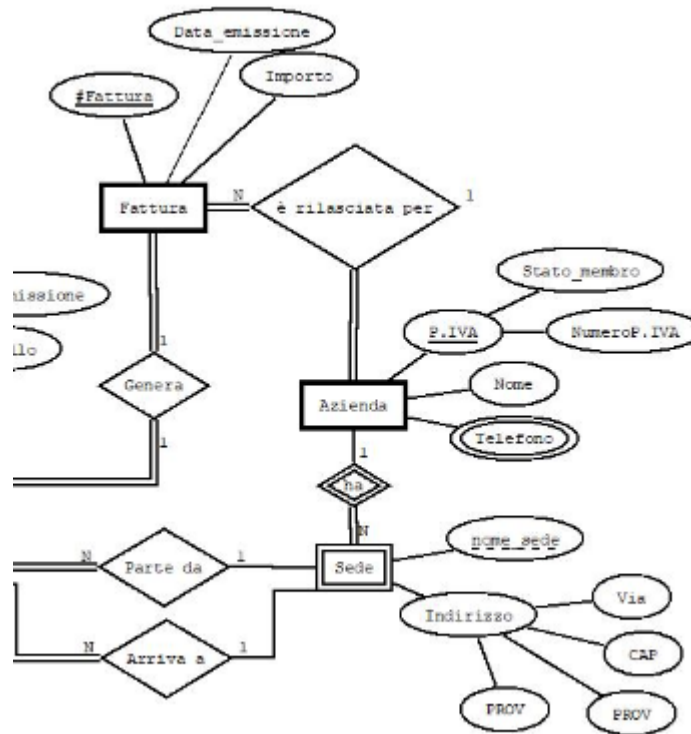


Figura 2: Diagramma EE/R

Nella parte inferiore sinistra invece(Figura 3), viene progettata l'entità mezzo che si specializza nei vari tipi e i contratti annessi ai mezzi(acquisto e assicurazione).Si noti che progettualmente nella specializzazione di mezzo non viene assunto il tipo "autotreno" in maniera diretta, ma a rigor di logica tale tipo, per come e' stato progettato l'EE/R, esiste nel momento in cui viene collegato un rimorchio ad un autocarro. L'entità CONTRATTO ASSUNZIONE, non può essere un'entità forte perche' nel caso eliminassimo l'autista non saremmo più in grado di sapere a quale autista faceva riferimento il contratto. A questo punto ci conviene tenere traccia di tutti gli autisti che abbiamo avuto nel corso del tempo, e per capire se un'autista lavora ancora con noi andremo a verificare se la data fine e' minore della data attuale (tramite una query).

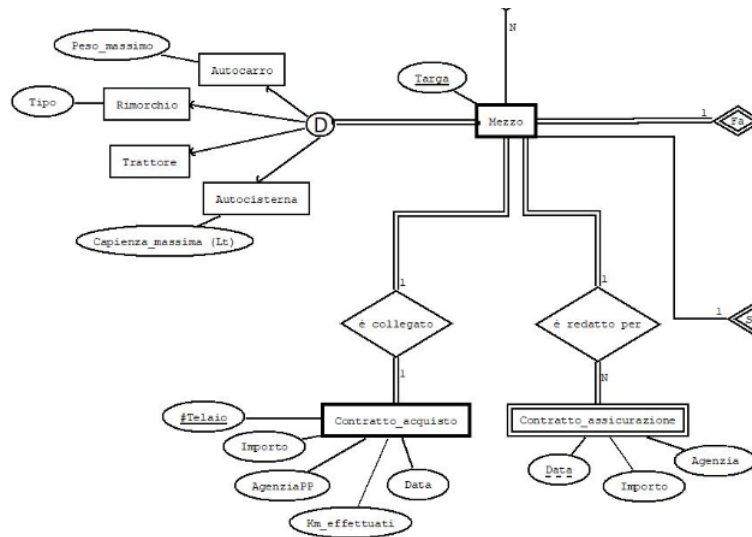


Figura 3: Diagramma EE/R

La parte inferiore a destra (Figure 4) riguarda i guasti, gli incidenti e quindi le annesse riparazioni e revisioni inerenti ai mezzi. Si noti che Incidente, Revisione e Riparazioni sono entità deboli, con chiave debole *Data*, e che l'entità Revisione, si specializza in esito positivo e esito negativo, in questo modo viene assunto che se la revisione risulti essere con esito negativo (non passata), allora deve notificare con un nome indicante il guasto (in maniera generica) ed eventualmente deve essere risolto mediante Riparazione.

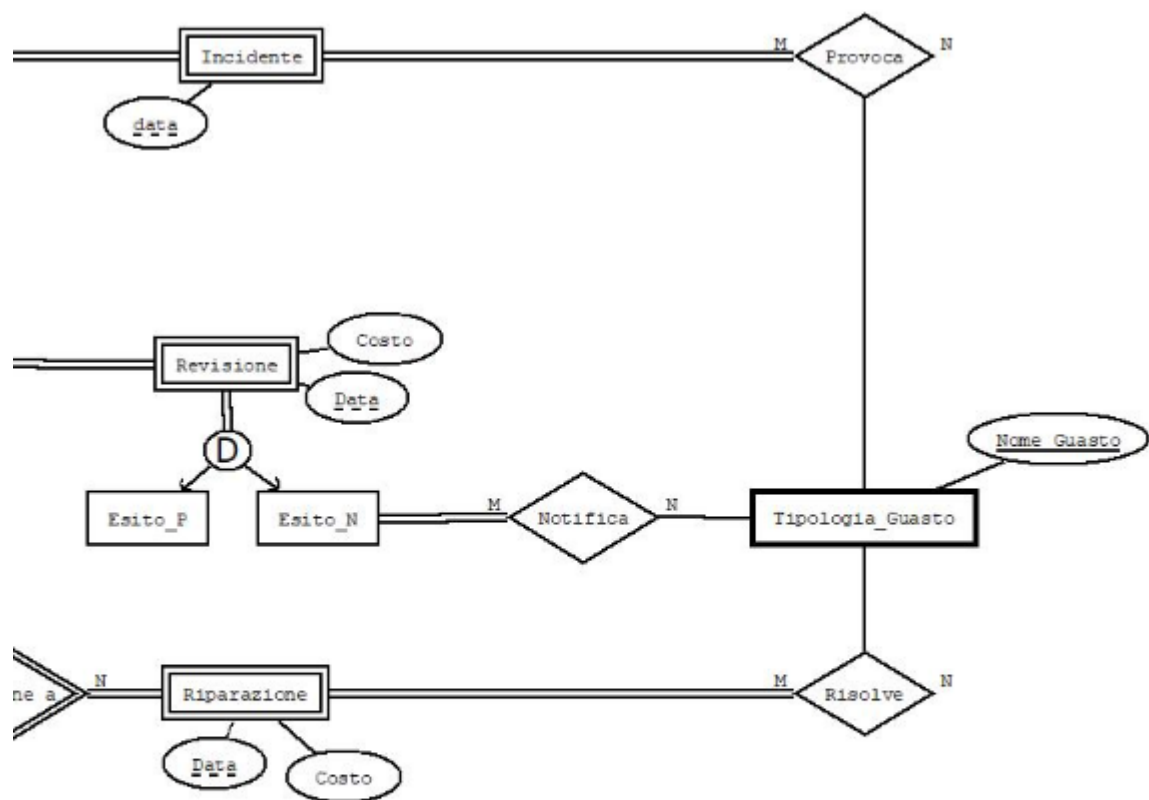


Figura 4: Diagramma EE/R

Di seguito (Figura 5), viene riportato il Diagramma EE/R completo, di tutte le parti precedentemente spiegate, si noti quanto esso è centrato sul viaggio, che è il servizio offerto dalla ditta in questione.

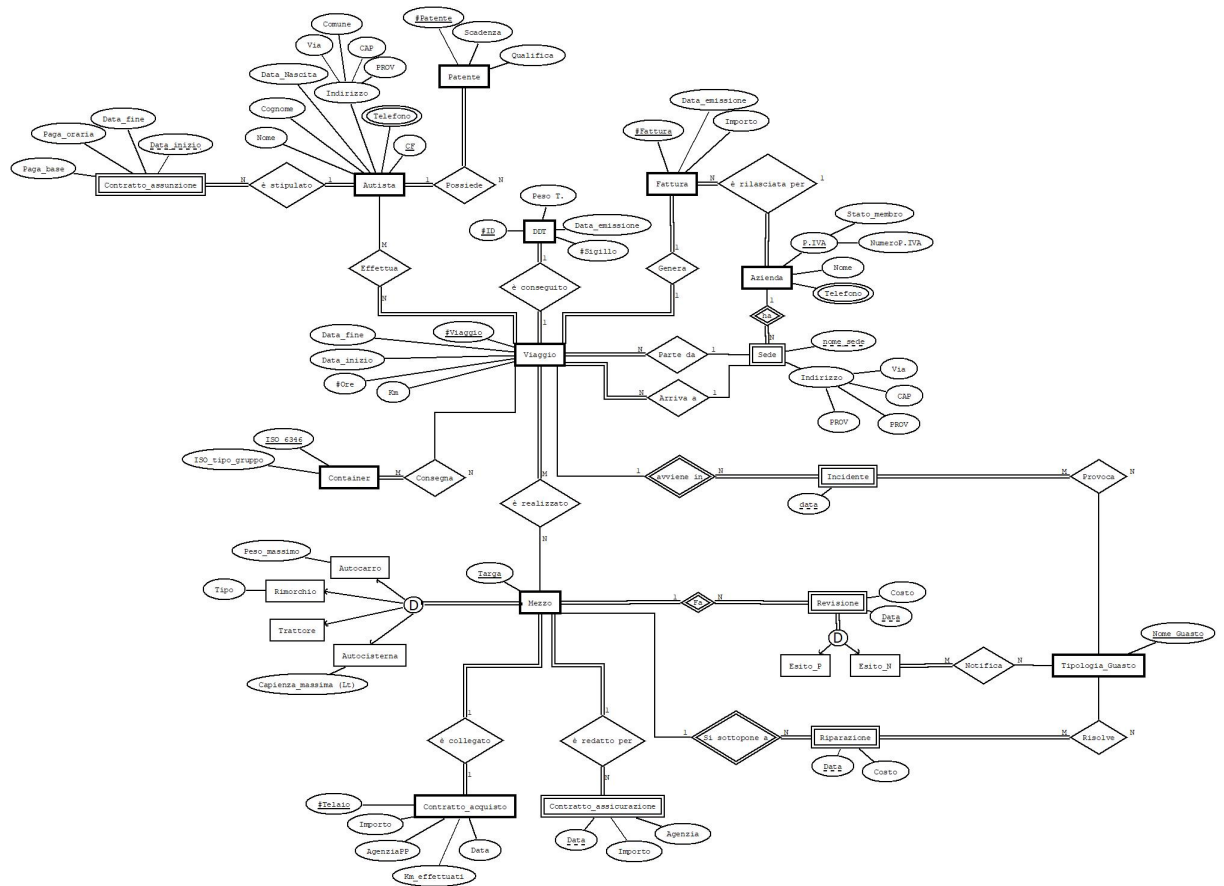


Figura 5: Diagramma EE/R

1.4 Diagramma Relazionale

In Figura 6 viene mostrato il diagramma relazionale, si noti l'utilizzo del tipo DATE¹ e il tipo VARCHAR2². Tale diagramma e' l'esatta traduzione del precedente diagramma EE/R ed inoltre, rispetto a quest'ultimo possiede un grado di astrazione maggiore, ma si avvicina al codice, in quanto ci permette di definire (nella modalità adottata) anche i tipo dei vari campi.

¹In Oracle, il tipo DATE, include data e ora e non solo data come avviene per altri DBMS.

²In Oracle, VARCHAR2 e' molto più efficiente in termini di spazio, ed e' standard di Oracle, ciò viene meglio argomentato nella parte iniziale di creazione tabelle.

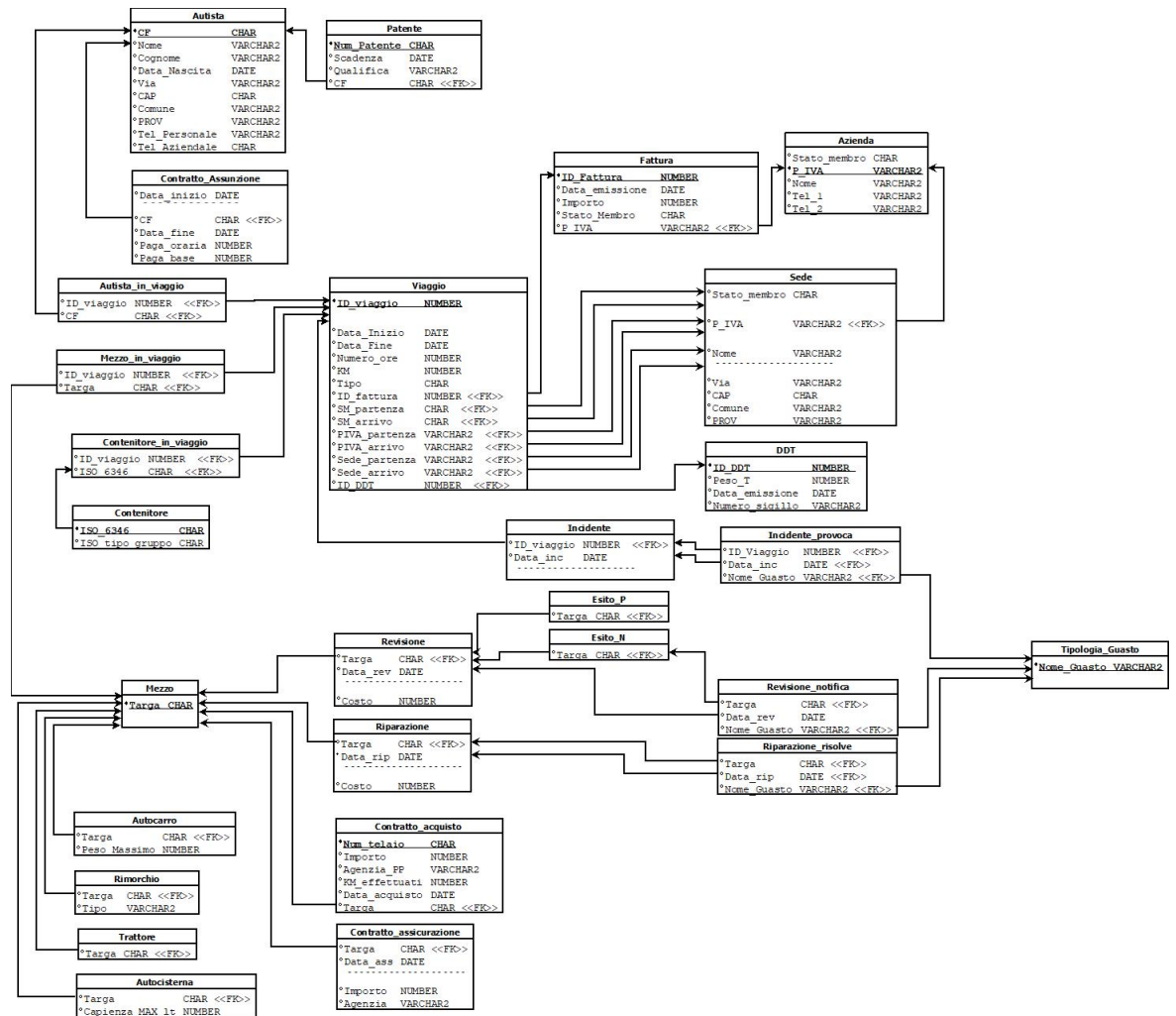


Figura 6: Diagramma Relazionale

1.5 Utenti e loro categorie

Per questo DB occorre definire quattro utenti. Il primo è l'utente `db.Ditta_trasporti` che possiede tutti i permessi necessari per modificare il DBMS, ovvero è *admin*. `u_gest_autisti`, è l'utente che si occupa dell'inserimento e della gestione degli autisti all'interno del DBMS. `u_gestmezzi`, analogamente si occupa dell'inserimento e della gestione dei mezzi. Infine, `u_gestviaggi`, è l'utente che si occupa dell'inserimento e la gestione dei viaggi e delle fatture allegate ai viaggi.

1.6 Utenti

Tabella 2: Tavola degli Utenti

UTENTE	TIPO	VOLUME	PERMESSI
db_Ditta_trasporti	Amministratore	1	ALL
u_gest_autisti	Comune	1	ALL ON INS_AUTISTI
u_gest_mezzi	Comune	1	ALL ON INS_MEZZI
u_gest_viaggi	Comune	1	ALL ON INS_FATTURA ALL ON INS_VIAGGIO

1.7 Verifica di normalità

Nella verifica di normalità si combatte la ridondanza attraverso la caccia alle dipendenze funzionali anomale. Mentre il rispetto della prima forma normale e' in qualche modo dato per scontato, implicito nel modello relazionale, la seconda e la terza vanno verificate analizzando il significato degli attributi, poiche' si tratta di una proprietà intensionale, non estensionale. Notare come la maggior problematica avuta con la prima forma normale sono stati gli attributo "tipo", i quali tuttavia sono stati implementati in modo tali da essere atomici e rispettano un dominio ben preciso. In particolare le tipologie di guasto sono stati gli attributi più insidiosi in quanto si trattavano di attributi "tipo" multivalore. Per ovviare tale problema abbiamo quindi creato una tabella "tipologia_guasto" che fungesse da dominio (Tale tabella e' stata descritta in dettaglio nella parte della relazione a lei dedicata). La relazione rispetta anche la seconda forma normale in quanto, sebbene siano presenti molte chiavi attributo, in nessun caso si verifica una dipendenza parziale da chiavi. Infine, la relazione rispetta anche la terza forma normale in quanto non sono presenti dipendenze anomale SE NON da attributi primi (es. $CF \rightarrow \text{nome}$). Infatti ricordo che nella BCNF non ci sono dipendenze anomale, mentre nella BNF sono ammesse dipendenze anomale ma solo da attributi primi. Ergo, lo schema e' in BNF.

2 Implementazione

Questo capitolo e' dedicato all'implementazione del DBMS, tutto ciò che viene progettato nel capitolo precedente viene tradotto in codice eseguibile sul DBMS Oracle 10 Express Edition.

Dopo la fase progettuale del DBMS dedichiamo questo capitolo all'implementazione di quest' ultimo. L'implementazione consiste difatti nella traduzione di tutte le specifiche precedentemente progettate in codice eseguibile. Il codice si riferisce al DBMS di Oracle 10 XE, e il linguaggio usato e' PL/SQL(estensione proprietaria di Oracle). Il capitolo segue un ordine logico con cui presentare

l'argomento, ma naturalmente l'ordine con cui lanciare gli script potrebbe anche non essere quello presentato dal capitolo. Si noti che tutte le parole riservate sono scritte in maiuscolo ed evidenziate in blu, mentre tutti i termini arbitrari sono scritti in nero e in formato minuscolo.

2.1 Creazione Utenti

Innanzitutto bisogna accedere al DBMS come amministratore di sistema, e successivamente creare l'utente proprietario della base di dati. Si noti che Oracle possiede l'architettura multi-tenant pertanto vanno definiti gli accessi a tutti gli utenti che intendono usufruire della base di dati. Seguono le righe di codice ove vengono creati gli utenti, ai quali però, non vengono assegnati privilegi di oggetto perche' lo schema ancora non esiste.

```
CREATE USER db_Ditta_trasporti IDENTIFIED BY admin_DT;  
CREATE USER u_gest_viaggi IDENTIFIED BY dittatrasportiL;  
CREATE USER u_gest_mezzi IDENTIFIED BY dittatrasportiM;  
CREATE USER u_gest_autisti IDENTIFIED BY dittatrasportiA;  
GRANT ALL PRIVILEGES TO db_Ditta_trasporti;
```

Successivamente, bisogna disconnettersi e riconnettersi con le credenziali di *DBAdmin*³, poiche' tale utente, essendo amministratore della nuova base di dati, possiede i permessi necessari alla creazione creazione degli oggetti che occorrono.

2.2 Data Definition Language(DDL)

Il Data Definition Language(DDL) contiene tutte le tabelle espresse nel *Diagramma Relazionale*, tradotte in linee di codice. Il comando predominante e' il CREATE TABLE, che si ripete ogni volta che creo una tabella. Vengono anche espressi tutti i *vincoli d'integrità esprimibili*⁴ nel modello relazionale ovvero: l'obbligatorietà, vincolo di dominio statico, l'univocità, l'integrità referenziale etc...

Si noti, inoltre, che in Oracle il tipo DATE, include data e ora, e non solo la data come in altri DBMS, per tanto tali campi che seguono nel codice hanno questo formato. Viene utilizzato il tipo VARCHAR2 anziche' VARCHAR in quanto risulta essere un tipo standard di Oracle, quindi nel tempo, non cambia la sua definizione, inoltre, può salvare un massimo di 4000 byte rispetto ai 2000 del VARCHAR.

³DBAdmin ricevendo tutti i privilegi attraverso il comando GRANT ALL PRIVILEGES, e' amministratore dell'intera base di dati. Nel caso il DBMS ospiti più DB, e' opportuno limitare i privilegi di ciascun amministratore unicamente alla base di dati di cui e' proprietario, e usare l'account SYSTEM come amministratore di sistema.

⁴Nel DDL, e' possibile esprimere solo un sottoinsieme dei vincoli, infatti, quelli dinamici e estensionali, non sono esprimerli.

2.2.1 Descrizione tabelle:

Si noti, che la maggior parte delle tabelle implementate risultano essere abbastanza semplici, ma ci sono alcune tabelle su cui vorremmo porre particolare attenzione:

Tabella autisti:

In tale tabella vengono definite le informazioni necessarie alla ditta riguardanti gli autisti. Interessano, infatti, i dati anagrafici di ogni autista. Il codice fiscale viene definito chiave primaria attraverso il comando PRIMARY KEY, che adotteremo in tutte le entità che non sono deboli, e che vengono tradotte in tabelle. Viene inserito anche un campo "Tel_aziendale" per tener conto anche del numero di telefono che la ditta assegna all'autista nel momento in cui viene assunto.

Tabella Patente:

Viene ritenuto opportuno creare una tabella Patente, con l'inserimento di tutti i dati necessari alla ditta per conoscere le qualifiche (Se in possesso di CQC, ADR o entrambi) e le scadenze di tutte le patenti dei propri autisti. Si noti che in questa tabella viene, viene inoltre, inserita una chiave esterna attraverso CONSTRAINT, denotata come FK_Patente che rappresenta il CF con referenza alla tabella autista. Adotteremo il comando CONSTRAINT con annessa referenza, ogni qual volta vi sarà necessità dell'inserimento di chiavi esterne nella tabella.

Tabella Tipologia Guasto:

Viene ritenuto opportuno ideare una tabella indicante la tipologia del guasto per i mezzi. La tabella possiede il campo Nome_guasto che a un primo sguardo sembrerebbe un campo di tipo descrizione, ma assumiamo questo campo come un dominio, utile semplicemente a conoscere il nome indicante il tipo di guasto avvenuto. Tale soluzione è sembrata essere quella più adatta allo scopo in quanto molto meno complessa, ma comunque funzionale. Si noti inoltre che Nome_guasto viene definito di tipo VARCHAR2 con un parametro massimo di 50.

Tabella Contratto Acquisto:

La ditta ritiene opportuno salvare i dati riguardanti i mezzi, quindi contratti d'acquisto compresi. Tra i campi salviamo i dati relativi all'importo per eventuali prospettive che la ditta può calcolarsi negli anni, l'agenzia presso cui è stato effettuato il passaggio di proprietà del veicolo, i chilometri effettuati dal mezzo al momento dell'acquisto e infine viene segnato anche il numero del telaio del veicolo.

TabelleDT

```

1      -- Sezione Autista
2
3  CREATE TABLE Autista(
4  CF          CHAR(16) PRIMARY KEY,
5  Nome        VARCHAR2(20),
6  Cognome     VARCHAR2(20),
7  Data_Nascita DATE,
8  Via         VARCHAR2(40),
9  CAP         CHAR(5),
10 Comune     VARCHAR2(30),
11 PROV        VARCHAR(5), -- consideriamo anche le "province" estere
12 Tel_Personale VARCHAR2(11) UNIQUE, --Numero di casa(11) o Numero Cellulare (10)
13 Tel_Aziendale CHAR(10) UNIQUE
14 );
15
16 CREATE TABLE Patente(
17 Num_Patente CHAR(10) PRIMARY KEY,
18 Scadenza    DATE,
19 Qualifica   VARCHAR2(3),
20 CF          CHAR(16),
21 CONSTRAINT FK_Patente FOREIGN KEY (CF) REFERENCES Autista(CF)
22 );
23
24 CREATE TABLE Contratto_assunzione( --TENIAMO IN CONTO TUTTI I CONTRATTI DI ASSUNZIONE, PURE PRECEDENTI
25 Data_inizio DATE,
26 CF          CHAR(16),
27 Data_fine   DATE,
28 Paga_oraria NUMBER(4,2),
29 Paga_base   NUMBER(7,2), --cambiato number da 4 a 5
30
31 CONSTRAINT FK_CA FOREIGN KEY (CF) REFERENCES Autista (CF),
32 CONSTRAINT PK_CA PRIMARY KEY (Data_inizio,CF)
33 );
34
35
36
37 -- Sezione Azienda
38
39 CREATE TABLE Azienda(
40 Stato_membro CHAR(2), --Inseriamo anche la sigla della nazione
41 P_IVA         VARCHAR2(12), --LA P.IVA ha numero variabile. Dipende dallo Stato.
42 Nome         VARCHAR2(50),
43 Tel_1        VARCHAR2(16) UNIQUE, --Incluso prefisso
44 Tel_2        VARCHAR2(16) UNIQUE, --Incluso prefisso
45
46 CONSTRAINT PK_AZ PRIMARY KEY (Stato_membro,P_IVA)
47 );
48
49 CREATE TABLE Sede(
50 Stato_membro CHAR(2), --Inseriamo anche la sigla della nazione
51 P_IVA         VARCHAR2(12), --LA P.IVA ha numero variabile. Dipende dallo Stato.
52 Nome         VARCHAR2(50),
53 Via         VARCHAR2(40),
54 CAP         CHAR(5),
55 Comune     VARCHAR2(30),
56 PROV        VARCHAR(5), -- consideriamo anche le "province" estere
57
58 CONSTRAINT FK_SEDE FOREIGN KEY (Stato_membro,P_IVA) REFERENCES Azienda (Stato_membro,P_IVA),
59 CONSTRAINT PK_SEDE PRIMARY KEY (Stato_membro,P_IVA, Nome)
60 );
61
62 CREATE TABLE Fattura(
63 ID_fattura    NUMBER(10) PRIMARY KEY,
64 Data_emissione DATE,
65 Importo      NUMBER(8,2),
66 Stato_membro CHAR(2), --Inseriamo anche la sigla della nazione
67 P_IVA         VARCHAR2(12), --LA P.IVA ha numero variabile. Dipende dallo Stato.
68

```

```

69 CONSTRAINT FK_FAT FOREIGN KEY (Stato_membro,P_IVA) REFERENCES Azienda (Stato_membro,P_IVA)
70 );
71
72 CREATE TABLE DDT(
73 ID_DDT          NUMBER(10) PRIMARY KEY,
74 Peso_T          NUMBER(4,2),
75 Data_emissione DATE,
76 Numero_Sigillo VARCHAR2(16)
77 );
78
79
80 -- Sezione MEZZO
81
82 CREATE TABLE Mezzo(
83 Targa          CHAR(7) PRIMARY KEY
84 );
85
86 CREATE TABLE Autocarro(
87 Targa          CHAR(7),
88 Peso_Massimo_T NUMBER(24,2),
89
90 CONSTRAINT FK_AUTOCA FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
91 CONSTRAINT PK_AUTOCA PRIMARY KEY (Targa)
92 );
93
94 CREATE TABLE Rimorchio(
95 Targa          CHAR(7),
96 Tipo           VARCHAR2(30),
97
98 CONSTRAINT FK_RIM FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
99 CONSTRAINT PK_RIM PRIMARY KEY (Targa)
100 );
101
102 CREATE TABLE Trattore(
103 Targa          CHAR(7),
104
105 CONSTRAINT FK_TRAT FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
106 CONSTRAINT PK_TRAT PRIMARY KEY (Targa)
107 );
108
109 CREATE TABLE Autocisterna(
110 Targa          CHAR(7),
111 Capienza_MAX_lt NUMBER(5,0),
112
113 CONSTRAINT FK_AUTOCI FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
114 CONSTRAINT PK_AUTOCI PRIMARY KEY (Targa)
115 );
116
117
118 CREATE TABLE Contratto_acquisto(
119 Num_telaio     CHAR(17) PRIMARY KEY,
120 Importo        NUMBER(9,2),
121 Agenzia_PP     VARCHAR2(50),
122 KM_effettuati  NUMBER(7,0),
123 Data_acquisto  DATE,
124 Targa          CHAR(7) NOT NULL,
125
126 CONSTRAINT FK_CACQUISTO FOREIGN KEY (Targa) REFERENCES Mezzo (Targa)
127 );
128
129 CREATE TABLE Contratto_assicurazione(
130 Targa          CHAR(7),
131 Data_ass       DATE,
132 Importo        NUMBER(7,2),
133 Agenzia        VARCHAR2(50),
134
135 CONSTRAINT FK_CASSIC FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
136 CONSTRAINT PK_CASSIC PRIMARY KEY (Targa,Data_ass)
137 );
138
139
140 -- sezione VIAGGIO
141
142 CREATE TABLE Viaggio(
143 ID_viaggio     NUMBER(6) PRIMARY KEY,
144 Data_inizio    DATE,
145 Data_fine      DATE,
146 Numero_ore     NUMBER(3),
147 KM             NUMBER(5),
148
149 ID_fattura     NUMBER(10) UNIQUE NOT NULL,
150 SM_partenza    CHAR(2),
151 PIVA_partenza  VARCHAR2(12),
152 Sede_partenza  VARCHAR2(50),
153 SM_arrivo      CHAR(2),
154 PIVA_arrivo    VARCHAR2(12),
155 Sede_arrivo    VARCHAR2(50),
156 ID_DDT         NUMBER(10) UNIQUE NOT NULL,
157
158 CONSTRAINT FK_V
159 FOREIGN KEY (ID_fattura) REFERENCES Fattura (ID_fattura),

```

```

160 FOREIGN KEY (SM_partenza, PIVA_partenza, Sede_partenza) REFERENCES Sede (Stato_membro, P_IVA, Nome),
161 FOREIGN KEY (SM_arrivo, PIVA_arrivo, Sede_arrivo) REFERENCES Sede (Stato_membro, P_IVA, Nome),
162 FOREIGN KEY (ID_DDT) REFERENCES DDT (ID_DDT)
163 );
164
165 CREATE TABLE Contenitore(          --Container non e' disponibile come nome
166 ISO_6346 CHAR(11) PRIMARY KEY,      --11 CIFRE
167 ISO_tipo_gruppo CHAR(4)
168 );
169
170 -- Tabelle transizione Viaggio
171
172 CREATE TABLE Contenitore_in_viaggio(
173 ID_viaggio NUMBER(6),
174 ISO_6346 CHAR(11),      --11 CIFRE
175
176 CONSTRAINT FK_CIV
177 FOREIGN KEY (ID_viaggio) REFERENCES Viaggio (ID_viaggio),
178 FOREIGN KEY (ISO_6346) REFERENCES Contenitore (ISO_6346),
179
180 CONSTRAINT PK_CIV PRIMARY KEY (ID_viaggio, ISO_6346)
181 );
182
183 CREATE TABLE Autista_in_viaggio(
184 ID_viaggio NUMBER(6),
185 CF CHAR(16),
186
187 CONSTRAINT FK_AIV
188 FOREIGN KEY (ID_viaggio) REFERENCES Viaggio (ID_viaggio ),
189 FOREIGN KEY (CF) REFERENCES Autista (CF),
190
191 CONSTRAINT PK_AIV PRIMARY KEY (ID_viaggio, CF)
192 );
193
194 CREATE TABLE Mezzo_in_viaggio(
195 ID_viaggio NUMBER(6),
196 Targa CHAR(7),
197
198 CONSTRAINT FK_MIV
199 FOREIGN KEY (ID_viaggio) REFERENCES Viaggio (ID_viaggio ),
200 FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
201
202 CONSTRAINT PK_MIV PRIMARY KEY (ID_viaggio, Targa)
203 );
204
205 -- sezione GUASTI
206
207 CREATE TABLE Incidente(
208 ID_viaggio NUMBER(6),
209 Data_inc DATE,
210
211 CONSTRAINT FK_INC FOREIGN KEY (ID_viaggio) REFERENCES Viaggio (ID_viaggio),
212 CONSTRAINT PK_INC PRIMARY KEY (ID_viaggio, Data_inc)
213 );
214
215
216 CREATE TABLE Revisione(
217 Targa CHAR(7),
218 Data_rev DATE,
219 Costo NUMBER(5,2),
220
221 CONSTRAINT FK_REV FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
222 CONSTRAINT PK_REV PRIMARY KEY (Targa, Data_rev)
223 );
224
225 CREATE TABLE Esito_P(          --esito Revisione
226 Targa CHAR(7),
227 Data_rev DATE,
228
229 CONSTRAINT FK_EP FOREIGN KEY (Targa, Data_rev) REFERENCES Revisione (Targa, Data_rev),
230 CONSTRAINT PK_EP PRIMARY KEY (Targa, Data_rev)
231 );
232
233 CREATE TABLE Esito_N(          --esito Revisione. e' collegato al tipo di guasto
234 Targa CHAR(7),
235 Data_rev DATE,
236
237 CONSTRAINT FK_EN FOREIGN KEY (Targa, Data_rev) REFERENCES Revisione (Targa, Data_rev),
238 CONSTRAINT PK_EN PRIMARY KEY (Targa, Data_rev)
239 );
240
241
242 CREATE TABLE Riparazione(
243 Targa CHAR(7),
244 Data_rip DATE,
245 Costo NUMBER(8,2),
246
247 CONSTRAINT FK_RIP FOREIGN KEY (Targa) REFERENCES Mezzo (Targa),
248 CONSTRAINT PK_RIP PRIMARY KEY (Targa, Data_rip)
249 );
250

```

```

251 CREATE TABLE Tipologia_guasto(
252 Nome_guasto VARCHAR2(50) PRIMARY KEY
253 );
254
255 -- tabelle transizione GUASTO
256
257 CREATE TABLE Incidente_provoca(
258 ID_viaggio NUMBER(6),
259 Data_inc DATE,
260 Nome_Guasto VARCHAR2(50),
261
262 CONSTRAINT FK_IP
263 FOREIGN KEY (ID_Viaggio,Data_inc) REFERENCES Incidente (ID_viaggio,Data_inc),
264 FOREIGN KEY (Nome_guasto) REFERENCES Tipologia_guasto (Nome_guasto),
265
266 CONSTRAINT PK_IP PRIMARY KEY (ID_viaggio, Data_inc, Nome_guasto)
267 );
268
269 CREATE TABLE Revisione_notifica(
270 Targa CHAR(7),
271 Data_rev DATE,
272 Nome_Guasto VARCHAR2(50),
273
274 CONSTRAINT FK_RN
275 FOREIGN KEY (Targa,Data_rev) REFERENCES Esito_N (Targa,Data_rev),
276 FOREIGN KEY (Nome_guasto) REFERENCES Tipologia_guasto (Nome_guasto),
277
278 CONSTRAINT PK_RN PRIMARY KEY (Targa,Data_rev, Nome_guasto)
279 );
280
281 CREATE TABLE Riparazione_risolve(
282 Targa CHAR(7),
283 Data_rip DATE,
284 Nome_Guasto VARCHAR2(50),
285
286 CONSTRAINT FK_RR
287 FOREIGN KEY (Targa,Data_rip) REFERENCES Riparazione (Targa,Data_rip),
288 FOREIGN KEY (Nome_guasto) REFERENCES Tipologia_guasto (Nome_guasto),
289
290 CONSTRAINT PK_RR PRIMARY KEY (Targa,Data_rip, Nome_guasto)
291 );
292

```

Sequenze

```

1
2 CREATE SEQUENCE seq_IDF -- autoincremento ID Fatture
3 MINVALUE 1
4 START WITH 1
5 INCREMENT BY 1
6 CACHE 10;
7
8 CREATE SEQUENCE seq_DDT -- autoincremento ID DDT
9 MINVALUE 1
10 START WITH 1
11 INCREMENT BY 1
12 CACHE 10;
13
14 CREATE SEQUENCE seq_IDV -- autoincremento ID Viaggio
15 MINVALUE 1
16 START WITH 1
17 INCREMENT BY 1
18 CACHE 10;

```

Si noti l'utilizzo delle sequenze per incrementare in modo automatico gli ID di fattura, DDT e Viaggio.

2.3 Data Manipulation Language(DML)

In questa sezione avviene il "popolamento" delle tabelle precedentemente implementate, fondamentalmente vengono inserite tramite comando apposito INSERT INTO ... VALUES ... , tutte le informazioni che soddisfano i campi delle tabelle implementate(una per una). *Si noti* che le tabelle: Autista, mezzo (e le specializzazioni autocisterna, rimorchio, trattore e autocarro), fattura, viaggio(e le tabelle di transizione con autista, mezzo e contenitore) non vengono popolate attraverso il popolamento diretto con comando INSERT, ma tramite procedure.

PopolamentoDT

```

1
2  -- PATENTE
3
4 INSERT INTO Patente VALUES ('RA1234567X',TO_DATE('15/05/2023','DD/MM/YYYY'),'CQC','SLVMRC75M15F839B');
5 INSERT INTO Patente VALUES ('RA1234567Z',TO_DATE('17/04/2023','DD/MM/YYYY'),'CQC','MTLLCU81E23F839P');
6 INSERT INTO Patente VALUES ('RA1234568C',TO_DATE('12/02/2023','DD/MM/YYYY'),'CQC','FRSMRA68L11A783S');
7 INSERT INTO Patente VALUES ('RA1234569D',TO_DATE('02/05/2023','DD/MM/YYYY'),'ADR','FRSMRA68L11A783S');
8 INSERT INTO Patente VALUES ('RB1234567E',TO_DATE('06/11/2023','DD/MM/YYYY'),'CQC','PLLLSS88R30B963D');
9 INSERT INTO Patente VALUES ('RB1234568D',TO_DATE('12/08/2024','DD/MM/YYYY'),'CQC','DNTGNN93B01F839G');
10 INSERT INTO Patente VALUES ('RB1234569A',TO_DATE('21/02/2024','DD/MM/YYYY'),'ADR','DNTGNN93B01F839G');
11 INSERT INTO Patente VALUES ('QA1234567Q',TO_DATE('23/01/2024','DD/MM/YYYY'),'CQC','PNDMRN76D06F839E');
12 INSERT INTO Patente VALUES ('QS1234565S',TO_DATE('07/02/2024','DD/MM/YYYY'),'CQC','CRNCRL96D17G309T');
13 INSERT INTO Patente VALUES ('QS123456A4',TO_DATE('02/08/2024','DD/MM/YYYY'),'ADR','CRNCRL96D17G309T');
14 INSERT INTO Patente VALUES ('QZ123456S5',TO_DATE('02/12/2024','DD/MM/YYYY'),'CQC','VRDSDR70B18A783P');
15 INSERT INTO Patente VALUES ('RQ1234567Q',TO_DATE('06/12/2023','DD/MM/YYYY'),'ADR','VRDSDR70B18A783P');
16 INSERT INTO Patente VALUES ('RA1234565B',TO_DATE('07/11/2023','DD/MM/YYYY'),'CQC','STRGNN62L08A509W');
17 INSERT INTO Patente VALUES ('RA1234564V',TO_DATE('09/08/2023','DD/MM/YYYY'),'ADR','STRGNN62L08A509W');
18 INSERT INTO Patente VALUES ('RM1234563X',TO_DATE('11/08/2023','DD/MM/YYYY'),'CQC','DFRCRI91T13F839A');
19 INSERT INTO Patente VALUES ('RM1234564D',TO_DATE('24/11/2023','DD/MM/YYYY'),'CQC','GVNGRN88R19F839G');
20 INSERT INTO Patente VALUES ('QS1234564W',TO_DATE('27/12/2024','DD/MM/YYYY'),'CQC','BMBFNC66H06A509W');
21 INSERT INTO Patente VALUES ('QA1234569U',TO_DATE('28/12/2024','DD/MM/YYYY'),'ADR','BMBFNC66H06A509W');
22 INSERT INTO Patente VALUES ('RT1234564U',TO_DATE('30/07/2025','DD/MM/YYYY'),'CQC','MRACRI91B05H703Q');
23 INSERT INTO Patente VALUES ('RT1234565U',TO_DATE('12/06/2025','DD/MM/YYYY'),'ADR','BCCBRN80R10H703L');
24 INSERT INTO Patente VALUES ('SM1234568L',TO_DATE('07/03/2025','DD/MM/YYYY'),'ADR','BCCBRN80R10H703L');
25 INSERT INTO Patente VALUES ('SA1234567K',TO_DATE('17/03/2025','DD/MM/YYYY'),'CQC','LCTLCU76B08F839Q');
26
27
28  -- CONTRATTO ASSUNZIONE
29
30 INSERT INTO Contratto_assunzione VALUES (TO_DATE('10/08/2019','DD/MM/YYYY'),'SLVMRC75M15F839B',NULL,12.50,17000.00);
31 INSERT INTO Contratto_assunzione VALUES (TO_DATE('12/08/2019','DD/MM/YYYY'),'MTLLCU81E23F839P',NULL,11.00,18200.00);
32 INSERT INTO Contratto_assunzione VALUES (TO_DATE('13/08/2019','DD/MM/YYYY'),'FRSMRA68L11A783S',NULL,10.50,18000.00);
33 INSERT INTO Contratto_assunzione VALUES (TO_DATE('15/08/2019','DD/MM/YYYY'),'PLLLSS88R30B963D',NULL,12.00,17500.00);
34 INSERT INTO Contratto_assunzione VALUES (TO_DATE('18/08/2019','DD/MM/YYYY'),'DNTGNN93B01F839G',NULL,12.00,19800.00);
35 INSERT INTO Contratto_assunzione VALUES (TO_DATE('20/08/2019','DD/MM/YYYY'),'PNDMRN76D06F839E',NULL,13.00,19000.00);
36 INSERT INTO Contratto_assunzione VALUES (TO_DATE('21/08/2019','DD/MM/YYYY'),'CRNCRL96D17G309T',NULL,10.50,17500.00);
37 INSERT INTO Contratto_assunzione VALUES (TO_DATE('05/09/2019','DD/MM/YYYY'),'VRDSDR70B18A783P',TO_DATE('05/08/2019','DD/MM/YYYY'),9.00,15000.00);
38 INSERT INTO Contratto_assunzione VALUES (TO_DATE('20/09/2019','DD/MM/YYYY'),'STRGNN62L08A509W',NULL,10.00,18300.00);
39 INSERT INTO Contratto_assunzione VALUES (TO_DATE('24/09/2019','DD/MM/YYYY'),'DFRCRI91T13F839A',NULL,11.50,18000.00);
40 INSERT INTO Contratto_assunzione VALUES (TO_DATE('25/09/2019','DD/MM/YYYY'),'GVNGRN88R19F839G',NULL,12.00,16000.00);
41 INSERT INTO Contratto_assunzione VALUES (TO_DATE('22/10/2019','DD/MM/YYYY'),'BMBFNC66H06A509W',NULL,13.00,17000.00);
42 INSERT INTO Contratto_assunzione VALUES (TO_DATE('06/11/2019','DD/MM/YYYY'),'VRDSDR70B18A783P',NULL,11.00,17000.00);
43 INSERT INTO Contratto_assunzione VALUES (TO_DATE('13/12/2019','DD/MM/YYYY'),'MRACRI91B05H703Q',NULL,13.75,17600.00);
44 INSERT INTO Contratto_assunzione VALUES (TO_DATE('15/12/2019','DD/MM/YYYY'),'BCCBRN80R10H703L',TO_DATE('26/09/2020','DD/MM/YYYY'),9.50,14600.00);
45 INSERT INTO Contratto_assunzione VALUES (TO_DATE('17/12/2019','DD/MM/YYYY'),'LCTLCU76B08F839Q',NULL,12.50,17200.00);
46 INSERT INTO Contratto_assunzione VALUES (TO_DATE('15/12/2020','DD/MM/YYYY'),'BCCBRN80R10H703L',NULL,12.00,16400.00);
47
48
49  -- AZIENDA
50
51 INSERT INTO Azienda VALUES ('IT','08500000001','F.lliCostanzo_s.r.l.','08111111151','0812111110');
52 INSERT INTO Azienda VALUES ('IT','08500000002','MobiliPierantonio_s.r.l.','08111111152','0812111111');
53 INSERT INTO Azienda VALUES ('IT','08500000003','Orsolini_s.r.l.','08111111151','0812111112');
54 INSERT INTO Azienda VALUES ('IT','08500000004','DeGregori_s.p.a.','08111112151','0812111113');
55 INSERT INTO Azienda VALUES ('IT','08500000005','Cimanonti_s.p.a.','08111111551','0812111114');
56 INSERT INTO Azienda VALUES ('IT','08500000006','F.lliMariani_s.n.c.','08141111151','0812111115');
57 INSERT INTO Azienda VALUES ('IT','08500000007','Cogesirm_s.r.l.','08111111153','0812111116');
58 INSERT INTO Azienda VALUES ('DE','043000001','Lufthansa_s.p.a.','08111111154','0812111117');
59 INSERT INTO Azienda VALUES ('DE','043000002','TheLindleGroup_s.p.a.','08111611151','0812111118');
60 INSERT INTO Azienda VALUES ('IT','08500000008','Camini_s.n.c.','08111111155','0812111119');
61 INSERT INTO Azienda VALUES ('IT','08500000009','DeAngeli_s.r.l.','08111111158','0812111120');
62 INSERT INTO Azienda VALUES ('IT','08500000010','AcciaiSpeciali_Terni_s.p.a.','08111111159','0812111121');
63 INSERT INTO Azienda VALUES ('IT','08500000011','Chimet_s.p.a.','08111111160','0812111122');
64 INSERT INTO Azienda VALUES ('IT','08500000012','Eusider_s.p.a.','08111111161','0812111123');
65 INSERT INTO Azienda VALUES ('IT','08500000013','Safimet_s.p.a.','08111111171','0812111124');
66 INSERT INTO Azienda VALUES ('IT','08500000014','F.lliBertolazzi_s.a.s.','08111111181','0812111125');
67 INSERT INTO Azienda VALUES ('IT','08500000015','MLMK_Verona_s.p.a.','08111111191','0812111126');
68
69  -- SEDE
70
71 INSERT INTO Sede VALUES ('IT','08500000001','Filiale_MI','Montale','20121','Milano','MI');
72 INSERT INTO Sede VALUES ('IT','08500000002','Sede_principale','Pesca','20025','Legnano','MI');
73 INSERT INTO Sede VALUES ('IT','08500000003','Sede_principale','DelleAnde','20064','Gorgonzola','MI');
74 INSERT INTO Sede VALUES ('IT','08500000004','Sede_principale','MonteFrassino','20052','Vignate','MI');
75 INSERT INTO Sede VALUES ('IT','08500000005','Sede_principale','Frontale','43121','Parma','PR');
76 INSERT INTO Sede VALUES ('IT','08500000006','Filiale_FE','PioXI','44121','Ferrara','FE');
77 INSERT INTO Sede VALUES ('IT','08500000007','Filiale_MI','Montella','20121','Milano','MI');
78 INSERT INTO Sede VALUES ('DE','043000001','Sede_principale','Montale','50667','Colonia','K');
79 INSERT INTO Sede VALUES ('DE','043000002','Sede_principale','Indaco','80031','Monaco_d_Baviera','M');
80 INSERT INTO Sede VALUES ('IT','08500000008','Filiale_PR','Rossa','43122','Parma','PR');
81 INSERT INTO Sede VALUES ('IT','08500000009','Filiale_BA','Verde','80121','Bari','BA');
82 INSERT INTO Sede VALUES ('IT','08500000005','Filiale_UD','Laterale','33100','Udine','UD');
83 INSERT INTO Sede VALUES ('IT','08500000010','Filiale_FG','Perle','71100','Foggia','FG');
84 INSERT INTO Sede VALUES ('IT','08500000011','Sede_principale','Cuba','16125','Genova','');
85 INSERT INTO Sede VALUES ('IT','08500000012','Sede_principale','Polo','20121','Milano','MI');
86 INSERT INTO Sede VALUES ('IT','08500000013','Filiale_MI','Carletti','20025','Legnano','MI');
87 INSERT INTO Sede VALUES ('IT','08500000014','Sede_principale','PieroAngela','44121','Ferrara','FE');
88 INSERT INTO Sede VALUES ('IT','08500000012','Filiale_BG','Conti','24121','Bergamo','BG');
89 INSERT INTO Sede VALUES ('IT','08500000015','Filiale_MI','Massima','20121','Milano','MI');

```

```

90
91
92 -- DDT
93
94 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,25,T0_DATE('13/08/2019', 'DD/MM/YYYY'),'1234500');
95 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,24.5,T0_DATE('29/08/2019', 'DD/MM/YYYY'),'123450123');
96 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28.5,T0_DATE('01/09/2019', 'DD/MM/YYYY'),'1234501');
97 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,23,T0_DATE('03/09/2019', 'DD/MM/YYYY'),'1234502');
98 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28,T0_DATE('04/09/2019', 'DD/MM/YYYY'),'1234502');
99 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,29,T0_DATE('06/09/2019', 'DD/MM/YYYY'),'123450025');
100 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28.5,T0_DATE('07/09/2019', 'DD/MM/YYYY'),'123450069');
101 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,26,T0_DATE('12/09/2019', 'DD/MM/YYYY'),'123450111');
102 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28,T0_DATE('14/09/2019', 'DD/MM/YYYY'),'123450112');
103 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,26,T0_DATE('16/09/2019', 'DD/MM/YYYY'),'123450113');
104 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28,T0_DATE('19/09/2019', 'DD/MM/YYYY'),'123450114');
105 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,24.5,T0_DATE('20/09/2019', 'DD/MM/YYYY'),'123450115');
106 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,23.5,T0_DATE('23/09/2019', 'DD/MM/YYYY'),'123450116');
107 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,27,T0_DATE('30/09/2019', 'DD/MM/YYYY'),'1234502');
108 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,22,T0_DATE('07/10/2019', 'DD/MM/YYYY'),'1234503');
109 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,25,T0_DATE('13/10/2019', 'DD/MM/YYYY'),'1234504');
110 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,25,T0_DATE('18/10/2019', 'DD/MM/YYYY'),'1234505');
111 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28,T0_DATE('30/10/2019', 'DD/MM/YYYY'),'1234506');
112 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,29.5,T0_DATE('05/11/2019', 'DD/MM/YYYY'),'1234507');
113 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,22,T0_DATE('11/11/2019', 'DD/MM/YYYY'),'1234508');
114 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,20,T0_DATE('17/12/2019', 'DD/MM/YYYY'),'1234509');
115 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,24,T0_DATE('02/01/2020', 'DD/MM/YYYY'),'1234510');
116 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,28,T0_DATE('03/01/2020', 'DD/MM/YYYY'),'1234511');
117 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,25.5,T0_DATE('06/01/2020', 'DD/MM/YYYY'),'1234512');
118 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,23.5,T0_DATE('06/01/2020', 'DD/MM/YYYY'),'123450117');
119 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,22,T0_DATE('14/01/2020', 'DD/MM/YYYY'),'123450118');
120 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,21,T0_DATE('15/01/2020', 'DD/MM/YYYY'),'123450119');
121 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,24,T0_DATE('19/01/2020', 'DD/MM/YYYY'),'123450124');
122 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,27.5,T0_DATE('21/01/2020', 'DD/MM/YYYY'),'123450125');
123 INSERT INTO DDT VALUES(seq_DDT.NEXTVAL,27,T0_DATE('22/01/2020', 'DD/MM/YYYY'),'123450130');
124
125
126 -- Contratto acquisto
127
128 INSERT INTO Contratto_acquisto VALUES('4403FCA90D4EB9F21',80000.00,'Quadrifoglio_s.r.l.',20000,T0_DATE('25/07/2019', 'DD/MM/YYYY'),'AT665AA');
129 INSERT INTO Contratto_acquisto VALUES('4826EC99D641C4B82',110000.00,'TuttoAuto_s.r.l.',0,T0_DATE('28/07/2019', 'DD/MM/YYYY'),'AT612BB');
130 INSERT INTO Contratto_acquisto VALUES('3F39803D3613698B2',75000.00,'TuttoAuto_s.r.l.',0,T0_DATE('05/08/2019', 'DD/MM/YYYY'),'TR235AA');
131 INSERT INTO Contratto_acquisto VALUES('3CD90EEC1F51089A8',82500.00,'TuttoAuto_s.r.l.',0,T0_DATE('05/08/2019', 'DD/MM/YYYY'),'TR645BB');
132 INSERT INTO Contratto_acquisto VALUES('2FBEE22069D19D39',69500.00,'Quadrifoglio_s.r.l.',50000,T0_DATE('12/08/2019', 'DD/MM/YYYY'),'TR665CC');
133 INSERT INTO Contratto_acquisto VALUES('F800C37527C34A7B3',74500.00,'Quadrifoglio_s.r.l.',36000,T0_DATE('12/08/2019', 'DD/MM/YYYY'),'TR685DD');
134 INSERT INTO Contratto_acquisto VALUES('45BF96E035A4E9C52',95000.00,'Giacomini_s.r.l.',0,T0_DATE('15/08/2019', 'DD/MM/YYYY'),'AC230AA');
135 INSERT INTO Contratto_acquisto VALUES('F6FESA6438130EE77',15000.00,'Giacomini_s.r.l.',0,T0_DATE('26/07/2019', 'DD/MM/YYYY'),'RM880AB');
136 INSERT INTO Contratto_acquisto VALUES('3A70CB9C217D259D9',14000.00,'TuttoAuto_s.r.l.',0,T0_DATE('27/07/2019', 'DD/MM/YYYY'),'RM887AC');
137 INSERT INTO Contratto_acquisto VALUES('7E7ABF19020152A6C',14500.00,'TuttoAuto_s.r.l.',0,T0_DATE('27/07/2019', 'DD/MM/YYYY'),'RM905AD');
138 INSERT INTO Contratto_acquisto VALUES('C16444BD1C9C8F8B',7800.00,'Quadrifoglio_s.r.l.',75000,T0_DATE('28/07/2019', 'DD/MM/YYYY'),'RM915AE');
139 INSERT INTO Contratto_acquisto VALUES('2D15235359EDBD29',8000.00,'Quadrifoglio_s.r.l.',70000,T0_DATE('28/07/2019', 'DD/MM/YYYY'),'RM930AF');
140 INSERT INTO Contratto_acquisto VALUES('F2AB7B57403F7B6FE',13000.00,'Giacomini_s.r.l.',0,T0_DATE('01/08/2019', 'DD/MM/YYYY'),'RM970AG');
141 INSERT INTO Contratto_acquisto VALUES('FC844BE76DF61AAB3',12800.00,'Quadrifoglio_s.r.l.',46500,T0_DATE('01/08/2019', 'DD/MM/YYYY'),'RM988AH');
142 INSERT INTO Contratto_acquisto VALUES('6395CF31E31BF6E69',97200.00,'TuttoAuto_s.r.l.',0,T0_DATE('28/10/2019', 'DD/MM/YYYY'),'AC991AI');
143
144 -- Contratto assicurazione
145
146 INSERT INTO Contratto_assicurazione VALUES('AT665AA',T0_DATE('25/07/2019', 'DD/MM/YYYY'),3000.15,'UnipolSai_s.p.a.');
```

```

181 INSERT INTO Contenitore VALUES('CSQA1234561','20GP');
182 INSERT INTO Contenitore VALUES('CSQA1234562','20GP');
183 INSERT INTO Contenitore VALUES('CSQA1234563','20GP');
184 INSERT INTO Contenitore VALUES('MSCA1234560','20TD');
185 INSERT INTO Contenitore VALUES('MSCA1234561','20TD');
186 INSERT INTO Contenitore VALUES('EVGA1234560','20GP');
187 INSERT INTO Contenitore VALUES('CSQA1234564','20GP');
188 INSERT INTO Contenitore VALUES('CSQA1234565','20GP');
189 INSERT INTO Contenitore VALUES('MSCA1234562','20HR');
190 INSERT INTO Contenitore VALUES('MSCA1234563','20HR');
191 INSERT INTO Contenitore VALUES('EVGA1234561','22BU');
192 INSERT INTO Contenitore VALUES('CSQA1234566','22PC');
193 INSERT INTO Contenitore VALUES('MSCA1234564','20GP');
194 INSERT INTO Contenitore VALUES('MSCA1234565','20GP');
195
196
197
198 -- SEZIONE GUASTI
199
200 -- TIPOLOGIA GUASTO
201
202 INSERT INTO Tipologia_guasto VALUES('Freni');
203 INSERT INTO Tipologia_guasto VALUES('Sterzo');
204 INSERT INTO Tipologia_guasto VALUES('Luci');
205 INSERT INTO Tipologia_guasto VALUES('Circuito_elettrico');
206 INSERT INTO Tipologia_guasto VALUES('Assi');
207 INSERT INTO Tipologia_guasto VALUES('Sospensioni');
208 INSERT INTO Tipologia_guasto VALUES('Foratura');
209 INSERT INTO Tipologia_guasto VALUES('Telaio');
210
211 -- INCIDENTE
212
213 INSERT INTO Incidente VALUES(8,TO_DATE('12/09/2019','DD/MM/YYYY'));
214 INSERT INTO Incidente VALUES(11,TO_DATE('19/09/2019','DD/MM/YYYY'));
215 INSERT INTO Incidente VALUES(15,TO_DATE('08/10/2019','DD/MM/YYYY'));
216 INSERT INTO Incidente VALUES(20,TO_DATE('11/11/2019','DD/MM/YYYY'));
217 INSERT INTO Incidente VALUES(27,TO_DATE('16/01/2020','DD/MM/YYYY'));
218
219 -- REVISIONE
220
221 INSERT INTO Revisione VALUES('AT665AA',TO_DATE('25/07/2019','DD/MM/YYYY'),66.88);
222 INSERT INTO Revisione VALUES('AT612BB',TO_DATE('28/07/2019','DD/MM/YYYY'),66.88);
223 INSERT INTO Revisione VALUES('AT665AA',TO_DATE('29/07/2019','DD/MM/YYYY'),66.88);
224 INSERT INTO Revisione VALUES('TR235AA',TO_DATE('05/08/2019','DD/MM/YYYY'),66.88);
225 INSERT INTO Revisione VALUES('TR645BB',TO_DATE('05/08/2019','DD/MM/YYYY'),45.00);
226 INSERT INTO Revisione VALUES('TR645BB',TO_DATE('08/08/2019','DD/MM/YYYY'),45.00);
227 INSERT INTO Revisione VALUES('TR665CC',TO_DATE('12/08/2019','DD/MM/YYYY'),45.00);
228 INSERT INTO Revisione VALUES('TR685DD',TO_DATE('12/08/2019','DD/MM/YYYY'),45.00);
229 INSERT INTO Revisione VALUES('AC230AA',TO_DATE('15/08/2019','DD/MM/YYYY'),50.00);
230 INSERT INTO Revisione VALUES('AC230AA',TO_DATE('22/08/2019','DD/MM/YYYY'),50.00);
231 INSERT INTO Revisione VALUES('RM880AB',TO_DATE('26/07/2019','DD/MM/YYYY'),45.00);
232 INSERT INTO Revisione VALUES('RM887AC',TO_DATE('27/07/2019','DD/MM/YYYY'),66.88);
233 INSERT INTO Revisione VALUES('RM905AD',TO_DATE('27/07/2019','DD/MM/YYYY'),66.88);
234 INSERT INTO Revisione VALUES('RM915AE',TO_DATE('28/07/2019','DD/MM/YYYY'),66.88);
235 INSERT INTO Revisione VALUES('RM930AF',TO_DATE('28/07/2019','DD/MM/YYYY'),66.88);
236 INSERT INTO Revisione VALUES('RM905AD',TO_DATE('01/08/2019','DD/MM/YYYY'),66.88);
237 INSERT INTO Revisione VALUES('RM970AG',TO_DATE('01/08/2019','DD/MM/YYYY'),66.88);
238 INSERT INTO Revisione VALUES('RM988AH',TO_DATE('01/08/2019','DD/MM/YYYY'),66.88);
239 INSERT INTO Revisione VALUES('RM930AF',TO_DATE('01/08/2019','DD/MM/YYYY'),66.88);
240 INSERT INTO Revisione VALUES('AC991AI',TO_DATE('28/10/2019','DD/MM/YYYY'),45.00);
241 INSERT INTO Revisione VALUES('AT665AA',TO_DATE('25/07/2020','DD/MM/YYYY'),45.00);
242 INSERT INTO Revisione VALUES('AT612BB',TO_DATE('28/07/2020','DD/MM/YYYY'),45.00);
243 INSERT INTO Revisione VALUES('TR235AA',TO_DATE('05/08/2020','DD/MM/YYYY'),50.00);
244 INSERT INTO Revisione VALUES('TR645BB',TO_DATE('05/08/2020','DD/MM/YYYY'),50.00);
245 INSERT INTO Revisione VALUES('TR665CC',TO_DATE('12/08/2020','DD/MM/YYYY'),50.00);
246 INSERT INTO Revisione VALUES('TR685DD',TO_DATE('12/08/2020','DD/MM/YYYY'),50.00);
247 INSERT INTO Revisione VALUES('AC230AA',TO_DATE('15/08/2020','DD/MM/YYYY'),50.00);
248 INSERT INTO Revisione VALUES('RM880AB',TO_DATE('26/07/2020','DD/MM/YYYY'),66.88);
249 INSERT INTO Revisione VALUES('RM887AC',TO_DATE('27/07/2020','DD/MM/YYYY'),66.88);
250 INSERT INTO Revisione VALUES('RM905AD',TO_DATE('27/07/2020','DD/MM/YYYY'),66.88);
251 INSERT INTO Revisione VALUES('RM915AE',TO_DATE('28/07/2020','DD/MM/YYYY'),66.88);
252 INSERT INTO Revisione VALUES('RM930AF',TO_DATE('28/07/2020','DD/MM/YYYY'),66.88);
253 INSERT INTO Revisione VALUES('RM970AG',TO_DATE('01/08/2020','DD/MM/YYYY'),66.88);
254 INSERT INTO Revisione VALUES('RM988AH',TO_DATE('01/08/2020','DD/MM/YYYY'),50.00);
255 INSERT INTO Revisione VALUES('AC991AI',TO_DATE('28/10/2020','DD/MM/YYYY'),45.00);
256
257 -- EsitoP
258
259 INSERT INTO Esito_P VALUES('AT612BB',TO_DATE('28/07/2019','DD/MM/YYYY'));
260 INSERT INTO Esito_P VALUES('AT665AA',TO_DATE('29/07/2019','DD/MM/YYYY'));
261 INSERT INTO Esito_P VALUES('TR235AA',TO_DATE('05/08/2019','DD/MM/YYYY'));
262 INSERT INTO Esito_P VALUES('TR645BB',TO_DATE('08/08/2019','DD/MM/YYYY'));
263 INSERT INTO Esito_P VALUES('TR665CC',TO_DATE('12/08/2019','DD/MM/YYYY'));
264 INSERT INTO Esito_P VALUES('TR685DD',TO_DATE('12/08/2019','DD/MM/YYYY'));
265 INSERT INTO Esito_P VALUES('AC230AA',TO_DATE('22/08/2019','DD/MM/YYYY'));
266 INSERT INTO Esito_P VALUES('RM880AB',TO_DATE('26/07/2019','DD/MM/YYYY'));
267 INSERT INTO Esito_P VALUES('RM887AC',TO_DATE('27/07/2019','DD/MM/YYYY'));
268 INSERT INTO Esito_P VALUES('RM915AE',TO_DATE('28/07/2019','DD/MM/YYYY'));
269 INSERT INTO Esito_P VALUES('RM905AD',TO_DATE('01/08/2019','DD/MM/YYYY'));
270 INSERT INTO Esito_P VALUES('RM970AG',TO_DATE('01/08/2019','DD/MM/YYYY'));
271 INSERT INTO Esito_P VALUES('RM988AH',TO_DATE('01/08/2019','DD/MM/YYYY'));

```

```

272 INSERT INTO Esito_P VALUES('RM930AF',TO_DATE('01/08/2019', 'DD/MM/YYYY'));
273 INSERT INTO Esito_P VALUES('AC991AI',TO_DATE('28/10/2019', 'DD/MM/YYYY'));
274 INSERT INTO Esito_P VALUES('AT665AA',TO_DATE('25/07/2020', 'DD/MM/YYYY'));
275 INSERT INTO Esito_P VALUES('AT612BB',TO_DATE('28/07/2020', 'DD/MM/YYYY'));
276 INSERT INTO Esito_P VALUES('TR235AA',TO_DATE('05/08/2020', 'DD/MM/YYYY'));
277 INSERT INTO Esito_P VALUES('TR645BB',TO_DATE('05/08/2020', 'DD/MM/YYYY'));
278 INSERT INTO Esito_P VALUES('TR665CC',TO_DATE('12/08/2020', 'DD/MM/YYYY'));
279 INSERT INTO Esito_P VALUES('TR685DD',TO_DATE('12/08/2020', 'DD/MM/YYYY'));
280 INSERT INTO Esito_P VALUES('AC230AA',TO_DATE('15/08/2020', 'DD/MM/YYYY'));
281 INSERT INTO Esito_P VALUES('RM880AB',TO_DATE('26/07/2020', 'DD/MM/YYYY'));
282 INSERT INTO Esito_P VALUES('RM887AC',TO_DATE('27/07/2020', 'DD/MM/YYYY'));
283 INSERT INTO Esito_P VALUES('RM905AD',TO_DATE('27/07/2020', 'DD/MM/YYYY'));
284 INSERT INTO Esito_P VALUES('RM915AE',TO_DATE('28/07/2020', 'DD/MM/YYYY'));
285 INSERT INTO Esito_P VALUES('RM930AF',TO_DATE('28/07/2020', 'DD/MM/YYYY'));
286 INSERT INTO Esito_P VALUES('RM970AG',TO_DATE('01/08/2020', 'DD/MM/YYYY'));
287 INSERT INTO Esito_P VALUES('RM988AH',TO_DATE('01/08/2020', 'DD/MM/YYYY'));
288 INSERT INTO Esito_P VALUES('AC991AI',TO_DATE('28/10/2020', 'DD/MM/YYYY'));
289
290 -- EsitoN
291
292 INSERT INTO Esito_N VALUES('AT665AA',TO_DATE('25/07/2019', 'DD/MM/YYYY'));
293 INSERT INTO Esito_N VALUES('TR645BB',TO_DATE('05/08/2019', 'DD/MM/YYYY'));
294 INSERT INTO Esito_N VALUES('AC230AA',TO_DATE('15/08/2019', 'DD/MM/YYYY'));
295 INSERT INTO Esito_N VALUES('RM930AF',TO_DATE('28/07/2019', 'DD/MM/YYYY'));
296
297
298 -- Riparazione
299
300 INSERT INTO Riparazione VALUES('AT665AA',TO_DATE('28/07/2019', 'DD/MM/YYYY'),2100.00);
301 INSERT INTO Riparazione VALUES('RM887AC',TO_DATE('30/07/2019', 'DD/MM/YYYY'),45.00);
302 INSERT INTO Riparazione VALUES('RM930AF',TO_DATE('30/07/2019', 'DD/MM/YYYY'),1300.00);
303 INSERT INTO Riparazione VALUES('TR645BB',TO_DATE('07/08/2019', 'DD/MM/YYYY'),1600.00);
304 INSERT INTO Riparazione VALUES('AC230AA',TO_DATE('20/08/2019', 'DD/MM/YYYY'),950.00);
305 INSERT INTO Riparazione VALUES('AT665AA',TO_DATE('08/09/2019', 'DD/MM/YYYY'),220.00);
306 INSERT INTO Riparazione VALUES('TR235AA',TO_DATE('15/09/2019', 'DD/MM/YYYY'),60.00);
307 INSERT INTO Riparazione VALUES('RM970AG',TO_DATE('22/09/2019', 'DD/MM/YYYY'),550.00);
308 INSERT INTO Riparazione VALUES('RM915AE',TO_DATE('22/09/2019', 'DD/MM/YYYY'),80.00);
309 INSERT INTO Riparazione VALUES('AT612BB',TO_DATE('11/10/2019', 'DD/MM/YYYY'),1200.00);
310 INSERT INTO Riparazione VALUES('TR685DD',TO_DATE('08/11/2019', 'DD/MM/YYYY'),1320.00);
311 INSERT INTO Riparazione VALUES('AT665AA',TO_DATE('14/11/2019', 'DD/MM/YYYY'),3600.00);
312 INSERT INTO Riparazione VALUES('TR665CC',TO_DATE('12/01/2020', 'DD/MM/YYYY'),315.00);
313 INSERT INTO Riparazione VALUES('RM880AB',TO_DATE('18/01/2020', 'DD/MM/YYYY'),885.00);
314 INSERT INTO Riparazione VALUES('AT665AA',TO_DATE('20/01/2019', 'DD/MM/YYYY'),1115.00);
315
316
317
318 -- TABELLE TRANSIZIONE GUASTO
319
320 -- Incidente Provoca
321
322 INSERT INTO Incidente_provoca VALUES(8,TO_DATE('12/09/2019', 'DD/MM/YYYY'),'Foratura');
323 INSERT INTO Incidente_provoca VALUES(11,TO_DATE('19/09/2019', 'DD/MM/YYYY'),'Foratura');
324 INSERT INTO Incidente_provoca VALUES(11,TO_DATE('19/09/2019', 'DD/MM/YYYY'),'Assi');
325 INSERT INTO Incidente_provoca VALUES(15,TO_DATE('08/10/2019', 'DD/MM/YYYY'),'Sospensioni');
326 INSERT INTO Incidente_provoca VALUES(20,TO_DATE('11/11/2019', 'DD/MM/YYYY'),'Telaio');
327 INSERT INTO Incidente_provoca VALUES(20,TO_DATE('11/11/2019', 'DD/MM/YYYY'),'Circuito_elettrico');
328 INSERT INTO Incidente_provoca VALUES(27,TO_DATE('16/01/2020', 'DD/MM/YYYY'),'Assi');
329
330 -- Revisione notifica
331
332 INSERT INTO Revisione_notifica VALUES('AT665AA',TO_DATE('25/07/2019', 'DD/MM/YYYY'),'Freni');
333 INSERT INTO Revisione_notifica VALUES('TR645BB',TO_DATE('05/08/2019', 'DD/MM/YYYY'),'Sterzo');
334 INSERT INTO Revisione_notifica VALUES('TR645BB',TO_DATE('05/08/2019', 'DD/MM/YYYY'),'Assi');
335 INSERT INTO Revisione_notifica VALUES('AC230AA',TO_DATE('15/08/2019', 'DD/MM/YYYY'),'Circuito_elettrico');
336 INSERT INTO Revisione_notifica VALUES('RM930AF',TO_DATE('28/07/2019', 'DD/MM/YYYY'),'Assi');
337
338 -- Riparazione risolve
339
340 INSERT INTO Riparazione_Risolve VALUES('AT665AA',TO_DATE('28/07/2019', 'DD/MM/YYYY'),'Freni');
341 INSERT INTO Riparazione_Risolve VALUES('RM887AC',TO_DATE('30/07/2019', 'DD/MM/YYYY'),'Foratura');
342 INSERT INTO Riparazione_Risolve VALUES('RM930AF',TO_DATE('30/07/2019', 'DD/MM/YYYY'),'Assi');
343 INSERT INTO Riparazione_Risolve VALUES('TR645BB',TO_DATE('07/08/2019', 'DD/MM/YYYY'),'Sterzo');
344 INSERT INTO Riparazione_Risolve VALUES('TR645BB',TO_DATE('07/08/2019', 'DD/MM/YYYY'),'Assi');
345 INSERT INTO Riparazione_Risolve VALUES('AC230AA',TO_DATE('20/08/2019', 'DD/MM/YYYY'),'Circuito_elettrico');
346 INSERT INTO Riparazione_Risolve VALUES('AT665AA',TO_DATE('08/09/2019', 'DD/MM/YYYY'),'Freni');
347 INSERT INTO Riparazione_Risolve VALUES('TR235AA',TO_DATE('15/09/2019', 'DD/MM/YYYY'),'Foratura');
348 INSERT INTO Riparazione_Risolve VALUES('RM970AG',TO_DATE('22/09/2019', 'DD/MM/YYYY'),'Foratura');
349 INSERT INTO Riparazione_Risolve VALUES('RM970AG',TO_DATE('22/09/2019', 'DD/MM/YYYY'),'Assi');
350 INSERT INTO Riparazione_Risolve VALUES('RM915AE',TO_DATE('22/09/2019', 'DD/MM/YYYY'),'Foratura');
351 INSERT INTO Riparazione_Risolve VALUES('AT612BB',TO_DATE('11/10/2019', 'DD/MM/YYYY'),'Sospensioni');
352 INSERT INTO Riparazione_Risolve VALUES('TR685DD',TO_DATE('08/11/2019', 'DD/MM/YYYY'),'Luci');
353 INSERT INTO Riparazione_Risolve VALUES('TR685DD',TO_DATE('08/11/2019', 'DD/MM/YYYY'),'Circuito_elettrico');
354 INSERT INTO Riparazione_Risolve VALUES('AT665AA',TO_DATE('14/11/2019', 'DD/MM/YYYY'),'Telaio');
355 INSERT INTO Riparazione_Risolve VALUES('AT665AA',TO_DATE('14/11/2019', 'DD/MM/YYYY'),'Circuito_elettrico');
356 INSERT INTO Riparazione_Risolve VALUES('TR665CC',TO_DATE('12/01/2020', 'DD/MM/YYYY'),'Freni');
357 INSERT INTO Riparazione_Risolve VALUES('RM880AB',TO_DATE('18/01/2020', 'DD/MM/YYYY'),'Foratura');
358 INSERT INTO Riparazione_Risolve VALUES('RM880AB',TO_DATE('18/01/2020', 'DD/MM/YYYY'),'Assi');
359 INSERT INTO Riparazione_Risolve VALUES('AT665AA',TO_DATE('20/01/2019', 'DD/MM/YYYY'),'Assi');

```

2.4 Trigger

I trigger fanno parte dello standard SQL, di seguito viene mostrata l'implementazione di complessivi 6 trigger. Il trigger in se' viene implementato in un blocco di codice programmato che parte automaticamente, quando si verifica un evento definito dal programmatore. Vengono utilizzati per il controllo dei vincoli dinamici in fase di immissione o aggiornamento dei dati.

TRIGGER - VIAGGIO

Tale Trigger si occupa di verificare se l'inserimento di un viaggio e' corretto. Notare che tale trigger riguarda nello specifico la sola tabella viaggio (e non anche quelle di transizione, che sono trattate a parte). Quando effettuiamo un inserimento dobbiamo accertarci di 3 cose:

- Controllare che la data di inizio del viaggio non sia maggiore della data di fine dello stesso viaggio (non avrebbe senso temporalmente).
- Controllare che il punto di arrivo non sia uguale al punto di partenza. Infatti non avrebbe senso che un'azienda richieda di trasportare delle merci a se' stessa (nella stessa sede).
- La data di emissione del DDT deve coincidere con la data di inizio viaggio.

Implementazione Trigger 1: Viaggio

```
1 CREATE OR REPLACE TRIGGER TR_Viaggio
2 BEFORE INSERT ON Viaggio
3 FOR EACH ROW
4
5 DECLARE
6 data_em DATE;
7 ERR_DATE EXCEPTION; -- data inizio > data fine viaggio
8 DEST_UGUALI EXCEPTION; -- La destinazione di partenza non pu essere uguale a quella di arrivo
9 PARTENZA_DIVERSA_EMISSIONE_DDT EXCEPTION; -- La data di emissione del DDT deve coincidere con la data di inizio viaggio.
10
11
12 BEGIN
13 -- ERRORE DATE INIZIO E FINE VIAGGIO
14 IF :NEW.Data_inizio > :NEW.Data_fine THEN
15 RAISE ERR_DATE;
16 END IF;
17
18 -- DESTINAZIONE PARTENZA UGUALE DESTINAZIONE ARRIVO
19 IF :NEW.SM_partenza = :NEW.SM_arrivo
20 AND :NEW.PIVA_partenza = :NEW.PIVA_arrivo
21 AND :NEW.Sede_partenza = :NEW.Sede_arrivo THEN
22 RAISE DEST_UGUALI;
23 END IF;
24
25 -- DATA PARTENZA DIVERSA DA EMISSIONE DDT
26 SELECT Data_emissione INTO data_em
27 FROM DDT
28 WHERE ID_DDT = :NEW.ID_DDT;
29
30 IF :NEW.Data_inizio != data_em THEN
31 RAISE PARTENZA_DIVERSA_EMISSIONE_DDT;
32 END IF;
33
34 EXCEPTION
35 WHEN ERR_DATE THEN
```

```

36 RAISE_APPLICATION_ERROR(-20001,'La data di inizio non pu  essere maggiore di quella di fine');
37
38 WHEN DEST_UGUALI THEN
39 RAISE_APPLICATION_ERROR(-20002,'La destinazione di partenza non pu  essere uguale a quella di arrivo');
40
41 WHEN PARTENZA_DIVERSA_EMISSIONE_DDT THEN
42 RAISE_APPLICATION_ERROR(-20050,'La data di emissione del DDT deve coincidere con la data di inizio viaggio');
43
44 END;

```

TRIGGER - AUTISTA IN VIAGGIO

Tale Trigger si occupa di verificare se una combinazione in inserimento Autista/Viaggio e' corretta. I controlli che effettuerò sono vari:

- Controllo se per quel viaggio non ho inserito già il massimo numero di autisti (cioe' 2). Quindi verificherò nella stessa tabella il numero di occorrenze dell'ID viaggio corrente. Se ho 0 o 1 occorrenze significa che ho 0 o 1 autista, se ne ho 2 significa che ho 2 autisti (che e' il massimo) e quindi non posso inserire la nuova tupla.
- Controllo le date di scadenza delle patenti. Quindi devo fare un confronto fra la patente dell'autista e la data di fine viaggio del viaggio. Seleziono il MAX(scadenza) perche' potrei avere più patenti inserite per quell'autista.
- Controllo se l'autista e' occupato. Vado a vedere l'ultimo viaggio inserito per quell'autista (escluso questo che stiamo inserendo ora) e ne controllo la date.
Se la data inizio del viaggio nuovo > della data fine del suo ultimo viaggio significa che l'autista, per quella data, e' ancora in viaggio (quindi e' occupato).
- Controllo se il contratto dell'autista e' scaduto. Prendo in considerazione il contratto dell'autista con la data inizio maggiore, in quanto sarà l'ultimo contratto stipulato.
Se la data fine di tale contratto non e' NULL e INOLTRE e' > Data fine viaggio allora significa che (almeno per quella data) il contratto e' scaduto e l'autista non può andare in viaggio.

Implementazione Trigger 2: Autista in Viaggio

```
1 CREATE OR REPLACE TRIGGER TR_Autista_in_viaggio
2 BEFORE INSERT ON Autista_in_viaggio
3 FOR EACH ROW
4
5 DECLARE
6 num_autisti NUMBER(1);
7 scadenza_patente DATE;
8 fine_viaggio DATE;
9 ultimo_viaggioA DATE;
10 ultimo_contrattoA DATE;
11
12 TROPPI_AUTISTI EXCEPTION;
13 PATENTE_SCADUTA EXCEPTION;
14 AUTISTA_OCCUPATO EXCEPTION;
15 CONTRATTO_SCADUTO EXCEPTION;
16
17
18 BEGIN
19
20 -- TROPPI_AUTISTI
21 SELECT COUNT(*) INTO num_autisti
```

```

22 FROM Autista_in_viaggio
23 WHERE ID_viaggio = :NEW.ID_viaggio;
24
25 IF num_autisti = 2 THEN
26 RAISE TROPPI_AUTISTI;
27 END IF;
28
29 -- SCADENZA PATENTE
30 SELECT MAX(Scadenza) INTO scadenza_patente
31 FROM Patente
32 WHERE CF = :NEW.CF;
33
34 SELECT Data_fine INTO fine_viaggio
35 FROM Viaggio
36 WHERE Viaggio.ID_viaggio = :NEW.ID_viaggio;
37
38 IF scadenza_patente < fine_viaggio THEN
39 RAISE PATENTE_SCADUTA;
40 END IF;
41
42 -- AUTISTA OCCUPATO
43 SELECT MAX(Data_fine) INTO ultimo_viaggioA -- Ultimo viaggio autista
44 FROM Viaggio V1 JOIN Autista_in_viaggio V2 ON V1.ID_viaggio = V2.ID_Viaggio
45 WHERE V2.CF = :NEW.CF;
46
47 SELECT Data_fine INTO fine_viaggio -- Data viaggio da inserire
48 FROM Viaggio
49 WHERE ID_Viaggio = :NEW.ID_Viaggio;
50
51 IF fine_viaggio < ultimo_viaggioA THEN
52 RAISE AUTISTA_OCCUPATO;
53 END IF;
54
55 -- CONTRATTO SCADUTO
56 SELECT Data_fine INTO ultimo_contrattoA
57 FROM Contratto_assunzione
58 WHERE Data_inizio = ( SELECT MAX(Data_inizio)
59 FROM Contratto_assunzione
60 WHERE CF = :NEW.CF );
61
62 SELECT Data_fine INTO fine_viaggio
63 FROM Viaggio
64 WHERE Viaggio.ID_viaggio = :NEW.ID_viaggio;
65
66 IF ultimo_contrattoA IS NOT NULL
67 AND fine_viaggio > ultimo_contrattoA THEN
68 RAISE CONTRATTO_SCADUTO;
69 END IF;
70
71
72
73 EXCEPTION
74 WHEN TROPPI_AUTISTI THEN
75 RAISE_APPLICATION_ERROR(-20003,'In un viaggio possono andare al massimo 2 autisti');
76
77 WHEN PATENTE_SCADUTA THEN
78 RAISE_APPLICATION_ERROR(-20004,'La patente dell'autista e' scaduta');
79
80 WHEN AUTISTA_OCCUPATO THEN
81 RAISE_APPLICATION_ERROR(-20005,'L autista e' occupato in un altro viaggio');
82
83 WHEN CONTRATTO_SCADUTO THEN
84 RAISE_APPLICATION_ERROR(-20006,'Il contratto dell'autista e' scaduto');
85
86 END;
87

```


TRIGGER - MEZZO IN VIAGGIO

Tale Trigger si occupa di verificare se una combinazione in inserimento Mezzo/-Viaggio e' corretta. I controlli che effettuerò sono vari:

- Verifico se il mezzo e' stato composto correttamente. Per far ciò mi assicurerò che non possa andare un rimorchio da solo in viaggio (quindi il primo mezzo inserito in tabella non può essere un rimorchio), ed inoltre se ho più mezzi in viaggi mi assicurerò che questi possano essere solo rimorchi.
- Controllo che l'assicurazione non sia scaduta, ovvero che non sia passato un anno dall'ultimo contratto di assicurazione.
- Controllo che la revisione non sia scaduta, ovvero che non sia passato un anno dall'ultima revisione.
- Controllo se il mezzo e' occupato. Vado a vedere l'ultimo viaggio inserito per quel mezzo (escluso questo che stiamo inserendo ora) e ne controllo la date. Se la data inizio del viaggio nuovo ; della data fine del suo ultimo viaggio significa che il mezzo, per quella data, e' ancora in viaggio (quindi e' occupato).

Implementazione Trigger 3: Mezzo in Viaggio

```
1 CREATE OR REPLACE TRIGGER TR_Mezzo_in_viaggio
2 BEFORE INSERT ON Mezzo_in_viaggio
3 FOR EACH ROW
4
5 DECLARE
6 num_mezzi NUMBER;
7 targa_rim Mezzo.Targa%TYPE;
8 ultima_data DATE;
9 fine_viaggio DATE;
10 ERR_MEZZO_COMPOSTO EXCEPTION;
11 ASSICURAZIONE_SCADUTA EXCEPTION;
12 REVISIONE_SCADUTA EXCEPTION;
13 MEZZO_OCCUPATO EXCEPTION;
14
15 BEGIN
16 -- ERR MEZZO COMPOSTO
17 SELECT COUNT(*) INTO num_mezzi
18 FROM Mezzo_in_viaggio
19 WHERE ID_viaggio = :NEW.ID_Viaggio;
20
21 SELECT Targa INTO targa_rim
22 FROM Rimorchio
23 WHERE Targa = :NEW.Targa;
24
25 IF num_mezzi = 0 AND targa_rim IS NOT NULL -- si tratta di un rimorchio
26 OR (num_mezzi = 1 OR num_mezzi = 2) AND targa_rim IS NULL -- non si tratta di un rimorchio
27 OR num_mezzi = 3 THEN
28 RAISE ERR_MEZZO_COMPOSTO;
29 END IF;
30
31 -- ASSICURAZIONE SCADUTA
32 SELECT MAX(Data_ass) INTO ultima_data
33 FROM Contratto_assicurazione
34 WHERE Targa = :NEW.Targa;
35
36 IF SYSDATE - ultima_data > 365 THEN
37 RAISE ASSICURAZIONE_SCADUTA;
```

```

38 END IF;
39
40 -- REVISIONE SCADUTA
41 SELECT MAX(Data_rev) INTO ultima_data
42 FROM Revisione
43 WHERE Targa = :NEW.Targa;
44
45 IF SYSDATE - ultima_data > 365 THEN
46 RAISE REVISIONE_SCADUTA;
47 END IF;
48
49 -- MEZZO OCCUPATO
50 SELECT MAX(Data_fine) INTO ultima_data -- Ultimo viaggio mezzo
51 FROM Viaggio V1 JOIN Mezzo_in_viaggio V2 ON V1.ID_viaggio = V2.ID_Viaggio
52 WHERE V2.Targa = :NEW.Targa;
53
54 SELECT Data_fine INTO fine_viaggio -- Data viaggio da inserire
55 FROM Viaggio
56 WHERE ID_Viaggio = :NEW.ID_Viaggio;
57
58 IF fine_viaggio < ultima_data THEN
59 RAISE MEZZO_OCCUPATO;
60 END IF;
61
62 EXCEPTION
63 WHEN ERR_MEZZO_COMPOSTO THEN
64 RAISE_APPLICATION_ERROR(-20007, 'Un complesso di mezzi così formato non può andare in viaggio');
65
66 WHEN ASSICURAZIONE_SCADUTA THEN
67 RAISE_APPLICATION_ERROR(-20008, 'Un mezzo con assicurazione scaduta non può andare in viaggio');
68
69 WHEN REVISIONE_SCADUTA THEN
70 RAISE_APPLICATION_ERROR(-20009, 'Un mezzo con revisione scaduta non può andare in viaggio');
71
72 WHEN MEZZO_OCCUPATO THEN
73 RAISE_APPLICATION_ERROR(-20010, 'Il mezzo è occupato in un altro viaggio in quelle date');
74
75
76 END;

```

TRIGGER - CONTENITORE IN VIAGGIO

Tale Trigger si occupa di verificare se una combinazione in inserimento Contenitore/Viaggio e' corretta. In questo caso mi basta semplicemente verificare se il container e' occupato in un altro viaggio. Vado a vedere l'ultimo viaggio inserito per quell contenitore (escluso questo che stiamo inserendo ora) e ne controllo la date. Se la data inizio del viaggio nuovo \geq della data fine del suo ultimo viaggio significa che il contenitore, per quella data, e' ancora in viaggio (quindi e' occupato).

Implementazione Trigger 4: Contenitore in Viaggio

```
1 CREATE OR REPLACE TRIGGER TR_Contenitore_in_viaggio
2 BEFORE INSERT ON Contenitore_in_viaggio
3 FOR EACH ROW
4
5 DECLARE
6   ultima_data DATE;
7   fine_viaggio DATE;
8   CONTENITORE_OCCUPATO EXCEPTION;
9
10 BEGIN
11
12   -- CONTENITORE OCCUPATO
13   SELECT MAX(Data_fine) INTO ultima_data -- Ultimo viaggio mezzo
14   FROM Viaggio V1 JOIN contenitore_in_viaggio V2 ON V1.ID_viaggio = V2.ID_Viaggio
15   WHERE V2.ISO_6346 = :NEW.ISO_6346;
16
17   SELECT Data_fine INTO fine_viaggio -- Data viaggio da inserire
18   FROM Viaggio
19   WHERE ID_Viaggio = :NEW.ID_Viaggio;
20
21   IF fine_viaggio < ultima_data THEN
22     RAISE CONTENITORE_OCCUPATO;
23   END IF;
24
25   EXCEPTION
26   WHEN CONTENITORE_OCCUPATO THEN
27     RAISE_APPLICATION_ERROR(-20011,'Il contenitore e' occupato in un altro viaggio in quelle date');
28
29 END;
```

TRIGGER - CONTRATTO DI ASSUNZIONE

Quando inserisco un nuovo di contratto di assunzione per una certa persona, se quella persona in precedenza ha avuto un altro contratto con la stessa azienda, allora devo controllare che la data di inizio del nuovo contratto per la stessa persona non sia minore della data di fine del vecchio contratto (altrimenti avrei due contratti di assunzione definiti sullo stesso periodo di tempo, cosa non possibile). Ovviamente tale errore sorge anche se la data di scadenza del vecchio contratto e' null, cioe' a tempo indeterminato.

Implementazione Trigger 5: Contratto Assunzione

```
1 CREATE OR REPLACE TRIGGER TR_Contratto_assunzione
2 BEFORE INSERT ON Contratto_assunzione
3 FOR EACH ROW
4
5 DECLARE
6 FineContrattoA DATE;
7 DATE_CONTRATTO_SOVRAPPOSTE EXCEPTION; -- Non posso avere due contratti di assunzioni della stessa persona definiti sullo stesso periodo.
8
9 BEGIN
10 SELECT MAX(Data_fine) INTO FineContrattoA
11 FROM Contratto_assunzione
12 WHERE CF = :NEW.CF;
13
14 IF FineContrattoA > :NEW.Data_inizio THEN
15 RAISE DATE_CONTRATTO_SOVRAPPOSTE;
16 END IF;
17
18 EXCEPTION
19 WHEN DATE_CONTRATTO_SOVRAPPOSTE THEN
20 RAISE_APPLICATION_ERROR(-20012, 'La data di inizio del nuovo contratto non pu essere precedente a quella di fine del vecchio contratto');
21
22 END;
```

TRIGGER - INCIDENTE

In questo Database teniamo in considerazione solo gli incidenti che si verificano durante un viaggio. Per tal motivo, se si verifica un incidente, la data in cui si verifica tale incidente deve essere compresa all'interno delle date del viaggio.

Implementazione Trigger 6: Incidente

```
1 CREATE OR REPLACE TRIGGER TR_Incidente
2 BEFORE INSERT ON Incidente
3 FOR EACH ROW
4
5 DECLARE
6 inizio_viaggio DATE;
7 fine_viaggio DATE;
8 INC_NON_IN_VIAGGIO EXCEPTION;
9
10 BEGIN
11 SELECT Data_inizio,Data_fine INTO inizio_viaggio,fine_viaggio
12 FROM Viaggio
13 WHERE ID_Viaggio = :NEW.ID_Viaggio;
14
15 IF :NEW.Data_inc < inizio_viaggio OR :NEW.Data_inc > fine_viaggio THEN -- cioè', se l'incidente e' avvenuto PRIMA o DOPO il viaggio
16 RAISE INC_NON_IN_VIAGGIO;
17 END IF;
18
19 EXCEPTION
20 WHEN INC_NON_IN_VIAGGIO THEN
21 RAISE_APPLICATION_ERROR(-20013,'La data di un incidente deve essere compresa in quelle del viaggio');
22
23 END;
```

2.5 Procedure

L'ideale all'interno di un DBMS sarebbe che l'inserimento delle nuove tuple avvengano attraverso delle procedure. Per una questione di semplicità abbiamo considerato solo alcune delle procedure che riguardavano gli inserimenti (non abbiamo opportuno creare una procedura per l'inserimento di ogni singola tabella o comunque di un insieme di tabelle). Consideriamo in particolare le procedure INS_AUTISTA, INS_FATTURA, INS_MEZZO e INS_VIAGGIO, le quali sono utilizzabili solo da specifici utenti (Vedi sezione 1.6).

2.5.1 Descrizione procedura Inserimento Autista

Ci occupiamo di inserire solo i dati anagrafici dell'autista, quindi gli argomenti di tale procedura saranno gli stessi degli attributi della tabella Autista.

IMPLEMENTAZIONE

```
1 CREATE OR REPLACE PROCEDURE INS_Autista(  
2   CF      Autista.CF%TYPE,  
3   Nome    Autista.Nome%TYPE,  
4   Cognome Autista.Cognome%TYPE,  
5   Data_Nascita Autista.Data_Nascita%TYPE,  
6   Via     Autista.Via%TYPE,  
7   CAP     Autista.CAP%TYPE,  
8   Comune Autista.Comune%TYPE,  
9   PROV    Autista.PROV%TYPE,  
10  Tel_Personale Autista.Tel_Personale%TYPE,  --Numero di casa(11) o Numero Cellulare (10)  
11  Tel_Aziendale Autista.Tel_Aziendale%TYPE  
12 )  
13  
14 AS  
15  
16 BEGIN  
17 INSERT INTO Autista VALUES(CF, Nome, Cognome,  
18 Data_Nascita, Via, CAP, Comune, PROV, Tel_personale, Tel_aziendale);  
19  
20 END INS_Autista;
```

2.5.2 Descrizione procedura Inserimento Mezzo

Ci occupiamo di inserire il nuovo mezzo, identificato da una targa. Ricordiamo che specializziamo i mezzi in vari tipi (Autocarro, Rimorchio, Trattore e Autocisterna) i quali alla loro volta avranno delle caratteristiche proprie. Per tale motivo gli argomenti di tale procedura saranno: Targa, TipoM (tipo del mezzo) e Tratto (Tipo per rimorchio, Peso massimo per autocarro o Capienza in Lt per autocisterna). Procederemo ad inserire la tupla prima nella tabella generale Mezzo (che contiene solo Targa) e poi, attraverso un controllo su TipoM, verifichiamo in quale delle tabelle specializzate dobbiamo inserire la tupla (Targa,Tratto). Notare che si tratta di una specializzazione disgiunta totale, quindi se abbiamo inserito un tipo di mezzo diverso da quelli previste verrà generato un errore.

IMPLEMENTAZIONE

```
1 CREATE OR REPLACE PROCEDURE INS_Mezzo(  
2   Targa Mezzo.Targa%TYPE,  
3   TipoM VARCHAR2,  
4   Tratto VARCHAR2  
5 )  
6  
7 AS  
8  
9 BEGIN  
10  INSERT INTO Mezzo VALUES(Targa);  
11  
12  CASE TipoM  
13    WHEN 'Autocarro' THEN INSERT INTO Autocarro VALUES(Targa,TO_NUMBER(Tratto));  
14    WHEN 'Rimorchio' THEN INSERT INTO Rimorchio VALUES(Targa,Tratto);  
15    WHEN 'Trattore' THEN INSERT INTO Trattore VALUES(Targa);  
16    WHEN 'Autocisterna' THEN INSERT INTO Autocisterna VALUES(Targa,TO_NUMBER(Tratto));  
17    ELSE RAISE_APPLICATION_ERROR(-20080,'Il tipo di mezzo inserito non è valido');  
18  END CASE;  
19  
20  
21 END INS_Mezzo;
```

2.5.3 Descrizione procedura Inserimento Fattura

Ci occupiamo di inserire i dati della fattura, quindi gli argomenti di tale procedura saranno gli stessi degli attributi della tabella Fattura MA con la differenza che non avremo bisogno di inserire il numero della fattura, in quanto sarà individuata automaticamente dal sistema (trattandosi di un ID che si autoincrementa).

IMPLEMENTAZIONE

```
1 CREATE OR REPLACE PROCEDURE INS_Fattura(  
2   Data_emissione Fattura.Data_emissione%TYPE,  
3   Importo Fattura.Importo%TYPE,  
4   Stato_membro Fattura.Stato_membro%TYPE,  
5   P_IVA Fattura.P_IVA%TYPE  
6 )  
7 AS  
8  
9 BEGIN  
10 INSERT INTO Fattura VALUES(seq_IDF.NEXTVAL,Data_emissione,Importo,Stato_membro,P_IVA);  
11  
12 END INS_Fattura;
```


2.5.4 Descrizione procedura Inserimento Viaggio

Ci occupiamo di inserire TUTTI i dati relativi al viaggio, quindi oltre a voler conoscere le informazioni riguardo al viaggio di per sé, vorremo inserire anche gli autisti, i mezzi e i contenitori che hanno partecipato in quel viaggio. Per tal motivo, il numero di argomenti di tale procedura è elevato, ma necessario affinché avvenga un inserimento comodo e privo di incoerenze. Anche in tal caso non avremo bisogno di inserire il numero del viaggio, in quanto sarà individuata automaticamente dal sistema (trattandosi di un ID che si autoincrementa). Possiamo distinguere 4 passaggi:

- L'inserimento dei dati riguardo il viaggio di per sé all'interno della tabella Viaggio.
- L'inserimento dei dati riguardo gli autisti in viaggio (ne inseriremo il CF). Notare come se ne possano inserire al massimo due in quanto in un viaggio non possono andare più di due autisti, tuttavia un viaggio per essere effettuato ha necessariamente bisogno di almeno un autista (per tale motivo AUT1 non può essere vuoto), invece il secondo autista è opzionale e il tentativo di inserimento verrà effettuato solo se AUT2 è diverso da NULL.
- L'inserimento dei dati riguardo ai mezzi in viaggio (ne inseriamo la targa). Notare come se ne possano inserire al massimo 3 in quanto diamo per assunto che in un viaggio possa andare un solo mezzo composto, e che un mezzo possa essere composto a sua volta al massimo da 3 mezzi (infatti è la massima combinazione possibile, data da: Trattore + rimorchio + rimorchio). Notare che almeno un mezzo deve andare in viaggio (per tal motivo MEZ1 non può essere vuoto), mentre gli altri due mezzi sono opzionali e il tentativo di inserimento verrà effettuato solo se sono diversi da NULL.
- L'inserimento dei dati riguardo i contenitori (container) in viaggio è puramente opzionale, in quanto non è detto che in un viaggio un mezzo debba per forza partire con dei container. Il tentativo di inserimento avverrà solo se sono diversi da NULL.

Ergo, noteremo come potremo effettuare l'inserimento su ben 4 tabelle: Viaggio, Autista_in_viaggio, Mezzo_in_viaggio e (opz) Contenitore_in_viaggio.

IMPLEMENTAIZONE

```

1 CREATE OR REPLACE PROCEDURE INS_Viaggio(
2 DI Viaggio.Data_inizio%TYPE,
3 DF Viaggio.Data_fine%TYPE,
4 NO Viaggio.Numero_ore%TYPE,
5 KM Viaggio.KM%TYPE,
6 IDF Viaggio.ID_Fattura%TYPE,
7 SMP Viaggio.SM_partenza%TYPE,
8 PIVA_P Viaggio.PIVA_partenza%TYPE,
9 SP Viaggio.Sede_partenza%TYPE,
10 SMA Viaggio.SM_arrivo%TYPE,
11 PIVA_A Viaggio.PIVA_arrivo%TYPE,
12 SA Viaggio.Sede_arrivo%TYPE,
13 ID_DDT Viaggio.ID_DDT%TYPE,
14
15 AUT1 Autista.CF%TYPE,
16 AUT2 Autista.CF%TYPE,
17
18 MEZ1 Mezzo.Targa%TYPE,
19 MEZ2 Mezzo.Targa%TYPE,
20 MEZ3 Mezzo.Targa%TYPE,
21
22 ISO1 Contenitore.ISO_6346%TYPE,
23 ISO2 Contenitore.ISO_6346%TYPE
24 )
25
26 IS
27 IDV Viaggio.Id_Viaggio%TYPE;
28
29 BEGIN
30 IDV := seq_IDV.NEXTVAL;
31 INSERT INTO Viaggio VALUES(IDV,DI,DF,NO,KM,IDF,SMP,PIVA_P,SP,SMA,PIVA_A,SA,ID_DDT);
32
33
34 IF AUT1 IS NULL THEN
35 RAISE_APPLICATION_ERROR(-20100,'Il campo dell'autista 1 non pu essere vuoto');
36 END IF;
37 INSERT INTO Autista_in_viaggio VALUES(AUT1,IDV);
38
39 IF AUT2 IS NOT NULL THEN
40 INSERT INTO Autista_in_viaggio VALUES(AUT2,IDV);
41 END IF;
42
43
44 IF MEZ1 IS NULL THEN
45 RAISE_APPLICATION_ERROR(-20100,'Il campo del mezzo 1 non pu essere vuoto');
46 END IF;
47 INSERT INTO Mezzo_in_viaggio VALUES(MEZ1,IDV);
48
49 IF MEZ2 IS NOT NULL THEN
50 INSERT INTO Mezzo_in_viaggio VALUES(MEZ2,IDV);
51 END IF;
52
53 IF MEZ3 IS NOT NULL THEN
54 INSERT INTO Mezzo_in_viaggio VALUES(MEZ3,IDV);
55 END IF;
56
57
58 IF ISO1 IS NOT NULL THEN
59 INSERT INTO Contenitore_in_viaggio VALUES(ISO1,IDV);
60 END IF;
61
62 IF ISO2 IS NOT NULL THEN
63 INSERT INTO Contenitore_in_viaggio VALUES(ISO2,IDV);
64 END IF;
65
66
67 END INS_Viaggio;

```

2.6 Chiamate a procedure

Chiamate InsViaggio

```
1 CALL INS_VIAGGIO(TO_DATE('13/08/2019', 'DD/MM/YYYY'),TO_DATE('14/08/2019', 'DD/MM/YYYY'),16,1521,1,
2 'IT','08500000001','Filiale_MI','IT','08500000008','Filiale_PR',1,
3 'SLVMRC75M15F839B',null,'TR235AA','RM880AB',null,'CSQA1234560',null);
4
5 CALL INS_VIAGGIO(TO_DATE('29/08/2019', 'DD/MM/YYYY'),TO_DATE('31/08/2019', 'DD/MM/YYYY'),22,2010,2,
6 'IT','08500000002','Sede_principale','IT','08500000010','Filiale_FG',2,
7 'MTLLCU81E23F839P',null,'TR665CC','RM887AC',null,'CSQA1234561',null);
8
9 CALL INS_VIAGGIO(TO_DATE('01/09/2019', 'DD/MM/YYYY'),TO_DATE('02/09/2019', 'DD/MM/YYYY'),17,1615,3,
10 'IT','08500000002','Sede_principale','IT','08500000015','Filiale_MI',3,
11 'FRSMRA68L11A783S',null,'TR685DD','RM930AF',null,'CSQA1234562',null);
12
13 CALL INS_VIAGGIO(TO_DATE('03/09/2019', 'DD/MM/YYYY'),TO_DATE('04/09/2019', 'DD/MM/YYYY'),17,1609,4,
14 'IT','08500000003','Sede_principale','IT','08500000011','Sede_principale',4,
15 'PLLLSS88R30B963D',null,'TR685DD','RM915AE','RM880AB','CSQA1234563','CSQA1234564');
16
17 CALL INS_VIAGGIO(TO_DATE('04/09/2019', 'DD/MM/YYYY'),TO_DATE('05/09/2019', 'DD/MM/YYYY'),16,1535,5,
18 'IT','08500000003','Sede_principale','IT','08500000012','Sede_principale',5,
19 'FRSMRA68L11A783S',null,'AC230AA',null,null,null,null);
20
21 CALL INS_VIAGGIO(TO_DATE('06/09/2019', 'DD/MM/YYYY'),TO_DATE('07/09/2019', 'DD/MM/YYYY'),16,1558,6,
22 'IT','08500000004','Sede_principale','IT','08500000013','Filiale_MI',6,
23 'MTLLCU81E23F839P',null,'AT665AA',null,null,null,null);
24
25 CALL INS_VIAGGIO(TO_DATE('07/09/2019', 'DD/MM/YYYY'),TO_DATE('08/09/2019', 'DD/MM/YYYY'),18,1825,7,
26 'IT','08500000004','Sede_principale','IT','08500000014','Sede_principale',7,
27 'SLVMRC75M15F839B',null,'AT612BB','RM905AD',null,'EVGA1234560',null);
28
29 CALL INS_VIAGGIO(TO_DATE('12/09/2019', 'DD/MM/YYYY'),TO_DATE('13/09/2019', 'DD/MM/YYYY'),17,1745,8,
30 'IT','08500000005','Filiale_UD','IT','08500000013','Filiale_MI',8,
31 'DNTGNN93B01F839G',null,'TR235AA','RM887AC',null,'CSQA1234560',null);
32
33 CALL INS_VIAGGIO(TO_DATE('14/09/2019', 'DD/MM/YYYY'),TO_DATE('15/09/2019', 'DD/MM/YYYY'),12,1123,9,
34 'IT','08500000006','Filiale_FE','IT','08500000008','Filiale_PR',9,
35 'PNDMRN76D06F839E',null,'AC230AA',null,null,null,null);
36
37 CALL INS_VIAGGIO(TO_DATE('16/09/2019', 'DD/MM/YYYY'),TO_DATE('17/09/2019', 'DD/MM/YYYY'),16,1682,10,
38 'IT','08500000004','Sede_principale','IT','08500000013','Filiale_MI',10,
39 'FRSMRA68L11A783S',null,'AC230AA',null,null,null,null);
40
41 CALL INS_VIAGGIO(TO_DATE('19/09/2019', 'DD/MM/YYYY'),TO_DATE('20/09/2019', 'DD/MM/YYYY'),13,1336,11,
42 'IT','08500000006','Filiale_FE','IT','08500000008','Filiale_PR',11,
43 'SLVMRC75M15F839B',null,'TR685DD','RM970AG',null,'MSCA1234564',null);
44
45 CALL INS_VIAGGIO(TO_DATE('20/09/2019', 'DD/MM/YYYY'),TO_DATE('24/09/2019', 'DD/MM/YYYY'),33,3225,12,
46 'IT','08500000011','Sede_principale','DE','043000001','Sede_principale',12,
47 'CRNCRL96D17G309T','VRSDSR70B18A783P','TR665CC','RM988AH','RM880AB','MSCA1234562','MSCA1234563');
48
49 CALL INS_VIAGGIO(TO_DATE('23/09/2019', 'DD/MM/YYYY'),TO_DATE('25/09/2019', 'DD/MM/YYYY'),20,1956,13,
50 'IT','08500000002','Sede_principale','IT','08500000010','Filiale_FG',13,
51 'STRGNN62L08A509W',null,'TR645BB','RM915AE',null,'EVGA1234561',null);
52
53 CALL INS_VIAGGIO(TO_DATE('30/09/2019', 'DD/MM/YYYY'),TO_DATE('01/10/2019', 'DD/MM/YYYY'),16,1521,14,
54 'IT','08500000001','Filiale_MI','IT','08500000008','Filiale_PR',14,
55 'PNDMRN76D06F839E',null,'AT665AA',null,null,null,null);
56
57 CALL INS_VIAGGIO(TO_DATE('07/10/2019', 'DD/MM/YYYY'),TO_DATE('08/10/2019', 'DD/MM/YYYY'),16,1535,15,
58 'IT','08500000003','Sede_principale','IT','08500000012','Sede_principale',15,
59 'FRSMRA68L11A783S',null,'AT612BB','RM905AD',null,null,null);
60
61 CALL INS_VIAGGIO(TO_DATE('13/10/2019', 'DD/MM/YYYY'),TO_DATE('15/10/2019', 'DD/MM/YYYY'),18,1828,16,
62 'IT','08500000006','Filiale_FE','IT','08500000012','Sede_principale',16,
63 'MTLLCU81E23F839P',null,'AC230AA',null,null,null,null);
64
65 CALL INS_VIAGGIO(TO_DATE('18/10/2019', 'DD/MM/YYYY'),TO_DATE('22/10/2019', 'DD/MM/YYYY'),32,3019,17,
66 'IT','08500000011','Sede_principale','DE','043000002','Sede_principale',17,
67 'SLVMRC75M15F839B','DFRCRI91T13F839A','TR235AA','RM930AF',null,null,null);
68
69 CALL INS_VIAGGIO(TO_DATE('30/10/2019', 'DD/MM/YYYY'),TO_DATE('01/11/2019', 'DD/MM/YYYY'),17,1620,18,
70 'IT','08500000002','Sede_principale','IT','08500000015','Filiale_MI',18,
71 'GVNGRN88R19F839G',null,'AC230AA',null,null,null,null);
72
73 CALL INS_VIAGGIO(TO_DATE('05/11/2019', 'DD/MM/YYYY'),TO_DATE('06/11/2019', 'DD/MM/YYYY'),16,1521,19,
74 'IT','08500000001','Filiale_MI','IT','08500000008','Filiale_PR',19,
75 'BMBFNC66H06A509W',null,'TR685DD','RM988AH',null,null,null);
76
77 CALL INS_VIAGGIO(TO_DATE('11/11/2019', 'DD/MM/YYYY'),TO_DATE('12/11/2019', 'DD/MM/YYYY'),16,1521,20,
78 'IT','08500000001','Filiale_MI','IT','08500000008','Filiale_PR',20,
79 'STRGNN62L08A509W',null,'AT665AA',null,null,null,null);
80
81 CALL INS_VIAGGIO(TO_DATE('17/12/2019', 'DD/MM/YYYY'),TO_DATE('19/12/2019', 'DD/MM/YYYY'),18,1842,21,
82 'IT','08500000009','Filiale_BA','IT','08500000011','Sede_principale',21,
83 'CRNCRL96D17G309T',null,'AT612BB',null,null,null,null);
84
85 CALL INS_VIAGGIO(TO_DATE('02/01/2020', 'DD/MM/YYYY'),TO_DATE('03/01/2020', 'DD/MM/YYYY'),17,1745,22,
```

```

86 'IT','08500000007','Filiale_MI','IT','08500000014','Sede_principale',22,
87 'DNTGNN93B01F839G',null,'TR235AA','RM880AB',null,'EVGA1234560',null);
88
89 CALL INS_VIAGGIO(TO_DATE('03/01/2020','DD/MM/YYYY'),TO_DATE('04/01/2020','DD/MM/YYYY'),16,1624,23,
90 'IT','08500000007','Filiale_MI','IT','08500000012','Sede_principale',23,
91 'SLVMRC75M15F839B',null,'TR645BB','RM887AC',null,null,null);
92
93 CALL INS_VIAGGIO(TO_DATE('06/01/2020','DD/MM/YYYY'),TO_DATE('10/01/2020','DD/MM/YYYY'),33,3115,24,
94 'IT','08500000011','Sede_principale','DE','043000002','Sede_principale',24,
95 'MRACRI91B05H703Q','BCCBRN80R10H703L','TR665CC','RM905AD',null,'MSCA1234564',null);
96
97 CALL INS_VIAGGIO(TO_DATE('06/01/2020','DD/MM/YYYY'),TO_DATE('08/01/2020','DD/MM/YYYY'),18,1842,25,
98 'IT','08500000009','Filiale_BA','IT','08500000011','Sede_principale',25,
99 'PLLLSS88R30B963D',null,'AC230AA',null,null,null,null);
100
101 CALL INS_VIAGGIO(TO_DATE('14/01/2020','DD/MM/YYYY'),TO_DATE('16/01/2020','DD/MM/YYYY'),18,1825,26,
102 'IT','08500000004','Sede_principale','IT','08500000014','Sede_principale',26,
103 'FRSMRA68L11A783S',null,'AC991AI',null,null,null,null);
104
105 CALL INS_VIAGGIO(TO_DATE('15/01/2020','DD/MM/YYYY'),TO_DATE('16/01/2020','DD/MM/YYYY'),17,1624,27,
106 'IT','08500000007','Filiale_MI','IT','08500000012','Filiale_BG',27,
107 'LCTLCU76B08F839Q',null,'AT665AA',null,null,null,null);
108
109 CALL INS_VIAGGIO(TO_DATE('19/01/2020','DD/MM/YYYY'),TO_DATE('20/01/2020','DD/MM/YYYY'),17,1760,28,
110 'IT','08500000006','Filiale_FE','IT','08500000012','Filiale_BG',28,
111 'CRNCRL96D17G309T',null,'TR235AA','RM988AH',null,null,null);
112
113 CALL INS_VIAGGIO(TO_DATE('21/01/2020','DD/MM/YYYY'),TO_DATE('23/01/2020','DD/MM/YYYY'),19,1889,29,
114
115
116 CALL INS_VIAGGIO(TO_DATE('22/01/2020','DD/MM/YYYY'),TO_DATE('24/01/2020','DD/MM/YYYY'),21,2101,30,
117 'IT','08500000009','Filiale_BA','IT','08500000011','Sede_principale',30,
118 'SLVMRC75M15F839B',null,'TR685DD','RM930AF',null,'EVGA1234561',null);

```

Chiamate InsMezzo

```
1      CALL INS_MEZZO('AT665AA','Autocarro','30');
2  CALL INS_MEZZO('AT612BB','Autocarro','30');
3  CALL INS_MEZZO('TR235AA','Trattore',null);
4  CALL INS_MEZZO('TR645BB','Trattore',null);
5  CALL INS_MEZZO('TR665CC','Trattore',null);
6  CALL INS_MEZZO('TR685DD','Trattore',null);
7  CALL INS_MEZZO('AC230AA','Autocisterna','25000');
8  CALL INS_MEZZO('RM880AB','Rimorchio','FLATBED');
9  CALL INS_MEZZO('RM887AC','Rimorchio','FLATBED');
10 CALL INS_MEZZO('RM905AD','Rimorchio','FLATBED');
11 CALL INS_MEZZO('RM915AE','Rimorchio','FLATBED');
12 CALL INS_MEZZO('RM930AF','Rimorchio','ENCLOSED');
13 CALL INS_MEZZO('RM970AG','Rimorchio','REFRIGERATED');
14 CALL INS_MEZZO('RM988AH','Rimorchio','STEPDECK');
15 CALL INS_MEZZO('AC991AI','Autocisterna','25000');
```

Chiamate InsFattura

```
1  CALL INS_FATTURA(TO_DATE('12/08/2019', 'DD/MM/YYYY'),1600.00,'IT','08500000001');
2  CALL INS_FATTURA(TO_DATE('28/08/2019', 'DD/MM/YYYY'),1750.00,'IT','08500000002');
3  CALL INS_FATTURA(TO_DATE('30/08/2019', 'DD/MM/YYYY'),1180.00,'IT','08500000002');
4  CALL INS_FATTURA(TO_DATE('02/09/2019', 'DD/MM/YYYY'),1700.00,'IT','08500000003');
5  CALL INS_FATTURA(TO_DATE('02/09/2019', 'DD/MM/YYYY'),1825.00,'IT','08500000003');
6  CALL INS_FATTURA(TO_DATE('05/09/2019', 'DD/MM/YYYY'),1200.00,'IT','08500000004');
7  CALL INS_FATTURA(TO_DATE('08/09/2019', 'DD/MM/YYYY'),1400.00,'IT','08500000004');
8  CALL INS_FATTURA(TO_DATE('11/09/2019', 'DD/MM/YYYY'),1785.00,'IT','08500000005');
9  CALL INS_FATTURA(TO_DATE('13/09/2019', 'DD/MM/YYYY'),1630.00,'IT','08500000006');
10 CALL INS_FATTURA(TO_DATE('15/09/2019', 'DD/MM/YYYY'),1675.00,'IT','08500000004');
11 CALL INS_FATTURA(TO_DATE('18/09/2019', 'DD/MM/YYYY'),1890.00,'IT','08500000006');
12 CALL INS_FATTURA(TO_DATE('18/09/2019', 'DD/MM/YYYY'),3100.00,'DE','043000001');
13 CALL INS_FATTURA(TO_DATE('22/09/2019', 'DD/MM/YYYY'),1450.00,'IT','08500000002');
14 CALL INS_FATTURA(TO_DATE('29/09/2019', 'DD/MM/YYYY'),1600.00,'IT','08500000001');
15 CALL INS_FATTURA(TO_DATE('05/10/2019', 'DD/MM/YYYY'),1320.00,'IT','08500000003');
16 CALL INS_FATTURA(TO_DATE('11/10/2019', 'DD/MM/YYYY'),1750.00,'IT','08500000006');
17 CALL INS_FATTURA(TO_DATE('16/10/2019', 'DD/MM/YYYY'),2900.00,'DE','043000002');
18 CALL INS_FATTURA(TO_DATE('28/10/2019', 'DD/MM/YYYY'),2100.00,'IT','08500000002');
19 CALL INS_FATTURA(TO_DATE('02/11/2019', 'DD/MM/YYYY'),1750.00,'IT','08500000001');
20 CALL INS_FATTURA(TO_DATE('08/11/2019', 'DD/MM/YYYY'),1400.00,'IT','08500000001');
21 CALL INS_FATTURA(TO_DATE('16/12/2019', 'DD/MM/YYYY'),1450.00,'IT','08500000009');
22 CALL INS_FATTURA(TO_DATE('01/01/2020', 'DD/MM/YYYY'),1690.00,'IT','08500000007');
23 CALL INS_FATTURA(TO_DATE('02/01/2020', 'DD/MM/YYYY'),1600.00,'IT','08500000007');
24 CALL INS_FATTURA(TO_DATE('05/01/2020', 'DD/MM/YYYY'),3400.00,'DE','043000002');
25 CALL INS_FATTURA(TO_DATE('05/01/2020', 'DD/MM/YYYY'),1500.00,'IT','08500000009');
26 CALL INS_FATTURA(TO_DATE('11/01/2020', 'DD/MM/YYYY'),1990.00,'IT','08500000004');
27 CALL INS_FATTURA(TO_DATE('16/01/2020', 'DD/MM/YYYY'),1435.00,'IT','08500000007');
28 CALL INS_FATTURA(TO_DATE('18/01/2020', 'DD/MM/YYYY'),1715.00,'IT','08500000006');
29 CALL INS_FATTURA(TO_DATE('20/01/2020', 'DD/MM/YYYY'),1350.00,'IT','08500000010');
30 CALL INS_FATTURA(TO_DATE('21/01/2020', 'DD/MM/YYYY'),1760.00,'IT','08500000009');
```

Chiamate InsAutista

```

1 CALL INS_AUTISTA('SLVMRC75M15F839B','Marco','Silvestri',
2 TO_DATE('15/08/1975','DD/MM/YYYY'),'Brambilla','80026','Casoria','NA','0811111111','3291111111');
3 CALL INS_AUTISTA('MTLLCU81E23F839P','Luca','Mirtilli',
4 TO_DATE('23/05/1981','DD/MM/YYYY'),'Monte_Rosa','81031','Aversa','CE','0811111112','3291111112');
5 CALL INS_AUTISTA('FRSMRA68L11A783S','Mario','Frassino',
6 TO_DATE('11/07/1968','DD/MM/YYYY'),'Monte_Bianco','80024','Cardito','NA','0811111113','3291111113');
7 CALL INS_AUTISTA('PLLS88R30B963D','Alessio','Palladino',
8 TO_DATE('30/10/1988','DD/MM/YYYY'),'Cervino','80040','Cercola','CE','0811111114','3291111114');
9 CALL INS_AUTISTA('DNTGNN93B01F839G','Gennaro','Dante',
10 TO_DATE('01/02/1993','DD/MM/YYYY'),'Marmolada','80022','Arzano','NA','0811111115','3291111115');
11 CALL INS_AUTISTA('PNDMRN76D06F839E','Mariano','Prandelli',
12 TO_DATE('06/04/1976','DD/MM/YYYY'),'Delle_Rocche','80026','Casoria','NA','0811111116','3291111116');
13 CALL INS_AUTISTA('CRNCRL96D17G309T','Carlo','Creanza',
14 TO_DATE('17/04/1996','DD/MM/YYYY'),'Crinale','81031','Aversa','NA','0811111117','3291111117');
15 CALL INS_AUTISTA('VRSDR70B18A783P','Sandro','Verdi',
16 TO_DATE('18/02/1970','DD/MM/YYYY'),'Appennini','80023','Caivano','NA','0811111118','3291111118');
17 CALL INS_AUTISTA('STRGNN62L08A509W','Giovanni','Storti',
18 TO_DATE('08/07/1962','DD/MM/YYYY'),'Monte_Pollino','80027','Frattamaggiore','NA','0811111119','3291111119');
19 CALL INS_AUTISTA('DFRCRI91T13F839A','Ciro','Di_Fiore',
20 TO_DATE('13/12/1991','DD/MM/YYYY'),'Monte_Cimone','80026','Casoria','NA','0811111120','3291111120');
21 CALL INS_AUTISTA('GVNGRN88R19F839G','Giorno','Giovanna',
22 TO_DATE('19/10/1988','DD/MM/YYYY'),'Monte_Miletto','80026','Casoria','NA','0811111121','3291111121');
23 CALL INS_AUTISTA('BMBFNC66H06A509W','Franco','Bombardelli',
24 TO_DATE('06/06/1966','DD/MM/YYYY'),'Delle_Acque','80023','Caivano','NA','0811111122','3291111122');
25 CALL INS_AUTISTA('MRACRI91B05H703Q','Ciro','Amore',
26 TO_DATE('05/02/1991','DD/MM/YYYY'),'Puglia','84040','Alfano','SA','0811111123','3291111123');
27 CALL INS_AUTISTA('BCCBRN80R10H703L','Bruno','Bucciarati',
28 TO_DATE('10/10/1980','DD/MM/YYYY'),'Roma','84011','Amalfi','SA','0811111124','3291111124');
29 CALL INS_AUTISTA('LCTLCU76B08F839Q','Luca','Locatelli',
30 TO_DATE('08/02/1976','DD/MM/YYYY'),'Monte_Vettore','82015','Durazzano','BN','0811111125','3291111125');

```

2.7 Data Control Language

u_gest_autisti, e' l'utente che si occupa dell'inserimento e della gestione degli autisti all'interno del DBMS.

```

1 GRANT CONNECT, CREATE SESSION TO u_gest_autisti;
2 GRANT ALL ON INS_AUTISTI TO u_gest_autisti;

```

u_gestmezzi, analogamente si occupa dell'inserimento e della gestione dei mezzi.

```

1 GRANT CONNECT, CREATE SESSION TO u_gestmezzi;
2 GRANT ALL ON INS_MEZZI TO u_gestmezzi;

```

Infine, *u_gest_viaggi*, e' l'utente che si occupa dell'inserimento e la gestione dei viaggi e delle fatture allegate ai viaggi.

```

1 GRANT CONNECT, CREATE SESSION TO u_gest_viaggi;
2 GRANT ALL ON INS_FATTURA TO u_gest_viaggi;
3 GRANT ALL ON INS_VIAGGIO TO u_gest_viaggi;

```