

*UNIVERSITÀ DEGLI STUDI DI ROMA “TOR VERGATA”*



*Corso di laurea triennale in Informatica*

## RADBD:

Database per la gestione di un sistema informativo radiologico

Studente: Eleonora Galgano  
Matricola: 0284291

Docente: Prof.ssa Loredana Vigliano

Anno Accademico 2020-2021

## Indice

<b>1</b>	<b>Introduzione .....</b>	<b>7</b>
<b>2</b>	<b>Regole Aziendali .....</b>	<b>8</b>
<b>3</b>	<b>Analisi dei requisiti.....</b>	<b>11</b>
3.1	<i>Obiettivo .....</i>	11
3.2	<i>Specifiche dei requisiti.....</i>	11
<b>4</b>	<b>Glossario dei Termini.....</b>	<b>13</b>
<b>5</b>	<b>Progettazione logica.....</b>	<b>16</b>
5.1	<i>Schema ER Logico.....</i>	16
5.2	<i>Normalizzazione .....</i>	17
<b>6</b>	<b>Progettazione fisica.....</b>	<b>18</b>
6.1	<i>Schema ER normalizzato .....</i>	18
6.2	<i>Descrizione Entità e Relazioni.....</i>	19
<b>7</b>	<b>Comandi per la creazione del database .....</b>	<b>20</b>
7.1	<i>ComuneResidenza.....</i>	20
7.2	<i>Zona.....</i>	20
7.3	<i>Paziente .....</i>	20
7.4	<i>Prestazione .....</i>	21
7.5	<i>Esenzione .....</i>	21
7.6	<i>PacchettoPrestazionale .....</i>	21
7.7	<i>Dipartimento.....</i>	22
7.8	<i>CategoriaEsposizione .....</i>	22
7.9	<i>Qualifica .....</i>	22

7.10	<i>Impiegato</i> .....	22
7.11	<i>Dosimetro</i> .....	23
7.12	<i>Sala</i> .....	23
7.13	<i>Calendario</i> .....	24
7.14	<i>Orario</i> .....	24
7.15	<i>Richiesta</i> .....	24
7.16	<i>Associata</i> .....	25
7.17	<i>Esame</i> .....	25
7.18	<i>Referta</i> .....	26
<b>8</b>	<b>Inserimenti dati nel Database (Esempi)</b> .....	<b>27</b>
8.1	<i>ComuneResidenza</i> .....	27
8.2	<i>Zona</i> .....	27
8.3	<i>Paziente</i> .....	28
8.4	<i>Prestazione</i> .....	29
8.5	<i>Esenzione</i> .....	30
8.6	<i>PacchettoPrestazionale</i> .....	31
8.7	<i>Dipartimento</i> .....	32
8.8	<i>CategoriaEsposizione</i> .....	33
8.9	<i>Qualifica</i> .....	33
8.10	<i>Impiegato</i> .....	33
8.11	<i>Dosimetro</i> .....	35
8.12	<i>Sala</i> .....	36
8.13	<i>Calendario</i> .....	37
8.14	<i>Orario</i> .....	38
8.15	<i>Richiesta</i> .....	39
8.16	<i>Associata</i> .....	40
8.17	<i>Esame</i> .....	41

## RADB

8.18	<i>Referta</i> .....	42
<b>9</b>	<b>Funzione Festivo</b> .....	<b>44</b>
<b>10</b>	<b>Trigger ed Event</b> .....	<b>44</b>
10.1	<i>Trigger Esenzione non Applicabile</i> .....	44
10.2	<i>Event Apri Agenda</i> .....	45
<b>11</b>	<b>View</b> .....	<b>47</b>
11.1	<i>View Planning Appuntamenti</i> .....	47
11.2	<i>View worklist per Amministrativi</i> .....	49
11.3	<i>View worklist per Tecnici di Radiologia</i> .....	50
11.4	<i>View Worklist Medici di Radiologia</i> .....	50
11.5	<i>View di consultazione esami</i> .....	50
<b>12</b>	<b>Stored Procedure</b> .....	<b>52</b>
12.1	<i>PrenotaAppuntamento</i> .....	52
12.2	<i>AccettaAppuntamento</i> .....	54
12.3	<i>CancellaAppuntamento</i> .....	55
12.4	<i>Estrai Dati Ricevuta</i> .....	56
12.5	<i>Aggiorna Paziente</i> .....	57
12.6	<i>Ritira Referto</i> .....	58
12.7	<i>Completa Esame</i> .....	59
12.8	<i>Annnulla Esame</i> .....	60
12.9	<i>Referta Esame</i> .....	61
<b>13</b>	<b>Sicurezza</b> .....	<b>62</b>
<b>14</b>	<b>Query</b> .....	<b>73</b>
14.1	<i>Tecnici di turno</i> .....	73
14.2	<i>Disponibilità</i> .....	74

<b>14.3</b>	<i>Prenotazioni a seguire</i> .....	75
<b>14.4</b>	<i>Storico Esami</i> .....	76
<b>14.5</b>	<i>Ammontare del costo totale per esenzioni</i> .....	77
<b>14.6</b>	<i>Mancata disdetta</i> .....	78
<b>14.7</b>	<i>Report Rimborsi</i> .....	79
<b>14.8</b>	<i>Dosimetria 1</i> .....	80
<b>14.9</b>	<i>Dosimetria 2</i> .....	81
<b>14.10</b>	<i>Radioprotezione</i> .....	82
<b>14.11</b>	<i>Non idoneità alla somministrazione di contrasto</i> .....	84
<b>14.12</b>	<i>Possibile Contagio</i> .....	85
<b>14.13</b>	<i>Valutazione di produttività</i> .....	86
<b>14.14</b>	<i>Valutazione di efficienza</i> .....	87
<b>14.15</b>	<i>Gestione Emergenze</i> .....	87
<b>14.16</b>	<i>Indagine Statistica 1</i> .....	88
<b>14.17</b>	<i>Indagine Statistica 2</i> .....	89
<b>14.18</b>	<i>Indagine statistica 3</i> .....	90
<b>14.19</b>	<i>Esami per patologie croniche</i> .....	91
<b>14.20</b>	<i>Indagine demografica</i> .....	92
<b>15</b>	<b>Query in algebra relazionale</b> .....	<b>93</b>
<b>15.1</b>	<i>Storico Esami</i> .....	93
<b>15.2</b>	<i>Dosimetria 2</i> .....	93
<b>16</b>	<b>Ottimizzazione Query</b> .....	<b>94</b>
<b>16.1</b>	<i>Disponibilità</i> .....	94
<b>17</b>	<b>MongoDB</b> .....	<b>99</b>
<b>17.1</b>	<i>Ramo Dosimetria</i> .....	99
<b>17.1.1</b>	<i>CATEGORIA d'appartenenza</i> .....	101
<b>17.1.2</b>	<i>Inquadramento in CATEGORIA</i> .....	102

## RADB

17.1.3	<i>Dosimetria 1</i> .....	103
17.1.4	<i>Dosimtria 2</i> .....	106
17.2	<i>Ramo Info Pazienti</i> .....	111
17.2.1	<i>Ricerca dati pazienti per età</i> .....	113
17.2.2	<i>Seleziona Pazienti</i> .....	114
17.2.3	<i>Indagine Statistica 1</i> .....	116
17.2.4	<i>Indagine Statistica 2</i> .....	116
17.3	<i>Confronto tempi di Esecuzione</i> .....	119
17.3.1	<i>Dosimetria 1</i> .....	119
17.3.2	<i>Indagine Statistica 1</i> .....	120
17.4	<i>Conclusioni</i> .....	123

# 1 Introduzione

RADB (*RAdiological DataBase*) è un database per un RIS( *Radiological information System*, Sistema Informativo Radiologico), pensato per la gestione dei dati in un reparto di Diagnostica per Immagini. Viene utilizzato dai professionisti della medicina radiologica per tenere traccia dei dati dei pazienti e delle informazioni generate tipicamente nel corso dell'esecuzione degli esami e della diagnosi. Viene utilizzato per gestire e distribuire i dati dei pazienti all'interno del reparto. Implementa funzioni di registrazione dei pazienti e programmazione degli appuntamenti. Permette di mantenere traccia anche delle informazioni riguardanti il personale, tra cui anche quelle relative all'aspetto radioprotezionistico. Garantisce la sicurezza impedendo a persone non autorizzate di accedere a informazioni riservate dei pazienti. Semplifica inoltre la produzione di *report*.

## 2 Regole Aziendali

Relativamente la presente struttura, per il Dipartimento di Diagnostica per Immagini vigono le seguenti politiche aziendali:

- Ogni impiegato deve avere accesso al Sistema Informatico Radiologico e mantenere aggiornati i *record* di propria competenza relativi l'*iter* diagnostico di ogni paziente.
- Un paziente che richiede o per cui si richiede una prestazione deve obbligatoriamente essere in possesso di una richiesta proveniente da un dipartimento interno dell'ospedale o dal medico curante.
- Il numero ID del paziente deve essere univoco, generato al momento della prima richiesta. Pertanto nel momento in cui un paziente o un dipartimento presenta una prescrizione, qualora non sia già presente nel database, dovrà essere inserito il nuovo paziente con la relativa anagrafica ed eventuali note utili.
- Ogni richiesta deve essere identificata con un ID univoco e riportare il quesito diagnostico, il dipartimento prescrivente, la priorità assegnata, l'eventuale esenzione associata e la data e ora di in cui viene effettuata.
- Ogni richiesta può essere relativa a una singola prestazione, alla quale verrà assegnato un posto disponibile in agenda a seconda delle necessità del richiedente.
- Alle richieste viene associata una priorità da 0 a 3 come segue:
  - ‘3’ identifica le urgenze
  - ‘2’ identifica le urgenze brevi
  - ‘1’ identifica le differibili
  - ‘0’ identifica assenza di priorità
- Al fine di garantire che le richieste interne con alta e media priorità vengano soddisfatte in tempi brevi, alcune sale sono riservate alle prenotazioni da parte di reparti della struttura.
- In caso di mancata disdetta della prestazione prenotata e non fruible, viene richiesto, anche ai titolari di esenzione, il pagamento dell'intero importo della stessa.
- I report delle ricevute devono riportare l'elenco delle prestazioni, l'imponibile e il netto, e devono conteggiare tutte le prestazioni accettate della giornata per il paziente.
- Nell'eventualità che un paziente sia in possesso di un'esenzione valida sul territorio questo verrà esentato dal pagamento della prestazione, qualora sia previsto un pacchetto prestazionale.

- Lo stato di una richiesta deve sempre essere aggiornato dal momento in cui essa viene inserita, fin quando il paziente si presenta il giorno dell'appuntamento e deve essere riportato il nome dell'amministrativo che ne esegue l'ultimo aggiornamento.
- Il reparto garantisce le prestazioni di Ecografia, Radiologia convenzionale, Mammografia, Radiologia Dentale, Risonanza Magnetica Nucleare e Tomografia Computerizzata.
- Per ogni impiegato si devono mantenere i *record* di dose assorbita mensili
- Ogni impiegato a seconda della sua qualifica è inquadrato in una categoria di esposizione.
- Le categorie vengono definite come segue:
  - Lavoratori esposti di categoria A: Lavoratori suscettibili di assorbire una dose di radiazioni ionizzanti che supera i  $6 \text{ mSv}^1$  di dose efficace in un anno solare;
  - Lavoratori esposti di categoria B: Lavoratori suscettibili di assorbire una dose di radiazioni ionizzanti che non supera i  $6 \text{ mSv}$  di dose efficace in un anno solare;
  - Lavoratori di categoria C: Lavoratori non esposti e suscettibili di ricevere una dose efficace di radiazione in un anno solare non superiore ai limiti stabiliti per la popolazione:  $1 \text{ mSv}$
- Alle categorie viene associato un limite di dose come segue:
  - Categoria A:  $20 \text{ mSv}$
  - Categoria B:  $6 \text{ mSv}$
  - Categoria C:  $1 \text{ mSv}$
- A ogni impiegato è fatto obbligo di indossare il dosimetro.
- Esso deve essere riconsegnato con scadenza mensile.
- Alla riconsegna deve essere comunicato la data in cui si è iniziato a indossarlo e la data dell'ultimo giorno in cui si è indossato.
- Qualora la dose per ogni impiegato superasse i limiti stabiliti per la propria categoria di esposizione deve essere aperta un'indagine per stabilire se la sovraesposizione sia dovuta a malfunzionamenti dei dispositivi di protezione, dei macchinari o a negligenza dell'impiegato.
- Il Tecnico di Radiologia è abilitato all'esecuzione di tutti gli esami.
- Lo studio eseguito deve riportare la data e ora di esecuzione, il numero di immagini accettate, il numero di quelle cancellate, la dose di radiazioni erogata al paziente in  $\text{mGycm}^{22}$  ed informazioni utili sull'esecuzione dell'esame.

---

<sup>1</sup> millisievert: unità di misura della dose equivalente e dose efficace.

<sup>2</sup> milliGray per centimetro quadro: unità di misura del prodotto dose in ingresso al paziente per area.

## RADB

- A uno studio completato seguirà poi la produzione del referto.
- Le immagini diagnostiche non possono essere rilasciate ai pazienti senza il referto.
- Per il ritiro del referto il paziente deve mostrare l'apposito foglio rilasciato al momento dell'accettazione con i riferimenti della richiesta.
- Tutte le sale sono operative dalle 8:00 alle 19:00 nei giorni feriali, fatta eccezione per quelle riservate ai ricoverati e pazienti del Pronto Soccorso le quali sono costantemente operative.
- La agenda delle sale è prenotabile fino a un anno solare a decorrere dalla data corrente
- Deve essere garantita la riservatezza e la sicurezza dei dati registrati nella struttura.

## **3 Analisi dei requisiti**

### **3.1 Obiettivo**

Si vuole realizzare una base di dati per la gestione del flusso di dati di un reparto di Diagnostica per Immagini di una struttura ospedaliera pubblica nel Lazio. In accordo con le regole aziendali, si vogliono rappresentare tutti i dati utili relativi all'esecuzione di uno studio diagnostico, i dati dei pazienti a cui viene prescritta una prestazione, i dipartimenti che la richiedono, informazioni del personale coinvolto nell'erogazione dei servizi proposti dalla struttura, e i dati relativi alla gestione dell'agenda degli appuntamenti.

### **3.2 Specifiche dei requisiti**

Relativamente i pazienti, identificati da un codice di identificazione univoco, rappresentiamo tutti i dati anagrafici a essi relativi (nome, cognome, età, città di nascita e indirizzo), contatti personali e d'emergenza, e note riguardanti condizioni patologiche rilevanti come ad esempio allergie, patologie, croniche e non, che possano influire sull'esecuzione di un esame.

Per le zone identificate dal codice postale di zona rappresentiamo il comune di appartenenza

Relativamente il comune di Residenza identificato dal nome del comune rappresentiamo la regione di appartenenza

Per i dipartimenti, identificati dal proprio nome, rappresentiamo una loro breve descrizione.

Relativamente la sala, identificata da un numero di stanza, rappresentiamo a chi è riservata (ricoverati, pazienti di Pronto Soccorso o ambulatoriali), il tipo di *Imaging System*<sup>3</sup> installato.

Per il calendario rappresentiamo tutte le date trascorse e le date di un anno solare a decorrere dal giorno corrente.

Per l'orario si vogliono rappresentare le fasce orarie in cui è possibile prenotare un appuntamento.

Per la richiesta identificata da un ID rappresentiamo l'ID del paziente per cui è stata scritta, il dipartimento richiedente, il quesito diagnostico, la priorità, la data e l'ora in cui viene presentata, la prestazione per cui si fa richiesta, lo stato della richiesta, il numero

---

<sup>3</sup> *Sistema di imaging*: Apparecchiatura dedicata a una metodologia di produzione delle immagini diagnostiche.

## RADB

della stanza prenotata, la data e l'ora dell'appuntamento e la matricola dell'amministrativo che ha fatto l'ultimo intervento sul record.

Per le esenzioni identificate da un codice esenzione, rappresentiamo le condizioni degli aventi diritto e il territorio di validità.

Per le associazioni di esenzione alle richieste rappresentiamo l'ID della richiesta col relativo codice di esenzione associato.

Per le prestazioni identificate da un codice prestazione, rappresentiamo la tipologia d'esame, il nome dell'esame, gli LDR (livelli di dose di riferimento), e il costo.

Per il pacchetto prestazionale rappresentiamo il codice esenzione e codice prestazione che lo identificano e che corrispondono a una prestazione coperta da quell'esenzione.

Relativamente l'esame identificato da un numero univoco, riportiamo l'ID della richiesta relativa, la matricola del TSRM che ha eseguito l'esame, la data e ora del completamento, lo stato dell'esame, il numero di immagini accettate, il numero di immagini rifiutate, la dose erogata al paziente in  $mGy\cdot cm^2$  e un campo commenti dove registrare informazioni riguardo l'esame svolto.

Relativamente la refertazione rappresentiamo l'ID dell'esame che viene refertato, la matricola del medico che si incarica della scrittura del referto, il referto e la data di refertazione.

Riguardo il personale, identificato da numero di matricola, rappresentiamo i dati degli impiegati: dati anagrafici, i contatti, la qualifica e l'username con cui hanno accesso al sistema.

Relativamente alla qualifica identificata mediante un'abbreviazione rappresentiamo una breve descrizione e la categoria d'esposizione in cui è inquadrata.

Per i dosimetri riconsegnati dagli impiegati, associati a un numero d'identificazione univoco, si vogliono rappresentare la matricola dell'impiegato che lo ha indossato, il periodo di inizio e fine validità e la dose da esso registrata.

Per la categoria d'esposizione, identificata da un proprio codice di categoria si vuole rappresentare una breve descrizione e limiti di dose, in  $mSv$ , ad essa corrispondenti.

## 4 Glossario dei Termini

A seguire un elenco dei principali termini utilizzati, al fine di consentire una migliore comprensione del database e chiarire eventuali ambiguità

TERMINI	DESCRIZIONE	SINONIMI	COLLEGAMENTI
<u>Associazione</u>	Si intende l'atto di inserire in una prescrizione un codice esenzione		Richiesta, Esenzione
<u>Calendario</u>	Insieme di date		Richiesta
<u>Categoria Esposizione</u>	Classe di individui soggetti alle stesse norme per l'esposizione a radiazioni ionizzanti		Qualifica
<u>Comune Residenza</u>	Ente locale Autonomo		Zona
<u>Dipartimento</u>	Unità operativa della struttura dedicata a una determinata area medica	Reparto	Richiesta
<u>Dosimetro</u>	Dispositivo di protezione individuale che registra la dose assorbita di chi lo indossa.		Impiegato
<u>Esame</u>	Indagine diagnostica a cui vengono sottoposti i pazienti	Studio	Richiesta, Referta, Impiegato
<u>Esenzione</u>	Sollevamento dall'obbligo di corrispondere l'imponibile di una o più prestazioni soggetta a determinate condizioni		Prestazione, Pacchetto Prestazionale, Richiesta, Associazione
<u>Impiegato</u>	Persona che presta la propria	Operatore	Qualifica, Esame,

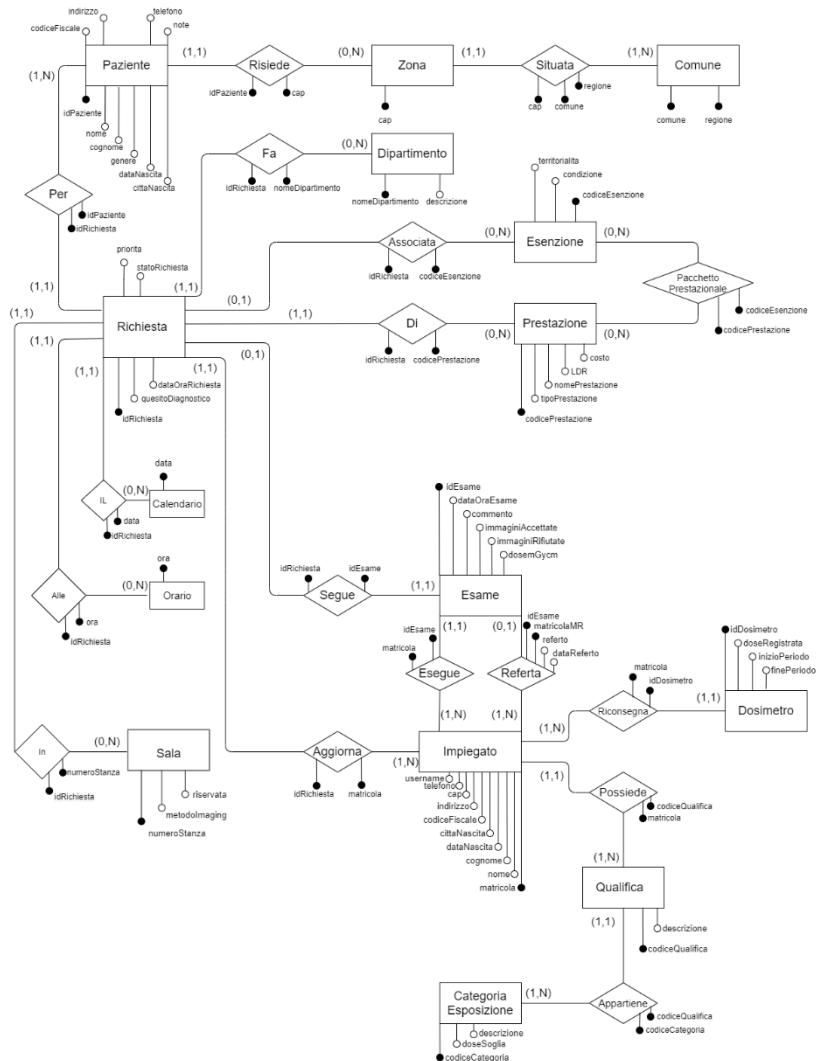
## RADB

	attività professionale		Richiesta, Referta
<u>Orario</u>	Ore prestabilite per la programmazione degli appuntamenti	Ora	Richiesta
<u>Pacchetto Prestazionale</u>	Coppie di esenzioni e prestazioni che danno diritto all'esenzione dall'aliquota di partecipazione diretta dei cittadini		Esenzione, Prestazione
<u>Paziente</u>	Persona affidata alle cure di un medico		Richiesta
<u>Prestazione</u>	Atto clinico di natura diagnostica prestato al cittadino		Richiesta, Pacchetto Prestazionale, Esenzione
<u>Qualifica</u>	Titolo professionale		Impiegato, Categoria d'esposizione
<u>Referto</u>	Relazione scritta rilasciata dal medico di un esame cui è stato sottoposto un paziente		Impiegato, Esame
<u>Richiesta</u>	Prescrizione scritta di un esame clinico	Prescrizione	Paziente, Dipartimento, Esenzione, Associazione, Prestazione, Sala, Calendario, Orario, Esame, Impiegato
<u>Sala</u>	Luogo fisico in cui è installato un dispositivo per	Stanza	Richiesta

	l'esecuzione di indagini diagnostiche		
<u>Zona</u>	Area circoscritta di un comune identificata da un codice postale		Comune Residenza, Paziente

# 5 Progettazione logica

## 5.1 Schema ER Logico



## 5.2 Normalizzazione

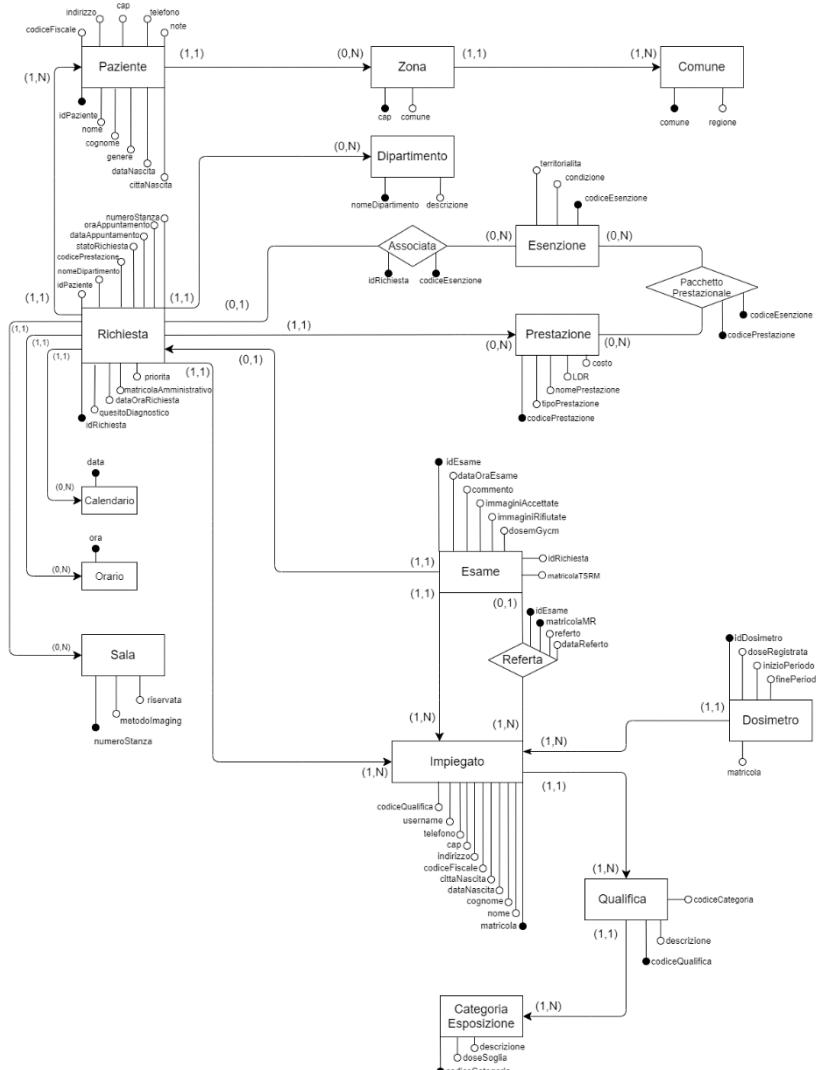
Per il passaggio dallo schema ER logico a schema ER normalizzato, verranno eseguite le seguenti operazioni:

- Le entità coinvolte in relazioni molti a molti rimarranno invariate;
- A seguito di relazioni uno a molti l'entità con cardinalità inferiore erediterà la chiave esterna dell'entità con cardinalità superiore;
- La relazione “Associata” (uno a molti) rimarrà come tabella opzionale, in quanto qualora “Richiesta” ereditasse la chiave esterna “codiceEsenzione” potrebbe contenere un numero di campi null superiore al 20% del totale delle tuple.
- La relazione “Segue” (uno a molti) sparirà, verrà mantenuta l'entità “Esame”, la quale erediterà la chiave “idRichiesta”: un esame può non esistere ancora se una richiesta è in stato “Prenotato”, o non potrà mai esistere se lo stato è “Cancellato”, e se “Richiesta” inglobasse tutti gli attributi di Esame risulterebbe un elevato numero di campi null;
- La relazione “Referta” (uno a molti) rimarrà come tabella opzionale, in quanto l'esistenza del referto dipende dallo stato dell'esame il quale può essere annullato e non prevedere quindi alcun referto e se Esame ereditasse gli attributi della relazione, ne risulterebbe un potenziale numero elevato di campi null.

# 6 Progettazione fisica

## 6.1 Schema ER normalizzato

Lo schema è in terza forma normale, in quanto è in prima e in seconda forma normale e tutti gli attributi dipendono funzionalmente in maniera non transitiva dalle chiavi delle entità.



## 6.2 Descrizione Entità e Relazioni

COMUNERESIDENZA(comune, regione)

ZONA(cap, comune(FK))

PAZIENTE(idPaziente, nome, cognome, genere, dataNascita, cittaNascita, codiceFiscale indirizzo, cap(FK), telefono, note)

PRESTAZIONE(codicePrestazione, tipoPrestazione, nomePrestazione, LDR, costo)

ESENZIONE(codiceEsenzione, condizione, territorio)

PACCHETTOPRESTAZIONALE(codicePrestazione(FK), codiceEsenzione(FK))

DIPARTIMENTO(nomeDipartimento, descrizione)

CATEGORIAESPOSIZIONE(codiceCategoria, doseSoglia, descrizione)

QUALIFICA(codiceQualifica, codiceCategoria(FK), descrizione)

IMPIEGATO(matricola, nome, cognome, dataNascita, cittaNascita, codiceFiscale, indirizzo, cap, telefono, codiceQualifica(FK), username)

DOSIMETRO(idDosimetro, matricola(FK), doseRegistrata, inizioPeriodo, finePeriodo)

SALA(numeroStanza, metodoImaging, riservata)

ORARIO(ora)

CALENDARIO(data)

RICHIESTA(idRichiesta, idPaziente(FK), quesitoDiagnostico, dataOraRichiesta, codicePrestazione(FK), priorita, nomeDipartimento(FK), dataAppuntamento(FK), oraAppuntamento(FK), numeroStanza(FK), statoRichiesta, matricolaAmministrativo(FK))

ASSOCIATA(idRichiesta(FK), codiceEsenzione(FK))

ESAME(idEsame, idRichiesta(FK), dataOraEsame, matricolaTSRM(FK), statoEsame, immaginiAccettate, immaginiRifiutate, dosemGycm, commento)

REFERTA(idEsame(FK), matricolaMR(FK), referto, dataOraReferto)

## 7 Comandi per la creazione del database

### 7.1 ComuneResidenza

```
CREATE TABLE ComuneResidenza (
    comune varchar(50) NOT NULL PRIMARY KEY,
    regione varchar(50) NOT NULL);
```

### 7.2 Zona

```
CREATE TABLE Zona (
    cap int NOT NULL PRIMARY KEY,
    comune varchar(50) NOT NULL, FOREIGN KEY (comune) REFERENCES
    comuneResidenza (comune) ON UPDATE CASCADE ON DELETE NO
    ACTION);
```

### 7.3 Paziente

```
CREATE TABLE Paziente (
    idPaziente int NOT NULL AUTO_INCREMENT PRIMARY KEY,
    nome varchar(20) NOT NULL,
    cognome varchar(20) NOT NULL,
    genere varchar(1) NOT NULL,
    dataNascita date NOT NULL,
    cittaNascita varchar(15) NOT NULL,
    codiceFiscale varchar(16) NOT NULL UNIQUE,
    indirizzo varchar(50) NOT NULL,
    cap int NOT NULL, FOREIGN KEY (cap) REFERENCES zona (cap) ON
    UPDATE CASCADE ON DELETE NO ACTION,
    telefono varchar(11) NOT NULL,
```

note text);

## 7.4 Prestazione

```
CREATE TABLE Prestazione (
codicePrestazione varchar(10) NOT NULL PRIMARY KEY,
tipoPrestazione varchar(3) NOT NULL,
nomePrestazione varchar(50) NOT NULL,
LDR decimal(6,2) NOT NULL DEFAULT 0.00,
costo decimal(6,2) NOT NULL);
```

## 7.5 Esenzione

```
CREATE TABLE Esenzione (
codiceEsenzione varchar(7) NOT NULL PRIMARY KEY,
condizione varchar(50) NOT NULL,
territorio varchar(20) NOT NULL);
```

## 7.6 PacchettoPrestazionale

```
CREATE TABLE PacchettoPrestazionale (
codicePrestazione varchar(15) NOT NULL, FOREIGN KEY (codicePrestazione)
REFERENCES Prestazione (codicePrestazione) ON UPDATE CASCADE ON DELETE
NO ACTION,
codiceEsenzione varchar(7) NOT NULL, FOREIGN KEY (codiceEsenzione)
REFERENCES Esenzione (codiceEsenzione) ON UPDATE CASCADE ON DELETE NO
ACTION,
PRIMARY KEY (codicePrestazione,codiceEsenzione));
```

## 7.7 Dipartimento

```
CREATE TABLE Dipartimento (
    nomeDipartimento varchar(5) NOT NULL PRIMARY KEY,
    descrizione varchar(30) NOT NULL);
```

## 7.8 CategoriaEsposizione

```
CREATE TABLE CategoriaEsposizione (
    codiceCategoria varchar(1) NOT NULL PRIMARY KEY,
    doseSoglia decimal(6,2) NOT NULL,
    descrizione varchar(50) NOT NULL);
```

## 7.9 Qualifica

```
CREATE TABLE Qualifica (
    codiceQualifica varchar(5) NOT NULL PRIMARY KEY,
    codiceCategoria varchar(1) NOT NULL, FOREIGN KEY (codiceCategoria)
    REFERENCES categoriaEsposizione (codiceCategoria) ON UPDATE CASCADE ON
    DELETE NO ACTION,
    descrizione varchar(30) NOT NULL);
```

## 7.10 Impiegato

```
CREATE TABLE Impiegato (
    matricola int NOT NULL PRIMARY KEY,
    nome varchar(15) NOT NULL,
    cognome varchar(15) NOT NULL,
    dataNascita date NOT NULL,
    cittaNascita varchar(15) NOT NULL,
```

```
codiceFiscale varchar(16) NOT NULL UNIQUE,  
indirizzo varchar(50) NOT NULL,  
cap int NOT NULL,  
telefono varchar(11) NOT NULL,  
codicequalifica varchar(5) NOT NULL, FOREIGN KEY (codicequalifica)  
REFERENCES Qualifica (codiceQualifica) ON UPDATE CASCADE ON  
DELETE NO ACTION,  
username varchar(50) NOT NULL UNIQUE);
```

## 7.11 Dosimetro

```
CREATE TABLE Dosimetro (  
idDosimetro int NOT NULL,  
matricola int NOT NULL, FOREIGN KEY (matricola) REFERENCES Impiegato  
(matricola) ON UPDATE CASCADE ON DELETE NO ACTION,  
doseRegistrata decimal(6,2) NOT NULL,  
inizioPeriodo date NOT NULL,  
finePeriodo date NOT NULL);
```

## 7.12 Sala

```
CREATE TABLE Sala (  
numeroStanza smallint NOT NULL PRIMARY KEY,  
metodoImaging varchar(5) NOT NULL,  
riservata varchar(15) NOT NULL);
```

## 7.13 Calendario

```
CREATE TABLE Calendario (
    data date NOT NULL PRIMARY KEY);
```

## 7.14 Orario

```
CREATE TABLE Orario (
    ora time NOT NULL PRIMARY KEY);
```

## 7.15 Richiesta

```
CREATE TABLE Richiesta (
    idRichiesta int NOT NULL AUTO_INCREMENT PRIMARY KEY,
    idPaziente int NOT NULL, FOREIGN KEY (idPaziente) REFERENCES Paziente
        (idPaziente) ON UPDATE CASCADE ON DELETE NO ACTION,
    quesitoDiagnostico varchar(50) NOT NULL,
    dataOraRichiesta datetime NOT NULL,
    codicePrestazione varchar(7) NOT NULL, FOREIGN KEY (codicePrestazione)
        REFERENCES Prestazione (codicePrestazione) ON UPDATE CASCADE ON
        DELETE NO ACTION,
    priorita int NOT NULL, CHECK (priorita between 0 and 3),
    nomeDipartimento varchar(5) NOT NULL, FOREIGN KEY (nomeDipartimento)
        REFERENCES Dipartimento (nomeDipartimento) ON UPDATE CASCADE ON
        DELETE NO ACTION,
    dataAppuntamento date NOT NULL, FOREIGN KEY (dataAppuntamento)
        REFERENCES Calendario (data) ON UPDATE CASCADE ON DELETE NO
        ACTION,
```

```
OraAppuntamento time NOT NULL, FOREIGN KEY (OraAppuntamento)
REFERENCES Orario (ora) ON UPDATE CASCADE ON DELETE NO
ACTION,
numeroStanza smallint NOT NULL, FOREIGN KEY (numeroStanza)
REFERENCES Sala (numeroStanza) ON UPDATE CASCADE ON DELETE NO
ACTION,
statoRichiesta set('Accettato','Prenotato','Cancellato') NOT NULL,
matricolaAmministrativo int, FOREIGN KEY (matricolaAmministrativo)
REFERENCES Impiegato (matricola) ON UPDATE CASCADE ON DELETE
NO ACTION);
```

## 7.16 Associata

```
CREATE TABLE Associata (
idRichiesta int NOT NULL UNIQUE, FOREIGN KEY (idRichiesta)
REFERENCES Richiesta (idRichiesta) ON UPDATE CASCADE ON DELETE
NO ACTION,
codiceEsenzione varchar(3) NOT NULL, FOREIGN KEY (codiceEsenzione)
REFERENCES Esenzione (codiceEsenzione) ON UPDATE CASCADE ON
DELETE NO ACTION,
PRIMARY KEY (idRichiesta, codiceEsenzione));
```

## 7.17 Esame

```
CREATE TABLE Esame (
idEsame int NOT NULL AUTO_INCREMENT PRIMARY KEY,
idRichiesta int NOT NULL, FOREIGN KEY (idRichiesta) REFERENCES
Richiesta (idRichiesta) ON UPDATE CASCADE ON DELETE NO ACTION,
dataOraEsame datetime NOT NULL,
matricolaTSRM int NOT NULL, FOREIGN KEY (matricolaTSRM)
REFERENCES Impiegato (matricola) ON UPDATE CASCADE ON DELETE
NO ACTION,
```

RADB  
statoEsame set('Annullato','Completato') NOT NULL,  
immaginiAccettate int NOT NULL DEFAULT 0,  
immaginiRifiutate int NOT NULL DEFAULT 0,  
dosemGycm decimal(6,2) NOT NULL DEFAULT 0.0,  
commento text);

## 7.18 Referta

CREATE TABLE Referta (  
matricolaMR int NOT NULL, FOREIGN KEY (matricolaMR) REFERENCES  
Impiegato (matricola) ON UPDATE CASCADE ON DELETE NO ACTION,  
idEsame int NOT NULL UNIQUE, FOREIGN KEY (idEsame) REFERENCES  
Esame (idEsame) ON UPDATE CASCADE ON DELETE NO ACTION,  
referto text,  
dataOraReferto datetime NOT NULL,  
PRIMARY KEY(matricolaMR,idEsame));

## **8 Inserimenti dati nel Database (Esempi)**

### **8.1 ComuneResidenza**

```
INSERT INTO comuneResidenza VALUES  
('Bari','Puglia'),  
('Bologna','EmiliaRomagna'),  
('Firenze','Toscana'),  
('Milano','Lombardia'),  
('Napoli','Campania'),  
('Palermo','Sicilia'),  
('Roma','Lazio'),  
('Torino','Piemonte');
```

### **8.2 Zona**

```
INSERT INTO zona VALUES  
(70121,'Bari'),  
(40121,'Bologna'),  
(50012,'Firenze'),  
(20121,'Milano'),  
(80121,'Napoli'),  
(90121,'Palermo'),  
(118,'Roma'),  
(10121,'Torino');
```

## 8.3 Paziente

INSERT INTO Paziente VALUES

- (1,'Flora','Calbo','F','1972-12-17','Torino','CLBFLR72T57L219F','Contrada Mondaini, 07,118,'1030699','Nessuna condizione nota'),
- (2,'Laura','Trombetta','F','2013-04-03','Napoli','TRMLRA13D43F839H','Piazza Corbo, 692','20121,'1084883','Nessuna condizione nota'),
- (3,'Ruggero','Tafuri','M','1959-04-05','Palermo','TFRRGR59D05G273N','Incrocio Pacillo, 337','80121,'1448903','Soggetto con storia pregressa di patologia X'),
- (4,'Luciana','Gianetti','F','1977-05-07','Roma','GNTLCN77E47H501P','Strada Mario, 9','70121,'1310717','Soggetto a mobilità ridotta'),
- (5,'Barbara','Einaudi','F','1998-05-11','Napoli','NDEBBR98E51F839L','Via Marisa, 1','20121,'1154252','Allergia a mdc gadolinio'),
- (6,'Marissa','Squarcione','F','1991-02-17','Milano','SQRMSS91B57F205P','Borgo Pertini, 012','118,'1888769','Nessuna condizione nota'),
- (7,'Maurizio','Bembo','M','2008-11-26','Bari','BMBMRZ08S26A662R','Strada Gaspare, 7,40121,'1072820','Nessuna condizione nota'),
- (8,'Barbara','Vidoni','F','2000-07-21','Torino','VDNBBR00L61L219W','Viale Tagliafierro, 34','80121,'1810772','Soggetto immunodepresso'),
- (9,'Stefania','Biagi','F','1972-06-04','Milano','BGISFN72H44F205L','Vicolo Balla, 60','80121,'1557666','Nessuna condizione nota'),
- (10,'Virgilio','Ricolfi','M','1980-12-29','Bari','RCLVGL80T29A662N','Strada Mario, 014','50012,'1252162','Nessuna condizione nota'),
- (11,'Marta','Cugia','F','2007-08-09','Bari','CGUMRT07M49A662N','Stretto Puccio, 156','70121,'1503628','Nessuna condizione nota'),
- (12,'Puccio','Govoni','M','1970-02-06','Napoli','GVNPCC70B06F839C','Contrada Bruno, 043','70121,'1419946','Soggetto sotto responsabilità tutore legale'),
- (13,'Serafina','Botticelli','F','1964-01-10','Torino','BTTSFN64A50L219L','Incrocio Boldù, 82','20121,'1551428','Allergia a mdc gadolinio'),
- (14,'Virginia','Peano','F','2012-09-23','Firenze','PNEVGN12P63D612M','Strada Carpaccio, 412','118,'1113468','Nessuna condizione nota'),
- (15,'Piersanti','Sinisi','M','1998-07-04','Milano','SNSPSN98L04F205M','Borgo Giacobbe, 855','70121,'1689200','Soggetto a mobilità ridotta'),

(16,'Giuliana','Modugno','F','1962-07-05','Bari','MDGGLN62L45A662T','Viale Calcedonio, 9',40121,'1365917','Soggetto a mobilità ridotta'),

(17,'Liana','Pedroni','F','1960-07-28','Palermo','PDRLNI60L68G273S','Viale Gianinazzi, 7',40121,'1298565','Soggetto categoria protetta'),

(18,'Lara','Pisacane','F','1974-11-09','Bari','PSCLRA74S49A662Z','Canale Emma, 802',40121,'1963772','Soggetto con storia pregressa di patologia X'),

(19,'Gianluca','Prati','M','1994-04-18','Bari','PRTGLC94D18A662U','Strada Gulotta, 637',40121,'1710293','Nessuna condizione nota'),

(20,'Anita','Speri','F','1994-10-18','Napoli','SPRNTA94R58F839L','Contrada Barillaro, 27',50012,'1245169','Nessuna condizione nota');

## 8.4 Prestazione

INSERT INTO Prestazione VALUES

('ECO01','ECO','Eco-color-doppler Addome Inferiore',0.00,36.15),

('ECO02','ECO','Eco-color-doppler Arti Superiori o inferiori',0.00,36.15),

('ECO03','ECO','Eco-color-doppler Fegato e Vie Biliari',0.00,36.15),

('ECO04','ECO','Eco-color-doppler Milza',0.00,36.15),

('ECO05','ECO','Eco-color-doppler Pancreas',0.00,36.15),

('RMN01','RM','Angio-RM Aorta Toracica e addominale',0.00,51.15),

('RMN02','RM','Angio-RM Aorta Toracica',0.00,51.15),

('RMN03','RM','Angio-RM Aorta Addominale',0.00,51.15),

('RMN04','RM','Angio-RM Arti Inferiori',0.00,51.15),

('RMN05','RM','Angio-RM Vasi Epiortici',0.00,51.15),

('RX01','RX','RX Addome',10.00,19.37),

('RX02','RX','RX Addome ped',1.00,19.37),

('RX03DX','RX','RX Avambraccio Destro',2.00,14.98),

('RX03SN','RX','RX Avambraccio Sinistro',2.00,14.98),

('RX04DX','RX','RX Braccio Destro',2.00,14.98),

RADB

('RX04SN','RX','RX Braccio Sinistro',2.00,14.98),  
('TC01','TC','Angio-TC Aorta Toracica',650.00,51.15),  
('TC02','TC','Angio-TC Aorta Addominale',800.00,51.15),  
('TC03','TC','Angio-TC Arterie Polmonari',650.00,51.15),  
('TC04','TC','Angio-TC Arti Inferiori',650.00,51.15),  
('TC05','TC','Angio-TC Vasi Epiaortici',800.00,51.15);

## 8.5 Esenzione

INSERT INTO Esenzione VALUES

('001','Acromegalia e Gigantismo','Nazionale'),  
('003','Anemia Emolitica Acquisita Autoimmune','Nazionale'),  
('0031','Ipertensione Arteriosa(con danno d'organo)','Nazionale'),  
('005','Anoressia Nervosa, Bulimia','Nazionale'),  
(C01,'Invalidità Civile','Nazionale'),  
(C02,'Invalidità Civile','Nazionale'),  
(C03,'Invalidità Civile','Nazionale'),  
(C04,'Invalidità Civile','Nazionale'),  
(C05,'Invalidità Civile','Nazionale'),  
(C06,'Invalidità Civile','Nazionale'),  
(D01,'Screening','Nazionale'),  
(D02,'Screening','Nazionale'),  
(D03,'Screening','Nazionale'),  
(D04,'Screening','Nazionale'),  
(D05,'Screening','Nazionale'),  
(E01,'Reddito','Nazionale'),  
(E02,'Reddito','Nazionale'),

('E03','Reddito','Nazionale'),  
('E04','Reddito','Nazionale'),  
('E05','Reddito','Regionale'),  
('E11','Reddito','Regionale'),  
('E12','Reddito','Regionale'),  
('E13','Reddito','Regionale'),  
('E15','Reddito','Regionale');

## 8.6 PacchettoPrestazionale

INSERT INTO PacchettoPrestazionale VALUES

('TC16','001'),  
('ECO09','003'),  
('ECO08','0031'),  
('ECO09','0031'),  
('RX29','0031'),  
('ECO09','006'),  
('RX03DX','006'),  
('RX03SN','006'),  
('RX04DX','006'),  
('RX04SN','006'),  
('RX05DX','006'),  
('RX05SN','006'),  
('RX07DX','006'),  
('RX07SN','006'),  
('RX08DX','006'),  
('RX08SN','006'),

RADB  
('RX09DX','006'),  
('RX09SN','006'),  
('RX10DX','006'),  
('RX10SN','006'),  
('RX19','006'),  
('RX21DX','006');

## 8.7 Dipartimento

INSERT INTO Dipartimento VALUES  
('MC','Esterno'),  
('PS','Pronto Soccorso'),  
('TIN','Terapia Intensiva Neonatale'),  
('UTI','Terapia Intensiva'),  
('UTIC','Terapia Intensiva Cardiologica'),  
('W1','Ortopedia1'),  
('W10','Cardiologia'),  
('W11','Ematologia'),  
('W15','Chirurgia1'),  
('W16','Chirurgia2'),  
('W17','Nefrologia'),  
('W18','Gastroenterologia'),  
('W19','Reparto Grandi Ustioni'),  
('W2','Ortopedia2'),  
('W3','Pediatria'),  
('W4','Urologia'),  
('W5','Geriatrica'),

('W6','Medicina d Urgenza 1'),

('W7','Medicina d Urgenza 2'),

('W8','Neurologia'),

('W9','Malattie Infettive');

## 8.8 CategoriaEsposizione

INSERT INTO CategoriaEsposizione VALUES

('A',20.00,'Supera 6 mSv dose efficace in un anno solare'),

('B',6.00,'Non supera 6 mSv dose efficace in un anno solare'),

('c',1.00,'Supera 1 mSv dose efficace in un anno solare');

## 8.9 Qualifica

INSERT INTO Qualifica VALUES ('AMM','C','Amministrativo'),

('AS','C','Amministratore Sistema'),

('MR','B','Medico Radiologo'),

('TSRM','A','Tecnico di Radiologia');

## 8.10 Impiegato

INSERT INTO Impiegato VALUES

(1,'Pier','Morlacchi','1979-05-19','Palermo','MRLPRI79E19G273W','Vicolo Parini, 384',118,'1889421','AS','MorlacchiPier@localhost'),

(2,'Cassandra','Cabibbo','1958-09-26','Bari','CBBCSN58P66A662A','Canale Calcedonio, 35',118,'1639998','AS','CabibboCassandra@localhost'),

(3,'Anita','Disdero','1985-04-10','Bari','DSDNTA85D50A662S','Strada Bragadin, 45',118,'1705103','MR','DisderoAnita@localhost'),

(4,'Claudio','Pistoletto','1992-04-27','Milano','PSTCLD92D27F205M','Vicolo Sanguineti, 5',118,'1587783','MR','PistolettoClaudio@localhost'),

(5,'Gioacchino','Sommaruga','1965-05-07','Firenze','SMMGCH65E07D612A','Incroci Coriolano, 86',118,'1978553','MR','SommarugaGioacchino@localhost'),

## RADB

- (6,'Marcella','Respighi','1988-03-02','Firenze','RSPMCL88C42D612S','Piazza Salvo, 769',118,'1208036','MR','RespighiMarcella@localhost'),
- (7,'Fiorino','Muratori','1968-09-27','Bologna','MRTFRN68P27A944Y','Piazza Rossi, 68',118,'1545823','MR','MuratoriFiorino@localhost'),
- (8,'Piersanti','Mercalli','1983-12-01','Palermo','MRCPSN83T01G273L','Piazza Niscoromni, 81',118,'1767121','MR','MercalliPiersanti@localhost'),
- (9,'Ruggiero','Corbo','1963-10-19','Napoli','CRBRGR63R19F839L','Piazza Dante, 863',118,'1644539','MR','CorboRuggiero@localhost'),
- (10,'Severino','Pedrazzini','1992-09-28','Firenze','PDRSRN92P28D612U','Rotonda Gozzi, 6',118,'1082590','MR','PedrazziniSeverino@localhost'),
- (11,'Eva','Federico','1971-01-14','Torino','FDRVEA71A54L219V','Borgo Eva, 04',118,'1060500','TSRM','FedericoEva@localhost'),
- (12,'Virginia','Chinnici','1965-08-23','Palermo','CHNVGN65M63G273B','Contrada Ruggiero, 266',118,'1640829','TSRM','ChinniciVirginia@localhost'),
- (13,'Lauretta','Casalodi','1978-09-23','Bari','CSLLTT78P63A662I','Contrada Trapanese, 6',118,'1895188','TSRM','CasalodiLauretta@localhost'),
- (14,'Maria','Pincherle','1996-04-24','Milano','PNCMRA96D64F205J','Piazza Raffaele, 8',118,'1490397','TSRM','PincherleMaria@localhost'),
- (15,'Mariana','Colletti','1958-05-27','Roma','CLLMRN58E67H501W','Via Baresi, 6',118,'1062060','TSRM','CollettiMariana@localhost'),
- (16,'Virginia','Caracciolo','1970-02-23','Bologna','CRCVGN70B63A944K','Vicolo Pedrazzini, 60',118,'1824979','TSRM','CaraccioloVirginia@localhost'),
- (17,'Natalia','Marino','1979-11-18','Firenze','MRNNNTL79S58D612X','Vicolo Silvia, 944',118,'1656114','TSRM','MarinoNatalia@localhost'),
- (18,'Antonietta','Camanni','1972-06-01','Torino','CMNNNT72H41L219L','Strada Panfilo, 798',118,'1510006','TSRM','CamanniAntonietta@localhost'),
- (19,'Marta','Antonelli','1958-05-09','Napoli','NTNMRT58E49F839T','Incrocio Eleanora, 772',118,'1226238','TSRM','AntonelliMarta@localhost'),
- (20,'Morena','Ossola','1987-12-25','Milano','SSLMRN87T65F205T','Incrocio Stefani, 4',118,'1001376','TSRM','OssolaMorena@localhost'),
- (21,'Aria','Pavone','1997-04-16','Roma','PVNRAI97D56H501P','Stretto Righi, 683',118,'1388287','TSRM','PavoneAria@localhost'),
- (22,'Orazio','Morucci','1976-05-22','Bari','MRCRZO76E22A662D','Canale Eleanora, 91',118,'1791349','TSRM','MorucciOrazio@localhost'),

(23,'Caterina','Ceravolo','1969-01-31','Roma','CRVCRN69A71H501D','Contrada Piergiuseppe, 28',118,'1433450','TSRM','CeravoloCaterina@localhost'),

(24,'Benvenuto','Troisi','1965-10-02','Roma','TRSBVN65R02H501Q','Contrada Ciani, 4',118,'1186242','TSRM','TroisiBenvenuto@localhost'),

(25,'Emilio','Sokolov','1997-05-09','Milano','SKLMLE97E09H501U','Vicolo Peano, 24',118,'1322534','AMM','SokolovEmilio@localhost'),

(26,'Federica','Porzio','1976-06-26','Roma','PRZFRC76H66H501J','Incrocio Antonelli, 3',118,'1085222','AMM','PorzioFederica@localhost'),

(27,'Atenulf','Mortati','1978-02-16','Torino','MRTTLF78B16H501I','Piazza Toselli, 4',118,'1527621','AMM','MortatiAtenulf@localhost'),

(28,'Morena','Salieri','1973-01-23','Napoli','SLRMRN73A63F839Q','Contrada Strangio, 752',118,'1708702','AMM','SalieriMorena@localhost'),

(29,'Giustino','Solimena','1980-09-07','Bari','SLMGTN80P07F839F','Rotonda Cugia, 261',118,'1599091','AMM','SolimenaGiustino@localhost'),

(30,'Bernardo','Bruno','1955-10-13','Bologna','BRNBNR55R13A662F','Stretto Amato, 88',118,'1191986','AMM','BrunoBernardo@localhost');

## 8.11 Dosimetro

```
INSERT INTO Dosimetro VALUES
(20191,1,0.82,'2019-01-01','2019-01-31'),
(20192,2,0.35,'2019-01-01','2019-01-31'),
(20193,3,0.72,'2019-01-01','2019-01-31'),
(20194,4,0.38,'2019-01-01','2019-01-31'),
(20195,5,0.80,'2019-01-01','2019-01-31'),
(20196,6,0.81,'2019-01-01','2019-01-31'),
(20197,7,0.85,'2019-01-01','2019-01-31'),
(20198,8,0.81,'2019-01-01','2019-01-31'),
(20199,9,0.85,'2019-01-01','2019-01-31'),
(20201,1,0.56,'2020-01-01','2020-01-31'),
(20202,2,0.48,'2020-01-01','2020-01-31'),
```

RADB

```
(20203,3,0.16,'2020-01-01','2020-01-31'),  
(20204,4,0.33,'2020-01-01','2020-01-31'),  
(20205,5,0.77,'2020-01-01','2020-01-31'),  
(20206,6,0.15,'2020-01-01','2020-01-31'),  
(20207,7,0.75,'2020-01-01','2020-01-31'),  
(20208,8,0.18,'2020-01-01','2020-01-31'),  
(20209,9,0.10,'2020-01-01','2020-01-31'),  
(20211,1,0.87,'2021-01-01','2021-01-31'),  
(20212,2,0.25,'2021-01-01','2021-01-31');
```

## 8.12 Sala

INSERT INTO Sala VALUES

```
(1,'RX','Ricov/PS'),  
(2,'RX','Esterni'),  
(3,'ECO','Ricov/PS'),  
(4,'ECO','Esterni'),  
(5,'RM','Ricov/PS'),  
(6,'RM','Esterni'),  
(7,'TC','Ricov/PS'),  
(8,'TC','Esterni'),  
(9,'OPG','Esterni'),  
(10,'MG','Esterni');
```

## 8.13 Calendario

```
INSERT INTO Calendario VALUES
```

```
('2021-07-01'),
```

```
('2021-07-02'),
```

```
('2021-07-03'),
```

```
('2021-07-04'),
```

```
('2021-07-05'),
```

```
('2021-07-06'),
```

```
('2021-07-07'),
```

```
('2021-07-08'),
```

```
('2021-07-09'),
```

```
('2021-07-10'),
```

```
('2021-07-11'),
```

```
('2021-07-12'),
```

```
('2021-07-13'),
```

```
('2021-07-14'),
```

```
('2021-07-15'),
```

```
('2021-07-16'),
```

```
('2021-07-17'),
```

```
('2021-07-18'),
```

```
('2021-07-19'),
```

```
('2021-07-20');
```

## 8.14 Orario

```
INSERT INTO Orario VALUES
```

```
('00:00:00'),
```

```
('00:30:00'),
```

```
('01:00:00'),
```

```
('01:30:00'),
```

```
('02:00:00'),
```

```
('02:30:00'),
```

```
('03:00:00'),
```

```
('03:30:00'),
```

```
('04:00:00'),
```

```
('04:30:00'),
```

```
('05:00:00'),
```

```
('05:30:00'),
```

```
('06:00:00'),
```

```
('06:30:00'),
```

```
('07:00:00'),
```

```
('07:30:00'),
```

```
('08:00:00'),
```

```
('08:30:00'),
```

```
('09:00:00'),
```

```
('09:30:00'),
```

```
('10:00:00');
```

## 8.15 Richiesta

INSERT INTO Richiesta VALUES

(1,1,'Insufficienza Renale','2021-06-30 23:09:00','ECO09',2,'PS','2021-07-01','00:00:00',3,'Accettato',25),

(2,2,'Connettiviti Indifferenziate','2021-05-02 08:00:00','ECO09',1,'MC','2021-07-01','08:00:00',4,'Cancellato',28),

(3,3,'Epatite Cronica','2021-06-24 00:30:00','ECO09',1,'W15','2021-07-01','00:30:00',3,'Accettato',28),

(4,4,'Sclerosi multipla','2021-06-28 00:00:00','RMN17',2,'W11','2021-07-01','00:00:00',5,'Prenotato',26),

(5,2,'Valutazione mezzo di sintesi','2021-06-28 00:00:00','RX02',2,'W17','2021-07-01','00:00:00',1,'Accettato',26),

(6,2,'Colite ulcerosa','2021-06-24 01:00:00','ECO10',1,'W19','2021-07-01','01:00:00',3,'Cancellato',27),

(7,3,'Epilessia','2021-06-24 00:30:00','RMN17',1,'W16','2021-07-01','00:30:00',5,'Prenotato',28),

(8,5,'Anterolistesi','2021-06-24 01:00:00','RMN11',1,'W5','2021-07-01','01:00:00',5,'Cancellato',27),

(9,4,'Connettiviti Indifferenziate','2021-06-30 00:30:00','RX08SN',3,'W7','2021-07-01','00:30:00',1,'Accettato',25),

(10,6,'Screening Tumore Mammella','2020-07-02 08:00:00','MG12DX',0,'MC','2021-07-01','08:00:00',10,'Accettato',30),

(11,5,'Aneurisma Aorta','2021-06-24 00:00:00','TC01',1,'W2','2021-07-01','00:00:00',7,'Accettato',25),

(12,4,'Valutazione mezzo di sintesi','2021-06-21 08:00:00','OPG16',2,'MC','2021-07-01','08:00:00',9,'Accettato',27),

(13,4,'Valutazione mezzo di sintesi','2021-06-21 09:00:00','OPG15',2,'MC','2021-07-01','09:00:00',9,'Accettato',27),

(14,6,'Insufficienza Renale','2021-06-24 01:30:00','ECO06',1,'W19','2021-07-01','01:30:00',3,'Accettato',30),

## RADB

(15,3,'Artrite Reumatoide','2021-06-30 01:00:00','RX19',3,'W18','2021-07-01','01:00:00',1,'Accettato',25),

(16,3,'Artrite Reumatoide','2021-06-30 01:30:00','RX23',3,'W18','2021-07-01','01:30:00',1,'Accettato',25),

(17,7,'Aneurisma Cerebrale','2021-06-21 08:00:00','TC06',2,'MC','2021-07-01','08:00:00',8,'Cancellato',28),

(18,1,'Trauma','2021-06-21 09:30:00','OPG16',2,'MC','2021-07-01','09:30:00',9,'Cancellato',26),

(19,1,'Trauma','2021-06-21 10:00:00','OPG15',2,'MC','2021-07-01','10:00:00',9,'Cancellato',26),

(20,1,'Cefalea','2021-06-30 01:30:00','RMN11',3,'W5','2021-07-01','01:30:00',5,'Cancellato',25);

## 8.16 Associata

INSERT INTO Associata VALUES

(2,'C02'),

(10,'E03'),

(12,'E01'),

(13,'E01'),

(17,'C03'),

(18,'C06'),

(19,'C06'),

(21,'E05'),

(25,'L04'),

(26,'E01'),

(27,'E01'),

(28,'V01'),

(29,'V01'),

(30,'E01'),  
(34,'E03'),  
(35,'L03'),  
(36,'C03'),  
(45,'067'),  
(46,'067'),  
(51,'S01');

## 8.17 Esame

INSERT INTO Esame VALUES

(1,1,'2021-07-01 01:51:00',12,'Completato',4,0,0.00,'Uso della Griglia, peso Pz > 100 kg '),  
(2,3,'2021-07-01 02:25:00',11,'Completato',4,0,0.00,'Reazione al mezzo di contrasto, nega allergia'),  
(3,5,'2021-07-01 02:59:00',13,'Completato',2,0,1.10,'Paziente in stato di incoscienza'),  
(4,9,'2021-07-01 02:41:00',21,'Completato',3,2,5.10,'Paziente non deambulante'),  
(5,10,'2021-07-01 10:47:00',22,'Completato',3,0,6.10,'Esame eseguito in condizioni ottimali'),  
(6,11,'2021-07-01 01:59:00',11,'Annullato',0,0,0.00,'Condizioni paziente non idonee all esecuzione dell esame'),  
(7,12,'2021-07-01 10:56:00',11,'Completato',1,0,5.46,'Paziente affetto da difficoltà dell apprendimento'),  
(8,13,'2021-07-01 11:43:00',11,'Annullato',0,0,0.00,'Discussione del caso. Inappropriatezza della richiesta'),  
(9,14,'2021-07-01 03:00:00',15,'Annullato',0,0,0.00,'Valori analisi incompatibili con somministrazione mdc'),  
(10,15,'2021-07-01 01:51:00',17,'Completato',2,1,257.35,'Paziente non collaborante'),  
(11,16,'2021-07-01 02:11:00',17,'Completato',3,0,33.23,'Esame eseguito in condizioni ottimali');

## RADB

(12,22,'2021-07-01 02:45:00',24,'Annullato',0,0,0.00,'Discussione del caso. Inappropriatezza della richiesta'),  
(13,23,'2021-07-01 03:34:00',24,'Completato',2,3,10.17,'Esame eseguito in condizioni ottimali'),  
(14,24,'2021-07-01 03:42:00',16,'Completato',4,3,1.21,'Paziente presenta gonfiore nel distretto interessato'),  
(15,25,'2021-07-01 10:14:00',12,'Annullato',0,0,0.00,'Condizioni paziente non idonee all'esecuzione dell'esame'),  
(16,32,'2021-07-01 02:58:00',23,'Completato',2592,0,1433.32,'Paziente non collaborante'),  
(17,34,'2021-07-01 10:23:00',15,'Completato',1507,0,0.00,'Paziente presenta gonfiore nel distretto interessato'),  
(18,36,'2021-07-01 11:42:00',21,'Completato',1872,0,1280.06,'Uso della Griglia, peso Pz > 100 kg ' ),  
(19,41,'2021-07-01 03:37:00',23,'Completato',3,0,0.00,'Reazione al mezzo di contrasto, nega allergia'),  
(20,42,'2021-07-01 05:23:00',11,'Completato',3,0,0.00,'Paziente non collaborante');

## 8.18 Referta

INSERT INTO Referta VALUES

(3,5,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-15 10:47:00'),  
(3,18,'Assenza di segni riferibili come causa della patologia','2021-07-04 11:42:00'),  
(3,21,'Non Apprezzabili segni riferibili alla patologia','2021-07-05 04:44:00'),  
(3,26,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-01 04:55:00'),  
(3,36,'Non Apprezzabili segni riferibili alla patologia','2021-07-04 11:56:00'),  
(3,77,'Assenza di segni riferibili come causa della patologia','2021-07-05 14:30:00'),  
(3,105,'Presenza di fratture e versamento diffuso','2021-07-09 14:53:00'),  
(3,122,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-03 10:05:00'),

(3,135,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-11 17:51:00'),

(3,183,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-02 13:26:00'),

(3,184,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-02 13:41:00'),

(3,188,'Presenza di anterolistesi di moderate dimensioni. Si consigliano ulteriori approfondimenti diagnostici','2021-07-09 11:34:00'),

(3,282,'Impianto in sede. Rapporti osteoarticolari mantenuti','2021-07-12 14:56:00'),

(3,316,'Assenza di fratture e/o versamenti','2021-07-19 12:18:00'),

(3,335,'Non Apprezzabili segni riferibili alla patologia','2021-07-04 01:52:00'),

(3,338,'Impianto in sede. Rapporti osteoarticolari mantenuti','2021-07-19 14:04:00'),

(3,367,'Presenza di fratture e versamento circoscritto','2021-07-20 14:35:00'),

(3,400,'Presenza di segni riferibili come causa patologia. Si consiglia visione da parte di uno specialista','2021-07-21 10:58:00'),(3,401,'Presenza di segni riferibili come causa patologia. Si consiglia visione da parte di uno specialista','2021-07-21 11:19:00'),

(3,424,'Non Apprezzabili segni riferibili alla patologia','2021-07-10 14:33:00'),

(3,456,'Non Apprezzabili segni riferibili alla patologia','2021-07-17 19:46:00'),

(3,459,'Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista','2021-07-03 16:26:00');

## 9 Funzione Festivo

A seguire una funzione, creata per semplificare la scrittura delle query, che ritorna 1 se il giorno passato in input è un festivo 0 altrimenti.

```
mysql> delimiter //
mysql> CREATE FUNCTION Festivo (data date)
-> RETURNS INTEGER
-> DETERMINISTIC
-> BEGIN
-> IF (date_format(data, '%m-%d') = '01-01'
-> or date_format(data, '%m-%d') = '12-25'
-> or date_format(data, '%m-%d') = '01-06'
-> or date_format(data, '%m-%d') = '04-25'
-> or date_format(data, '%m-%d') = '05-01'
-> or date_format(data, '%m-%d') = '06-02'
-> or date_format(data, '%m-%d') = '08-15'
-> or date_format(data, '%m-%d') = '11-01'
-> or date_format(data, '%m-%d') = '12-08'
-> or weekday(data) = 6) then
-> RETURN 1;
-> ELSE
-> RETURN 0;
-> END IF;
-> END//
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> delimiter ;
```

## 10 Trigger ed Event

A seguire verranno presentati un Trigger e un Event pensati per garantire la coerenza dei dati nel database.

### 10.1 Trigger Esenzione non Applicabile

Poiché alcuni tipi di esenzione hanno validità solo regionale e i pazienti ricoverati o provenienti dal Pronto Soccorso non sono associati ad alcuna esenzione, il seguente trigger ha lo scopo di sollevare un errore qualora si inserisca una esenzione, ma il paziente risieda fuori dalla regione Lazio o sia un ricoverato o proveniente dal PS. L'esempio di funzionamento è rimandato al paragrafo Sicurezza dove verrà mostrato in combinazione con la chiamata di alcune procedure da parte di un user.

```

mysql> delimiter //
mysql> CREATE TRIGGER Esenzione_Non_Applicabile BEFORE INSERT ON Associata FOR EACH ROW
-> BEGIN
-> DECLARE reg varchar(50);
-> DECLARE tipoEsen varchar(15);
-> DECLARE nomeDip varchar(15);
-> SELECT territorio
-> FROM Esenzione
-> WHERE codiceEsenzione = new.codiceEsenzione into tipoEsen;
-> SELECT ComuneResidenza.regione, Richiesta.nomeDipartimento into reg, nomeDip
-> FROM Richiesta, ComuneResidenza, Zona, Paziente
-> WHERE Richiesta.idRichiesta = new.idRichiesta
-> AND Paziente.idPaziente = Richiesta.idPaziente
-> AND Paziente.cap = Zona.cap
-> AND Zona.comune = comuneResidenza.comune;
-> if (reg != 'Lazio' and tipoEsen = 'Regionale' and nomeDip = 'MC') then
-> signal sqlstate '45000' set message_text = 'Paziente residente fuori regione, esenzione non Applicabile';
-> elseif (nomeDip != 'MC') then
-> signal sqlstate '45000' set message_text = 'Paziente Ricoverato, esenzione non applicabile';
-> end if;
-> END;//

```

Query OK, 0 rows affected (0.02 sec)

```
mysql> delimiter ;
```

## 10.2 Event Apri Agenda

L’event apriAgenda serve ad inserire automaticamente ogni giorno una nuova data per le prenotazioni. L’esempio mostra la stessa query prima e dopo la creazione dell’event.

```

mysql> SELECT * FROM Calendario ORDER BY data desc LIMIT 10;
+-----+
| data    |
+-----+
| 2021-08-31 |
| 2021-08-30 |
| 2021-08-29 |

```

## RADB

```
| 2021-08-28 |
| 2021-08-27 |
| 2021-08-26 |
| 2021-08-25 |
| 2021-08-24 |
| 2021-08-23 |
| 2021-08-22 |
+-----+
10 rows in set (0.00 sec)

mysql> CREATE EVENT AggiornaCalendario
      -> ON SCHEDULE EVERY 1 DAY
      -> DO
      -> INSERT into Calendario values (date_add(curdate(), INTERVAL 1 year));

Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM Calendario ORDER BY data desc LIMIT 10;

+-----+
| data      |
+-----+
| 2022-08-21 |
| 2021-08-31 |
| 2021-08-30 |
| 2021-08-29 |
| 2021-08-28 |
| 2021-08-27 |
| 2021-08-26 |
| 2021-08-25 |
| 2021-08-24 |
| 2021-08-23 |
+-----+
10 rows in set (0.00 sec)
```

# 11 View

A seguire saranno presentate le View d'esempio create. Gli esempi di funzionamento di alcune verranno trattati successivamente, nell'ambito della Sicurezza in combinazione con i tipi di utente per cui sono state pensate.

## 11.1 View Planning Appuntamenti

Uno dei compiti degli amministrativi è di prenotare i pazienti al loro arrivo in reparto. La seguente View è pensata per gli user amministrativi e serve a visualizzare tutti gli appuntamenti futuri fissati e non.

La prima View è dedicata agli appuntamenti esterni:

```
mysql> CREATE VIEW PlanningAppuntamentiEsterni AS
-> SELECT idRichiesta, statoRichiesta, data, ora, Sala.numeroStanza
-> FROM Calendario INNER JOIN Orario INNER JOIN Sala
-> LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza, statoRichiesta FROM Richiesta
-> WHERE statoRichiesta != 'Cancellato') as agenda
-> ON agenda.dataAppuntamento = Calendario.data
-> AND agenda.oraAppuntamento = Orario.ora
-> AND agenda.numeroStanza = Sala.numeroStanza
-> WHERE Sala.Riservata = 'Esterni'
-> AND timestamp(data, ora) >= current_timestamp()
-> AND ora >= '8:00'
-> AND ora <= '19:00'
-> AND Festivo(data) = 0;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM PlanningAppuntamentiEsterni ORDER BY data, ora limit 20;
+-----+
| idRichiesta | statoRichiesta | data       | ora        | numeroStanza |
+-----+
|      NULL    |      NULL       | 2021-08-21 | 09:00:00  |      8        |
|     11565    | Prenotato      | 2021-08-21 | 09:00:00  |      4        |
|      NULL    |      NULL       | 2021-08-21 | 09:00:00  |     10        |
|     8049     | Prenotato      | 2021-08-21 | 09:00:00  |      9        |
|      NULL    |      NULL       | 2021-08-21 | 09:00:00  |      6        |
|      NULL    |      NULL       | 2021-08-21 | 09:00:00  |      2        |
|      NULL    |      NULL       | 2021-08-21 | 09:30:00  |      6        |
|     11568    | Prenotato      | 2021-08-21 | 09:30:00  |      4        |
|     11828    | Prenotato      | 2021-08-21 | 09:30:00  |      8        |
|      NULL    |      NULL       | 2021-08-21 | 09:30:00  |     10        |
|      NULL    |      NULL       | 2021-08-21 | 09:30:00  |      9        |
|     6654     | Prenotato      | 2021-08-21 | 09:30:00  |      2        |
+-----+
```

## RADB

```
|     NULL | NULL      | 2021-08-21 | 10:00:00 |      10 |
|     NULL | NULL      | 2021-08-21 | 10:00:00 |       9 |
| 10662 | Prenotato | 2021-08-21 | 10:00:00 |       6 |
|     NULL | NULL      | 2021-08-21 | 10:00:00 |       2 |
| 11569 | Prenotato | 2021-08-21 | 10:00:00 |       4 |
| 11873 | Prenotato | 2021-08-21 | 10:00:00 |       8 |
|     NULL | NULL      | 2021-08-21 | 10:30:00 |       4 |
|     NULL | NULL      | 2021-08-21 | 10:30:00 |       2 |
+-----+
20 rows in set (0.39 sec)
```

La seconda View è dedicata agli appuntamenti interni:

```
mysql> CREATE VIEW PlanningAppuntamentiInterni AS
-> SELECT idRichiesta, statoRichiesta, data, ora, Sala.numeroStanza
-> FROM Calendario INNER JOIN Orario INNER JOIN Sala
-> LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza,
-> statoRichiesta
-> FROM Richiesta
-> WHERE statoRichiesta != 'Cancellato') as agenda
-> ON agenda.dataAppuntamento = Calendario.data
-> AND agenda.oraAppuntamento = Orario.ora
-> AND agenda.numeroStanza = Sala.numeroStanza
-> WHERE Sala.Riservata = 'Ricov/PS'
-> AND timestamp(data, ora) >= current_timestamp();
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> SELECT * FROM PlanningAppuntamentiInterni ORDER BY data, ora limit 20;
+-----+
| idRichiesta | statoRichiesta | data      | ora      | numeroStanza |
+-----+
|     NULL | NULL      | 2021-08-21 | 09:00:00 |       1 |
|     NULL | NULL      | 2021-08-21 | 09:00:00 |       5 |
| 17234 | Prenotato | 2021-08-21 | 09:00:00 |       7 |
|     NULL | NULL      | 2021-08-21 | 09:00:00 |       3 |
| 12446 | Prenotato | 2021-08-21 | 09:30:00 |       1 |
| 17274 | Prenotato | 2021-08-21 | 09:30:00 |       3 |
| 16151 | Prenotato | 2021-08-21 | 09:30:00 |       5 |
|     NULL | NULL      | 2021-08-21 | 09:30:00 |       7 |
| 17237 | Prenotato | 2021-08-21 | 10:00:00 |       7 |
|     NULL | NULL      | 2021-08-21 | 10:00:00 |       3 |
| 16152 | Prenotato | 2021-08-21 | 10:00:00 |       5 |
|     NULL | NULL      | 2021-08-21 | 10:00:00 |       1 |
|     NULL | NULL      | 2021-08-21 | 10:30:00 |       5 |
|     NULL | NULL      | 2021-08-21 | 10:30:00 |       7 |
+-----+
```

```

|      17277 | Prenotato    | 2021-08-21 | 10:30:00 |          3 |
|      NULL  | NULL          | 2021-08-21 | 10:30:00 |          1 |
|      17240 | Prenotato    | 2021-08-21 | 11:00:00 |          7 |
|      NULL  | NULL          | 2021-08-21 | 11:00:00 |          5 |
|      12467 | Prenotato    | 2021-08-21 | 11:00:00 |          1 |
|      17280 | Prenotato    | 2021-08-21 | 11:00:00 |          3 |
+-----+

```

20 rows in set (0.58 sec)

## 11.2 View worklist per Amministrativi

Uno dei compiti degli amministrativi è accettare i pazienti al loro arrivo in reparto. La seguente View è pensata per gli user amministrativi e serve a visualizzare i pazienti con appuntamento fissato per quel giorno e che devono essere accettati.

```

mysql> CREATE VIEW worklistAmministrativi AS
-> SELECT Richiesta.idPaziente, Paziente.nome, Paziente.cognome, Paziente.codiceFiscale,
-> Richiesta.idRichiesta, Richiesta.numeroStanza as stanza, Richiesta.dataAppuntamento as data,
-> Richiesta.oraAppuntamento as ora, Prestazione.nomePrestazione as prestazione
-> FROM Paziente, Richiesta, Prestazione
-> WHERE statoRichiesta = 'Prenotato'
-> AND Richiesta.dataAppuntamento = curdate()
-> AND Richiesta.idPaziente = Paziente.idPaziente
-> AND Prestazione.codicePrestazione = Richiesta.codicePrestazione
-> ORDER BY Richiesta.oraAppuntamento, Richiesta.numeroStanza;
Query OK, 0 rows affected (0.01 sec)

```

mysql> SELECT \* FROM worklistAmministrativi LIMIT 10;

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| idPaziente | nome      | cognome   | codiceFiscale | idRichiesta | stanza | data       | ora        | prestazione |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|      72 | Diana     | Monduzzi  | MNNDONI66T41F839A | 16105 | 5 | 2021-08-21 | 01:00:00 | Angio-RM Vasi Intracrani |
|      73 | Agnolo    | Barsanti   | BRSGNL80R17A944P | 17220 | 3 | 2021-08-21 | 01:30:00 | Ecografia Addome Superiore |
|     112 | Michelangelo | Proietti  | PRTHML91P22M501W | 16112 | 5 | 2021-08-21 | 03:00:00 | RM Encefalo |
|      57 | Amico     | Maspero   | NSPMCA61M08M501Y | 12410 | 1 | 2021-08-21 | 04:00:00 | RX Polso Sinistro |
|      57 | Amico     | Maspero   | NSPMCA61M08M501Y | 12411 | 1 | 2021-08-21 | 04:30:00 | RX Mano Sinistra |
|     147 | Lamberto  | Bonomo    | BMMLBR7IS17Dc12B | 17244 | 3 | 2021-08-21 | 04:30:00 | Ecografia Pelvica |
|      57 | Amico     | Maspero   | NSPMCA61M08M501Y | 12412 | 1 | 2021-08-21 | 05:00:00 | RX Polso Destro |
|      27 | Imelda    | Tamburello | TMBMLD81R43F839C | 16130 | 5 | 2021-08-21 | 05:00:00 | RM Addome Superiore |
|     122 | Livia     | Mancini   | MNCLVII9BE65A944A | 17246 | 3 | 2021-08-21 | 05:30:00 | Ecografia Linfonodi |
|      27 | Imelda    | Tamburello | TMBMLD81R43F839C | 16131 | 5 | 2021-08-21 | 05:30:00 | ColAngio-RM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

10 rows in set (0.00 sec)

## 11.3 View worklist per Tecnici di Radiologia

Il compito dei Tecnici di Radiologia è quello di eseguire gli esami. La seguente View serve a mostrare i pazienti arrivati in reparto e che hanno già eseguito l'accettazione e attendono per l'esame.

```
mysql> CREATE VIEW worklistTSRM AS
-> SELECT Richiesta.idPaziente, Paziente.nome, Paziente.cognome, Paziente.codiceFiscale, Paziente.note,
-> Richiesta.idRichiesta, Richiesta.oraAppuntamento, Richiesta.numeroStanza, Prestazione.nomePrestazione
-> FROM Paziente inner join Richiesta on Richiesta.idPaziente = Paziente.idPaziente
-> inner join Prestazione on Richiesta.codicePrestazione = Prestazione.codicePrestazione
-> LEFT JOIN Esame on Esame.idRichiesta = Richiesta.idRichiesta
-> WHERE statoRichiesta = 'Accettato'
-> AND Richiesta.dataAppuntamento = curdate()
-> AND Esame.idEsame is null;
Query OK, 0 rows affected (0.01 sec)
```

## 11.4 View Worklist Medici di Radiologia

Dopo il completamento degli esami i medici radiologi devono eseguirne la refertazione. La seguente View è pensata quindi per mostrare ai medici gli esami in attesa di referto.

```
mysql> CREATE VIEW worklistMR AS
-> SELECT Richiesta.idPaziente, Paziente.nome, Paziente.cognome, Paziente.codiceFiscale,
-> Richiesta.idRichiesta, Esame.idEsame, Esame.dataOraEsame,
-> Prestazione.nomePrestazione
-> FROM Paziente INNER JOIN Richiesta ON Richiesta.idPaziente = Paziente.idPaziente
-> INNER JOIN Prestazione ON Richiesta.codicePrestazione = Prestazione.codicePrestazione
-> INNER JOIN Esame ON Esame.idRichiesta = Richiesta.idRichiesta
-> LEFT JOIN Referta ON Esame.idEsame = Referta.idEsame
-> WHERE Esame.statoEsame = 'Completato'
-> AND Referta.idEsame is null;
Query OK, 0 rows affected (0.01 sec)
```

## 11.5 View di consultazione esami

Medici e Tecnici di Radiologia potrebbero avere la necessità di consultare esami precedenti per ottenere informazioni necessarie alla buona riuscita dell'indagine diagnostica. Pertanto la seguente View permette di consultare lo storico degli esami dei pazienti.

```
mysql> CREATE VIEW consultaEsami AS
-> SELECT Richiesta.idPaziente, Prestazione.nomePrestazione, Esame.idEsame, Esame.dataOraEsame,
-> Esame.statoEsame, Esame.commento, referta.referto
-> FROM Richiesta INNER JOIN Prestazione ON Richiesta.codicePrestazione = Prestazione.codicePrestazione
-> INNER JOIN Esame ON Richiesta.idRichiesta = Esame.idRichiesta
-> LEFT JOIN Referta ON Esame.idEsame = Referta.idEsame
-> WHERE date(dataOraEsame) <= current_date();
Query OK, 0 rows affected (0.02 sec)
```

## 12 Stored Procedure

A seguire saranno presentate le Stored Procedure d'esempio create. Gli esempi di funzionamento verranno trattati successivamente nell'ambito della Sicurezza in combinazione con i tipi di utente per cui sono state pensate.

### 12.1 PrenotaAppuntamento

La procedura è pensata per essere assegnata come privilegio agli amministrativi. La sua esecuzione permette di inserire una nuova Richiesta compilando automaticamente i campi di dataRichiesta e oraRichiesta, con la data e ora in cui la procedura viene eseguita, e la matricola dell'amministrativo che procede alla prenotazione. Prende in input come parametri l'id del paziente, il quesito diagnostico, il nome dipartimento, il codice della prestazione richiesta e il codice dell'esenzione, in caso di richieste esterne e se specificata sull'impegnativa del medico curante, in caso contrario verrà inserito null(In caso di assegnazione di esenzione a richieste interne o a pazienti residenti fuori regione verrà automaticamente rimossa dal trigger "Esenzione\_Non\_Applicabile"). La procedura si interrompe inoltre nel momento in cui si tenti di prenotare appuntamenti in date passate, o prestazioni nelle stanze sbagliate, o già prenotate agli stessi orari lo stesso giorno, segnalando un messaggio di errore.

```
mysql> CREATE PROCEDURE PrenotaAppuntamento(in idPz int, in ques varchar(50), in nomeDip
-> varchar(50), in codPres varchar(10), in codEsen varchar(5),
-> in codPrio varchar(5), in numStanza int, in dataApp date, in oraApp time)
-> BEGIN
-> DECLARE disp tinyint;
-> DECLARE tipoSala varchar(5);
-> DECLARE tipoPrest varchar(5);
-> DECLARE prenotante int;
-> DECLARE statoRich varchar(15);
-> DECLARE idRich int;
-> START transaction;
-> SELECT matricola FROM Impiegato WHERE username = user() INTO prenotante;
-> SELECT metodoImaging FROM Sala WHERE numeroStanza = numStanza INTO tipoSala;
-> SELECT tipoPrestazione FROM Prestazione WHERE codicePrestazione = codPres into tipoPrest;
-> IF (nomeDip = 'MC') THEN
->   SELECT statoRichiesta INTO statoRich
->   FROM PlanningAppuntamentiEsterni
->   WHERE data = dataApp
->   AND ora = oraApp
->   AND numeroStanza = numStanza;
-> ELSEIF (nomeDip != 'MC') THEN
```

```

-> SELECT statoRichiesta INTO statoRich
-> FROM PlanningAppuntamentiInterni
-> WHERE data = dataApp
-> AND ora = oraApp
-> AND numeroStanza = numStanza;
-> END IF;
-> IF (statoRich = 'Accettato') THEN
-> signal sqlstate '45000' set message_text = 'Appuntamento non disponibile';
-> rollback;
-> ELSEIF (statoRich = 'Prenotato') THEN
-> signal sqlstate '45000' set message_text = 'Appuntamento non disponibile';
-> rollback;
-> ELSEIF (statoRich is null) THEN
-> IF (dataApp < current_date()) then
-> signal sqlstate '45000' set message_text = 'Impossibile prenotare appuntamento passato';
-> rollback;
-> ELSEIF (dataApp = current_date() and oraApp < current_time()) then
-> signal sqlstate '45000' set message_text = 'Impossibile prenotare appuntamento passato';
-> rollback;
-> ELSEIF (tipoSala != tipoPrest) THEN
-> signal sqlstate '45000' set message_text = 'Prestazione non eseguibile nella Stanza';
-> rollback;
-> ELSEIF (codEsen is null) THEN
-> INSERT INTO Richiesta (idPaziente, quesitoDiagnostico, dataOraRichiesta,
-> codicePrestazione, priorita, nomeDipartimento, dataAppuntamento,
-> oraAppuntamento, numeroStanza, statoRichiesta, matricolaAmministrativo)
-> values (idPz, ques, current_timestamp(), codPres, codPrio, nomeDip, dataApp,
-> oraApp, numStanza, 'Prenotato', prenotante);
-> Commit;
-> ELSEIF (codEsen is not null) THEN
-> INSERT INTO Richiesta (idPaziente, quesitoDiagnostico, dataOraRichiesta,
-> codicePrestazione, priorita, nomeDipartimento, dataAppuntamento,
-> oraAppuntamento, numeroStanza, statoRichiesta, matricolaAmministrativo)
-> values (idPz, ques, current_timestamp(), codPres, codPrio, nomeDip, dataApp,
-> oraApp, numStanza, 'Prenotato', prenotante);
-> SELECT Richiesta.idRichiesta INTO idRich
-> FROM Richiesta
-> WHERE dataAppuntamento = dataApp
-> AND oraAppuntamento = oraApp
-> AND numeroStanza = numStanza
-> AND statoRichiesta = 'Prenotato';
-> INSERT INTO Associata values
-> (idRich, codEsen);
-> Commit;
-> END IF;

```

## RADB

```
-> END IF;  
-> END//  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> delimiter ;
```

## 12.2 AccettaAppuntamento

La procedura è pensata per essere assegnata come privilegio agli amministrativi. La sua esecuzione permette di aggiornare lo stato della richiesta ad accettato nel momento in cui si registra l'arrivo del paziente, e di inserire in automatico la matricola dell'amministrativo che sta accettando la richiesta. La procedura si interrompe segnalando un errore in caso si tenti di accettare richieste inesistenti, già accettate, cancellate, o che siano prenotate, ma la data dell'appuntamento sia passata.

```
mysql> delimiter //  
mysql> CREATE PROCEDURE AccettaAppuntamento(in idRich int)  
-> BEGIN  
-> DECLARE statoRich varchar(15);  
-> DECLARE accettante int;  
-> DECLARE dataApp date;  
-> START transaction;  
-> SELECT matricola FROM Impiegato WHERE username = user() into accettante;  
-> SELECT statoRichiesta FROM Richiesta WHERE idRichiesta = idRich INTO statoRich;  
-> SELECT dataAppuntamento FROM Richiesta WHERE idRichiesta = idRich INTO dataApp;  
-> IF (isnull(statoRich)) THEN  
-> SIGNAL sqlstate '45000' set message_text = 'Appuntamento Inesistente';  
-> rollback;  
-> ELSEIF (statoRich = 'Cancellato') then  
-> SIGNAL sqlstate '45000' set message_text = 'Prenotazione cancellata';  
-> rollback;  
-> ELSEIF (statoRich = 'Accettato') then  
-> SIGNAL sqlstate '45000' set message_text = 'Prenotazione già Accettata';  
-> rollback;  
-> ELSEIF (dataApp < curdate()) then  
-> SIGNAL sqlstate '45000' set message_text = 'Data appuntamento passato';  
-> rollback;  
-> ELSEIF (statoRich = 'Prenotato') then  
-> UPDATE Richiesta  
-> SET statorichiesta = 'Accettato',  
-> matricolaAmministrativo = accettante  
-> WHERE idRichiesta = idRich;
```

```
-> commit;
-> end if;
-> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> delimiter ;
```

## 12.3 CancellaAppuntamento

La procedura è pensata per essere assegnata come privilegio agli amministrativi. La sua esecuzione permette di cancellare prestazioni ancora non avvenute e prenotate per date future, aggiornando automaticamente la matricola dell'amministrativo che interviene sul record. La procedura si interrompe segnalando un errore in caso si tenti di accettare richieste inesistenti, già accettate, cancellate, o che siano prenotate, ma la data dell'appuntamento sia passata.

```
mysql> delimiter //
mysql> CREATE PROCEDURE cancellaAppuntamento(in idRich int)
-> BEGIN
->     DECLARE statoRich varchar(15);
->     DECLARE cancellante varchar(50);
->     DECLARE dataApp date;
->     START transaction;
->     SELECT matricola FROM Impiegato WHERE username = user() INTO cancellante;
->     SELECT statoRichiesta FROM Richiesta WHERE idRichiesta = idRich into statoRich;
->     SELECT dataAppuntamento FROM Richiesta WHERE idRichiesta = idRich into dataApp;
->     if isnull(statoRich) then
->         signal sqlstate '45000' set message_text = 'Appuntamento inesistente';
->         rollback;
->     elseif (statoRich = 'Cancellato') then
->         signal sqlstate '45000' set message_text = 'Prenotazione già cancellata';
->         rollback;
->     elseif (statoRich = 'Accettato') then
->         signal sqlstate '45000' set message_text = 'Impossibile Cancellare Prestazioni Accettate';
->         rollback;
->     elseif (dataApp < curdate()) then
->         signal sqlstate '45000' set message_text = 'Data Prestazione Passata';
->         rollback;
->     elseif (dataApp >= curdate() and statoRich = 'Prenotato') then
->         update Richiesta
->             set statorichiesta = 'Cancellato',
->                 matricolaAmministrativo = cancellante
->             WHERE idRichiesta = idRich;
```

## RADB

```
-> commit;  
-> end if;  
-> END //  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> delimiter ;
```

## 12.4 Estrai Dati Ricevuta

La procedura è pensata per essere assegnata come privilegio agli amministrativi.  
La sua esecuzione permette di estrarre i dati per i report delle ricevute.

```
mysql> delimiter //  
mysql> CREATE PROCEDURE estraiDatiRicevuta(in idPz int, in dataApp date,  
-> out totPres int, out Imponibile decimal(6,2), out netto decimal(6,2))  
-> BEGIN  
-> DECLARE diff decimal(6,2);  
-> SELECT sum(Prestazione.costo)  
-> FROM Richiesta, Associaata, Prestazione, PacchettoPrestazionale  
-> WHERE Richiesta.idPaziente = idPz  
-> AND Richiesta.dataAppuntamento = dataApp  
-> AND Richiesta.statoRichiesta = 'Accettato'  
-> AND Richiesta.idRichiesta = Associaata.idRichiesta  
-> AND Prestazione.codicePrestazione = Richiesta.codicePrestazione  
-> AND Associaata.codiceEsenzione = PacchettoPrestazionale.codiceEsenzione  
-> AND Richiesta.codicePrestazione = PacchettoPrestazionale.codicePrestazione INTO diff;  
-> IF diff is null then  
-> SET diff = 0;  
-> END IF;  
-> SELECT count(Richiesta.idRichiesta), sum(Prestazione.costo), sum(Prestazione.costo)-diff  
-> INTO totPres, imponibile, netto  
-> FROM Richiesta, Prestazione WHERE Richiesta.idPaziente = idPz  
-> AND Richiesta.dataAppuntamento = dataApp and Richiesta.statoRichiesta = 'Accettato'  
-> AND Prestazione.codicePrestazione = Richiesta.codicePrestazione;  
-> SELECT nomePrestazione  
-> FROM Richiesta, Prestazione  
-> WHERE Richiesta.idPaziente = idPz  
-> AND Richiesta.dataAppuntamento = dataApp and Richiesta.statoRichiesta = 'Accettato'  
-> AND Prestazione.codicePrestazione = Richiesta.codicePrestazione;  
-> END //
```

```
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> delimiter ;
```

## 12.5 Aggiorna Paziente

La procedura è pensata per essere assegnata come privilegio agli amministrativi. La sua esecuzione permette di aggiornare determinati campi di Paziente. Possono essere passati come null i parametri che non si desiderano cambiare, la procedura gestisce i diversi casi. Se tutti i parametri sono null la procedura si interrompe e segnala un errore.

```
mysql> delimiter //  
mysql> CREATE PROCEDURE aggiornaPaziente(in idPz int, in ind varchar(50), in ncap int,  
-> in tel varchar(11), in nnote text)  
-> BEGIN  
-> START transaction;  
-> IF ncap is null and ind is null and tel is null and nnote is null then  
-> SIGNAL sqlstate '45000' set message_text = 'Nessun dato da cambiare inserito';  
-> rollback;  
-> ELSEIF ind is not null and ncap is null then  
-> SIGNAL sqlstate '45000' set message_text = 'cap invariato, specificare nuovo cap';  
-> rollback;  
-> ELSEIF ncap is not null and ind is null then  
-> SIGNAL sqlstate '45000' set message_text = 'Indirizzo invariato, specificare nuovo indirizzo';  
-> rollback;  
-> ELSEIF ncap is null and ind is null and nnote is null and tel is not null then  
-> UPDATE PAZIENTE  
-> SET telefono = tel  
-> WHERE idPaziente = idPz ;  
-> commit;  
-> ELSEIF tel is null and nnote is null and ncap is not null and ind is not null then  
-> UPDATE PAZIENTE  
-> SET indirizzo = ind,  
-> cap = ncap  
-> WHERE idPaziente = idPz ;  
-> commit;  
-> ELSEIF nnote is not null and tel is null and ncap is null and ind is null then  
-> UPDATE PAZIENTE  
-> SET note = nnote  
-> WHERE idPaziente = idPz ;  
-> commit;  
-> ELSEIF nnote is not null and tel is not null and ncap is null and ind is null then  
-> UPDATE PAZIENTE  
-> SET note = nnote,  
-> telefono = tel  
-> WHERE idPaziente = idPz ;  
-> commit;
```

## RADB

```
--> ELSEIF nnote is null and tel is not null and ncap is not null and ind is not null then
--> UPDATE PAZIENTE
--> SET indirizzo = ind,
--> cap = ncap,
--> telefono = tel
--> WHERE idPaziente = idPz ;
--> commit;

--> ELSEIF nnote is not null and tel is null and ncap is not null and ind is not null then
--> UPDATE PAZIENTE
--> SET indirizzo = ind,
--> cap = ncap,
--> note = nnote
--> WHERE idPaziente = idPz ;
--> commit;

--> ELSE
--> UPDATE PAZIENTE
--> SET indirizzo = ind,
--> cap = ncap,
--> telefono = tel,
--> note = nnote
--> WHERE idPaziente = idPz;
--> commit;
--> END IF;
--> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> delimiter ;
```

## 12.6 Ritira Referto

La procedura è pensata per essere assegnata come privilegio agli amministrativi. La sua esecuzione permette di estrarre il referto dagli esami se disponibile, altrimenti la procedura si interrompe e segnala un errore. Essa si interrompe e segnala un errore anche in caso l'esame non esista

```
mysql> delimiter //
mysql> CREATE PROCEDURE ritiraReferto(in idRich int, out ref text)
--> BEGIN
--> DECLARE idEs int;
--> SELECT idEsame INTO idEs FROM Esame WHERE Esame.idRichiesta = idRich;
--> IF isnull(idEs) then
--> signal sqlstate '45000' set message_text = 'Esame inesistente';
--> END IF;
--> SELECT Referta.referto INTO ref FROM Referta WHERE Referta.idEsame = idEs;
```

```

-> if isnull(ref) then
-> signal sqlstate '45000' set message_text = 'Referto non disponibile';
-> ELSE
->   SELECT Referta.referto
->   FROM Referta
->   WHERE Referta.idEsame = idEs INTO ref;
-> end if;
-> END //
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> delimiter ;
```

## 12.7 Completa Esame

La procedura è pensata per essere assegnata come privilegio ai Tecnici di Radiologia. La sua esecuzione permette di inserire uno studio completato. Vengono inserite in automatico la data e l'ora del completamento e la matricola del TSRM. Devono essere passati alla procedura i campi relativi alle informazioni sull'esecuzione dell'esame. La procedura si interrompe in caso si stia tentando di completare un esame inesistente, non accettato, annullato, già completato o già refertato e segnala un errore in questi casi.

```

mysql> delimiter //
mysql> CREATE PROCEDURE completaEsame(in idRich int, in immAcc int, in immRif int, in doseRad float, in commento text)
-> BEGIN
->   DECLARE statoRich varchar(50);
->   DECLARE statoEs varchar(50);
->   DECLARE completante int;
->   START transaction;
->   SELECT statoRichiesta FROM Richiesta where idRichiesta = idRich into statoRich;
->   SELECT statoEsame FROM Esame where idRichiesta = idRich into statoEs;
->   SELECT matricola FROM Impiegato WHERE username = user() INTO completante;
->   IF (statoRich != 'Accettato') THEN
->     SIGNAL sqlstate '45000' set message_text = 'Impossibile completare un esame non accettato';
->     rollback;
->   ELSEIF (statoEs = 'Completato') THEN
->     SIGNAL sqlstate '45000' set message_text = 'Esame già completato';
->     rollback;
->   ELSEIF (statoEs = 'Annullato') then
->     SIGNAL sqlstate '45000' set message_text = 'Esame Annullato';
->     rollback;
->   ELSE
->     INSERT INTO Esame (idRichiesta, dataOraEsame, matricolaTSRM,
->     statoEsame, immaginiAccettate, immaginiRifiutate,
```

## RADB

```
-> dosemGycm, commento) values
-> (idRich, current_timestamp(), completante, 'Completato', immAcc, immRif, doseRad, commento);
-> commit;
-> end if;
-> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> delimiter ;
```

## 12.8 Annulla Esame

La procedura è pensata per essere assegnata come privilegio ai Tecnici di Radiologia. La sua esecuzione permette di inserire uno studio annullato. Vengono inserite in automatico la data e l'ora dell'annullamento e la matricola del TSRM. Deve essere passato alla procedura la motivazione dell'annullamento che verrà inserita nel campo commenti. La procedura si interrompe in caso si stia tentando di annullare un esame inesistente, non accettato, già annullato, completato o refertato e segnala un errore in questi casi.

```
mysql> delimiter //
mysql> CREATE PROCEDURE annullaEsame(in idRich int, in commento text)
-> BEGIN
-> DECLARE statoRich varchar(50);
-> DECLARE statoEs varchar(50);
-> DECLARE completante int;
-> START transaction;
-> SELECT statoRichiesta FROM Richiesta where idRichiesta = idRich into statoRich;
-> SELECT Esame.statoEsame FROM Esame where Esame.idRichiesta = idRich into statoEs;
-> SELECT matricola FROM Impiegato WHERE username = user() INTO completante;
-> IF (statoRich != 'Accettato') THEN
-> SIGNAL sqlstate '45000';
-> SET message_text = 'Impossibile annullare un esame non accettato, cancellare richiesta';
-> rollback;
-> ELSEIF (statoEs = 'Annullato') then
-> SIGNAL sqlstate '45000' set message_text = 'Esame già annullato';
-> rollback;
-> ELSEIF (statoEs = 'Completato') then
-> SIGNAL sqlstate '45000' set message_text = 'Impossibile annullare un esame completato';
-> rollback;
-> ELSE
-> INSERT INTO Esame (idRichiesta, dataOraEsame, matricolaTSRM, statoEsame, commento) values
-> (idRich, current_timestamp(), completante, 'Annullato', commento);
-> commit;
-> END IF;
-> END //
```

```
Query OK, 0 rows affected (0.02 sec)
mysql> delimiter ;
```

## 12.9 Referta Esame

La procedura è pensata per essere assegnata come privilegio ai Medici di Radiologia. La sua esecuzione permette di inserire la matricola del medico refertante, id dell'esame refertato, il referto e in automatico la data di refertazione. La procedura si interrompe in caso si stia tentando di refertare un esame inesistente, non accettato, annullato, refertato e segnala un errore in questi casi.

```
mysql> delimiter //
mysql> CREATE PROCEDURE refertaEsame(in idEs int, in ref text)
-> BEGIN
->     DECLARE statoEs varchar(50);
->     DECLARE refertante int;
->     DECLARE ref2 text;
->     START transaction;
->     SELECT statoEsame FROM Esame WHERE idEsame = idEs into statoEs;
->     SELECT matricola FROM Impiegato WHERE username = user() INTO refertante;
->     SELECT referto INTO ref2 FROM Referta WHERE Referta.idEsame = idEs;
->     IF (ref2 is not null) then
->         signal sqlstate '45000' set message_text = 'Esame già refertato';
->         rollback;
->     ELSEIF(statoEs = 'Annullato') then
->         signal sqlstate '45000' set message_text = 'Impossibile refertare esame annullato';
->         rollback;
->     ELSEIF(isnull(statoEs)) then
->         signal sqlstate '45000' set message_text = 'Esame inesistente';
->         rollback;
->     ELSEIF(ref is null) then
->         signal sqlstate '45000' set message_text = 'Referto non inserito';
->         rollback;
->     ELSE
->         INSERT INTO Referta values(refertante, idEs, ref, current_timestamp());
->         commit;
->     end if;
-> END //
```

```
Query OK, 0 rows affected (0.02 sec)
```

## 13 Sicurezza

Per il database sono Previste 4 tipologie di utenti:

- Amministratore di Sistema
- Amministrativo
- Medico Radiologo
- Tecnico di Radiologia.

Verranno presi 4 utenti, uno per tipologia come esempio per l'assegnazione dei permessi sul database RaDB:

```
mysql> CREATE user MorlacchiPier IDENTIFIED BY 'MorlacchiPier';
Query OK, 0 rows affected (0.02 sec)

mysql> GRANT ALL ON RaDB.* TO 'MorlacchiPier' WITH grant option;
Query OK, 0 rows affected (0.00 sec)
```

Il secondo utente, di tipo Amministrativo, avrà accesso limitato alle funzioni del database. Tipicamente un amministrativo ha il compito di prenotare le richieste nel sistema, registrare i nuovi pazienti qualora non presenti nel database, accettare i pazienti che si presentano per l'appuntamento e stampare le ricevute di pagamento. Pertanto verranno assegnati privilegi su Stored Procedure apposite, piuttosto che alle singole insert e update su campi specifici, sulle select limitate a Paziente, e su View che mostrino loro i dati limitatamente alla loro area di competenza.

```
mysql> CREATE user SokolovEmilio IDENTIFIED BY 'SokolovEmilio';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT INSERT ON RaDB.Paziente TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT SELECT ON RaDB.Paziente TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT SELECT ON RaDB.planningAppuntamentiEsterni TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT SELECT ON RaDB.planningAppuntamentiInterni TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT SELECT ON RaDB.worklistAmministrativi TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT EXECUTE ON PROCEDURE RaDB.PrenotaAppuntamento TO 'SokolovEmilio';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT EXECUTE ON PROCEDURE RaDB.AccettaAppuntamento TO 'SokolovEmilio';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT EXECUTE ON PROCEDURE RaDB.cancellaAppuntamento TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT EXECUTE ON PROCEDURE RaDB.estraiDatiRicevuta TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)
```

```

mysql> GRANT EXECUTE ON PROCEDURE RaDB.aggiornaPaziente TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT EXECUTE ON PROCEDURE RaDB.ritiraReferto TO 'SokolovEmilio';
Query OK, 0 rows affected (0.00 sec)

```

La seguente sequenza di chiamate rappresenta una routine tipica di un amministrativo.

Cancellazione di un appuntamento (le select su richiesta sono state fatte tramite utente root): prese le seguenti richieste come esempio e si procederà con la chiamata della procedura ‘cancellaAppuntamento’ da parte dell’utente amministrativo creato.

```

mysql> SELECT * FROM Richiesta WHERE dataAppuntamento = '2021-08-20' AND statoRichiesta = 'Prenotato' limit 1;
+-----+-----+-----+-----+-----+-----+-----+-----+
| idRichiesta | idPaziente | quesitoDiagnostico | dataOraRichiesta | codicePrestazione | priorita | nomeDipartimento | dataAppuntamento | OraAppuntamento | numeroStanza |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 6586 | 36 | BPC0 | 2021-08-17 16:00:00 | RX29 | 3 | NC | 2021-08-20 | 16:00:00 | 2 |
+-----+-----+-----+-----+-----+-----+-----+-----+
statoRichiesta | matricolaAmministrativo |
+-----+-----+
Prenotato | 30 |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM Richiesta WHERE dataAppuntamento > curdate() AND statoRichiesta = 'Prenotato' limit 1;
+-----+-----+-----+-----+-----+-----+-----+
| idRichiesta | idPaziente | quesitoDiagnostico | dataOraRichiesta | codicePrestazione | priorita | nomeDipartimento | dataAppuntamento | OraAppuntamento | numeroStanza |
+-----+-----+-----+-----+-----+-----+-----+
| 12629 | 111 | Connettività Indifferenziate | 2021-08-21 00:30:00 | RX10SN | 3 | WI0 | 2021-08-22 | 00:30:00 | 1 |
+-----+-----+-----+-----+-----+-----+-----+
numeroStanza | statoRichiesta | matricolaAmministrativo |
+-----+-----+
1 | Prenotato | 27 |
+-----+
1 row in set (0.01 sec)

mysql> SELECT * FROM Richiesta WHERE dataAppuntamento = '2021-08-20' AND statoRichiesta = 'Accettato' limit 1;
+-----+-----+-----+-----+-----+-----+-----+
| idRichiesta | idPaziente | quesitoDiagnostico | dataOraRichiesta | codicePrestazione | priorita | nomeDipartimento | dataAppuntamento | OraAppuntamento | numeroStanza |
+-----+-----+-----+-----+-----+-----+-----+
| 6523 | 4 | Ostomielite Cronica | 2021-08-17 08:00:00 | RX10SN | 3 | NC | 2021-08-20 | 08:00:00 | 2 |
+-----+-----+-----+-----+-----+-----+-----+
statoRichiesta | matricolaAmministrativo |
+-----+-----+
Accettato | 28 |
+-----+

```

## RADB

```
(base) Eleonora:~ stefanogalgano$ mysql -u SokolovEmilio -p
mysql> call cancellaAppuntamento(6586);
ERROR 1644 (45000): Data Prestazione Passata

mysql> call cancellaAppuntamento(12529);
Query OK, 0 rows affected (0.01 sec)

mysql> call cancellaAppuntamento(12529);
ERROR 1644 (45000): Prenotazione già cancellata

mysql> call cancellaAppuntamento(6523);
ERROR 1644 (45000): Impossibile Cancellare Prestazioni Accettate
```

### Controllo dell'effettivo funzionamento della chiamata su "idRichiesta" = 12529

```
mysql> SELECT * FROM Richiesta WHERE idRichiesta = 12529;
+-----+-----+-----+-----+-----+-----+-----+
| idRichiesta | idPaziente | quesitoDiagnostico | dataOraRichiesta | codicePrestazione | priorita | nomeDipartimento |
+-----+-----+-----+-----+-----+-----+-----+
| 12529 | 111 | Connivenza Indifferenziate | 2021-08-21 00:30:00 | RX10SN | 3 | VI9 |
|       |       |                         |                   |                 |         |           |
|       |       |                         | 2021-08-22        |               | 00:30:00 |           |
+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+
| numeroStanza | statoRichiesta | matricolaAmministrativo |
+-----+-----+-----+
| 1 | Cancellato | 25 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

### Prenotazione e Accettazione di un appuntamento:

```
mysql> SELECT * FROM PlanningAppuntamentiEsterni WHERE numeroStanza = 2 ORDER BY data, ora LIMIT 5;
+-----+-----+-----+-----+-----+
| idRichiesta | statoRichiesta | data | ora | numerostanza |
+-----+-----+-----+-----+-----+
| NULL | NULL | 2021-08-21 | 10:30:00 | 2 |
| 6695 | Prenotato | 2021-08-21 | 11:00:00 | 2 |
| 6696 | Prenotato | 2021-08-21 | 11:30:00 | 2 |
| NULL | NULL | 2021-08-21 | 12:00:00 | 2 |
| 6713 | Prenotato | 2021-08-21 | 12:30:00 | 2 |
+-----+-----+-----+-----+-----+
5 rows in set (0.06 sec)

mysql> CALL prenotaAppuntamento(70, 'BPCO', 'MC', 'RX29', 'E05', 2, 2, '2021-08-21', '09:00:00');
ERROR 1644 (45000): Impossibile prenotare appuntamento passato

mysql> CALL prenotaAppuntamento(70, 'BPCO', 'MC', 'RX29', 'E05', 2, 2, '2021-08-20', '10:00:00');
ERROR 1644 (45000): Impossibile prenotare appuntamento passato

mysql> CALL prenotaAppuntamento(70, 'BPCO', 'MC', 'RX29', 'E05', 2, 2, '2021-08-21', '11:00:00');
ERROR 1644 (45000): Prestazione non eseguibile nella Stanza

mysql> CALL prenotaAppuntamento(70, 'BPCO', 'MC', 'RX29', 'E05', 2, 2, '2021-08-21', '11:00:00');
ERROR 1644 (45000): Appuntamento non disponibile
```

```

mysql> CALL prenotaAppuntamento(70, 'BPCO', 'MC', 'RX29', 'E05', 2, 2, '2021-08-21', '10:30:00');
ERROR 1644 (45000): Paziente residente fuori regione, esenzione non Applicabile
mysql> CALL prenotaAppuntamento(70, 'BPCO', 'MC', 'RX29', 'E05', 2, 2, '2021-08-21', '10:30:00');
ERROR 1644 (45000): Appuntamento non disponibile
mysql> SELECT * FROM worklistAmministrativi WHERE stanza = 2 AND ora >= current_time() LIMIT 5;
+-----+-----+-----+-----+-----+-----+-----+-----+
| idPaziente | nome | cognome | codiceFiscale | idRichiesta | stanza | data | ora | prestazione |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      70 | Lilla | Riccati | RCCLLL64T41F205X |     18324 |      2 | 2021-08-21 | 10:30:00 | RX Torace |
|       1 | Flora | Calbo  | CLBFLR72T57L219F |     6695  |      2 | 2021-08-21 | 11:00:00 | RX Torace |
|      47 | Lisa  | Petruzzi | PTRLSI70D52G273Q |     6696  |      2 | 2021-08-21 | 11:30:00 | RX Braccio Sinistro |
|      84 | Mirko | Passalacqua | PSSMRK72L03A944X |     6713  |      2 | 2021-08-21 | 12:30:00 | RX Spalla Sinistra |
|      84 | Mirko | Passalacqua | PSSMRK72L03A944X |     6714  |      2 | 2021-08-21 | 13:00:00 | RX Rachide Cervicale |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)

mysql> CALL accettaappuntamento(6649);
ERROR 1644 (45000): Prenotazione cancellata
mysql> CALL accettaappuntamento(664900000);
ERROR 1644 (45000): Appuntamento Inesistente
mysql> CALL accettaappuntamento(7947);
ERROR 1644 (45000): Data appuntamento passato
mysql> CALL accettaappuntamento(18324);
Query OK, 0 rows affected (0.00 sec)

mysql> CALL accettaappuntamento(18324);
ERROR 1644 (45000): Prenotazione già Accettata

```

### Estrazione dei dati per il report di ricevuta:

```

mysql> CALL estraiDatiRicevuta(70, '2021-08-21', @totalePrestazioni, @imponibile, @netto);
+-----+
| nomePrestazione |
+-----+
| RX Torace      |
+-----+
1 row in set (0.00 sec)

```

Query OK, 0 rows affected (0.00 sec)

```

mysql> SELECT @totalePrestazioni, @imponibile, @netto;
+-----+-----+-----+
| @totalePrestazioni | @imponibile | @netto |
+-----+-----+-----+
|           1 |      15.49 |   15.49 |
+-----+-----+-----+

```

## RADB

```
1 row in set (0.00 sec)

mysql> CALL accettaappuntamento(6695);
Query OK, 0 rows affected (0.00 sec)
```

Di seguito il risultato delle chiamate delle procedure (select su Richiesta eseguite tramite utente root) andate a buon fine, la query su “Associata” non presenta tuple in quanto “E05” è un’esonzione regionale e il paziente risiede fuori dal Lazio, per cui ha innescato il Trigger “Esenzione\_Non\_Applicabile”.

```
mysql> SELECT * FROM Associata WHERE idRichiesta = 18324;
Empty set (0.00 sec)
```

```
mysql> select * from Richiesta where idRichiesta = 18324;
+-----+
| idRichiesta | idPaziente | quesitoDiagnostico | dataOraRichiesta | codicePrestazione | priorita | nomeDipartimento |
+-----+
| 18324 | 70 | BPCO | 2021-08-21 10:01:59 | RX29 | 2 | MC |
+-----+
+-----+
| dataAppuntamento | OraAppuntamento | numeroStanza | statoRichiesta | matricolaAmministrativo |
+-----+
| 2021-08-21 | 10:30:00 | 2 | Accettato | 25 |
+-----+
1 row in set (0.00 sec)
```

Ritiro del referto (eseguito dopo le chiamate “completaEsame” e “refertaEsame” da parte dei rispettivi utenti):

```
(base) Eleonora:~ stefanogalgano$ mysql -u SokolovEmilio -p
mysql> CALL ritiraReferto(000000, @referto);
ERROR 1644 (45000): Esame inesistente
mysql> CALL ritiraReferto(16139, @referto);
ERROR 1644 (45000): Referto non disponibile
mysql> CALL ritiraReferto(18324, @referto);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT @referto;
+-----+
| @referto | 
+-----+
| Negativo per BPCO | 
+-----+
1 row in set (0.00 sec)
```

A seguire esempi di tutte le possibili chiamate sulla procedura “aggiornaPaziente”:

```
(base) Eleonora:~ stefanogalgano$ mysql -u SokolovEmilio -p

mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;

+-----+-----+-----+-----+
| indirizzo | cap | telefono | note |
+-----+-----+-----+-----+
| Strada Mario, 014 | 50012 | 1252162 | Nessuna condizione nota |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, null, null, null, null);

ERROR 1644 (45000): Nessun dato da cambiare inserito

mysql> CALL aggiornaPaziente(10, 'Via dei Gelsi, 3', null, null, null);

ERROR 1644 (45000): cap invariato, specificare nuovo cap

mysql> CALL aggiornaPaziente(10, null, 118, null, null);

ERROR 1644 (45000): Indirizzo invariato, specificare nuovo indirizzo

mysql> CALL aggiornaPaziente(10, 'Via dei Gelsi, 3', 118, null, null);

Query OK, 0 rows affected (0.00 sec)

mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;

+-----+-----+-----+-----+
| indirizzo | cap | telefono | note |
+-----+-----+-----+-----+
| Via dei Gelsi, 3 | 118 | 1252162 | Nessuna condizione nota |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, null, null, '1234', null);

Query OK, 0 rows affected (0.01 sec)

mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;

+-----+-----+-----+-----+
| indirizzo | cap | telefono | note |
+-----+-----+-----+-----+
| Via dei Gelsi, 3 | 118 | 1234 | Nessuna condizione nota |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, null, null, null, 'Soggetto allergico');

Query OK, 0 rows affected (0.01 sec)
```

## RADB

```
mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;
+-----+-----+-----+-----+
| indirizzo | cap | telefono | note          |
+-----+-----+-----+-----+
| Via dei Gelsi, 3 | 118 | 1234      | Soggetto allergico |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, null, null, '2345', 'Allergia mdc');
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;
+-----+-----+-----+-----+
| indirizzo | cap | telefono | note          |
+-----+-----+-----+-----+
| Via dei Gelsi, 3 | 118 | 2345      | Allergia mdc |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, 'Parco della Vittoria, 10', 40121, '1234', null);
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;
+-----+-----+-----+-----+
| indirizzo | cap | telefono | note          |
+-----+-----+-----+-----+
| Parco della Vittoria, 10 | 40121 | 1234      | Allergia mdc |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, 'Via Roma, 15', 20121, null, 'Allergia a mdc iodati');
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;
+-----+-----+-----+-----+
| indirizzo | cap | telefono | note          |
+-----+-----+-----+-----+
| Via Roma, 15 | 20121 | 1234      | Allergia a mdc iodati |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> CALL aggiornaPaziente(10, 'Via Marco Polo, 100', 80121, '4567', 'Allergia a mdc iodati');
```

```
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> SELECT indirizzo, cap, telefono, note FROM Paziente where idPaziente = 10;
+-----+-----+-----+-----+
| indirizzo | cap | telefono | note |
+-----+-----+-----+-----+
| Via Marco Polo, 100 | 80121 | 4567 | Allergia a mdc iodati |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Il terzo utente, di tipo Tecnico di Radiologia, avrà accesso limitato alle funzioni del database. Tipicamente a un tecnico compete il controllo dell'anagrafica del paziente(al fine di evitare di svolgere l'esame sulla persona sbagliata), l'esecuzione dell'esame e il completamento dello stesso con tutti i dettagli a esso relativi. Per l'esecuzione di una buona indagine diagnostica può essere utile e/o necessario accedere a informazioni su esami precedenti e sulle note del paziente. Pertanto verranno assegnati privilegi su Stored Procedure apposite, piuttosto che alle insert e update su campi specifici, sulle select limitate a Paziente, e su View che mostrino loro i dati limitatamente alla loro area di competenza.

```
mysql> CREATE user FedericoEva IDENTIFIED BY 'FedericoEva';
Query OK, 0 rows affected (0.01 sec)
mysql> GRANT SELECT on RaDB.Paziente TO 'FedericoEva';
Query OK, 0 rows affected (0.00 sec)
mysql> GRANT SELECT on RaDB.worklistTSRM TO 'FedericoEva';
Query OK, 0 rows affected (0.00 sec)
mysql> GRANT SELECT on RaDB.consultaEsami TO 'FedericoEva';
Query OK, 0 rows affected (0.00 sec)
mysql> GRANT EXECUTE on PROCEDURE RaDB.completaEsame TO 'FedericoEva';
Query OK, 0 rows affected (0.01 sec)
mysql> GRANT EXECUTE on PROCEDURE RaDB.annullaEsame TO 'FedericoEva';
Query OK, 0 rows affected (0.00 sec)
```

La seguente sequenza di chiamate rappresenta la routine di un tecnico di radiologia: un tecnico può eseguire e quindi completare esami oppure annullarli.

### Consultazione della Worklist:

```
(base) Eleonora:~ stefanogalgano$ mysql -u FedericoEva -p
mysql> SELECT * FROM worklistTSRM ORDER BY oraAppuntamento LIMIT 5;
```

## RADB

```
+-----+-----+-----+-----+-----+-----+-----+
| idPaziente | nome | cognome | codiceFiscale | note | idRichiesta | oraAppuntamento | numeroStanza |
+-----+-----+-----+-----+-----+-----+-----+
|      70 | Lilla | Riccati | RCCLLL64T41F206X | Allergia a mdc gadolinio |     18324 | 10:30:00 |          2 |
|      1 | Flora | Calbo   | CLBFLR72T57L219F | Nessuna condizione nota |      6695 | 11:00:00 |          2 |
+-----+-----+-----+-----+-----+-----+-----+
-----+
nomePrestazione |
```

idPaziente	nome	cognome	codiceFiscale	note	idRichiesta	oraAppuntamento	numeroStanza
70	Lilla	Riccati	RCCLLL64T41F206X	Allergia a mdc gadolinio	18324	10:30:00	2
1	Flora	Calbo	CLBFLR72T57L219F	Nessuna condizione nota	6695	11:00:00	2

nomePrestazione
RX Torace
RX Torace

```
2 rows in set (0.00 sec)
```

### Completamento di un esame:

```
mysql> CALL completaEsame(6696, 2, 2, 4.5, 'Eseguito in AP');
ERROR 1644 (45000): Impossibile completare un esame non accettato
mysql> CALL completaEsame(17251, 2, 2, 4.5, 'Eseguito in AP');
ERROR 1644 (45000): Esame già completato
mysql> CALL completaEsame(18324, 2, 2, 4.5, 'Eseguito in AP');
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> SELECT * FROM worklistTSRM ORDER BY oraAppuntamento LIMIT 5;
Empty set (0.01 sec)
```

### Annulloamento di un esame:

```
mysql> CALL annullaEsame(6696, 'Paziente si rifiuta');
ERROR 1644 (45000): Impossibile annullare un esame non accettato, cancellare richiesta
mysql> CALL annullaEsame(18324, 'Paziente si rifiuta');
ERROR 1644 (45000): Impossibile annullare un esame completato
mysql> CALL annullaEsame(6695, 'Paziente si rifiuta');
Query OK, 0 rows affected (0.00 sec)

mysql> CALL annullaEsame(6695, 'Paziente si rifiuta');
ERROR 1644 (45000): Esame già annullato
mysql> CALL completaEsame(6695, 2, 2, 4.5, 'Eseguito in AP');
ERROR 1644 (45000): Esame Annullato
mysql> SELECT * FROM worklistTSRM ORDER BY oraAppuntamento LIMIT 5;
Empty set (0.00 sec)
```

Di seguito il risultato delle chiamate delle procedure (select su Esame eseguite tramite utente root) andate a buon fine:

```
mysql> SELECT * FROM Esame WHERE idEsame = 9990;
+-----+-----+-----+-----+-----+-----+-----+
| idEsame | idRichiesta | dataOraEsame | matricolaTSRM | statoEsame | immaginiAccettate | immaginiRifiutate | dosemGycm | commento |
+-----+-----+-----+-----+-----+-----+-----+
| 9990 | 18324 | 2021-08-21 16:15:28 | 1 | Completato | 2 | 2 | 4.50 | Eseguito in AP |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM Esame WHERE idEsame = 9992;
+-----+-----+-----+-----+-----+-----+-----+
| idEsame | idRichiesta | dataOraEsame | matricolaTSRM | statoEsame | immaginiAccettate | immaginiRifiutate | dosemGycm | commento |
+-----+-----+-----+-----+-----+-----+-----+
| 9992 | 6695 | 2021-08-21 16:21:37 | 11 | Annullato | 0 | 0 | 0.00 | Paziente si rifiuta |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Il quarto utente, di tipo Medico Radiologo, avrà accesso limitato alle funzioni del database. Tipicamente un Medico Radiologo interviene sul sistema per l'inserimento dei referti di esami a lui assegnati, e per la consultazione di dati relativi al paziente che possano aiutare la diagnosi. Pertanto verranno assegnati privilegi su Stored Procedure apposite, piuttosto che alle singole insert e update sui campi referto e dataReferto, sulle select limitate a Paziente, e su View che mostrino loro i dati essenziali alla formulazione di una diagnosi.

```
mysql> CREATE user DisderoAnita IDENTIFIED BY 'DisderoAnita';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT SELECT on RaDB.Paziente TO 'DisderoAnita';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT SELECT on RaDB.worklistMR TO 'DisderoAnita';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT SELECT on RaDB.consultaEsami TO 'DisderoAnita';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT EXECUTE on PROCEDURE RaDB.RefertaEsame TO 'DisderoAnita';
Query OK, 0 rows affected (0.01 sec)
```

La seguente sequenza di chiamate rappresenta la routine di un Medico Radiologo: un medico referta esami ed eventualmente può consultare i precedenti per coadiuvare la diagnosi.

### Consultazione della Worklist:

```
(base) Eleonora:~ stefanogalgano$ mysql -u DisderoAnita -p
mysql> select * FROM worklistMR WHERE date(dataOraEsame) = current_date() AND time(dataOraEsame) >= '07:00:00' ORDER BY dataOraEsame
LIMIT 5;
```

## RADB

```
+-----+-----+-----+-----+-----+-----+-----+
| idPaziente | nome      | cognome    | codiceFiscale | idRichiesta | idEsame   | dataOraEsame      | nomePrestazione |
+-----+-----+-----+-----+-----+-----+-----+
|      12 | Puccio   | Govoni     | GVNPPCC70B06F839C |      12427 | 6964 | 2021-08-21 07:04:00 | RX Rachide Cervicale |
|    138 | Roberto  | Sgalambro  | SGLRRT81R28H501F |      16139 | 8655 | 2021-08-21 07:10:00 | RM Colonna Lombare  |
|     70 | Lilla    | Riccati    | RCCLLL64T41F205X |      18324 | 9090 | 2021-08-21 10:15:23 | RX Torace          |
+-----+-----+-----+-----+-----+-----+-----+
```

3 rows in set (0.00 sec)

```
mysql> SELECT idEsame, dataOraEsame, referto FROM consultaEsami WHERE idPaziente = 70 AND nomePrestazione = 'TC Torace';
+-----+-----+-----+
| idEsame | dataOraEsame      | referto          |
+-----+-----+-----+
|    4010 | 2021-07-27 17:54:00 | Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista |
|    7263 | 2021-07-27 07:55:00 | Assenza di segni riferibili come causa della patologia |
+-----+-----+-----+
```

2 rows in set (0.00 sec)

### Refertazione dell'esame:

```
mysql> CALL refertaEsame(6695000, 'Negativo per BPCO');
ERROR 1644 (45000): Esame inesistente
mysql> CALL refertaEsame(9092, 'Negativo per BPCO');
ERROR 1644 (45000): Impossibile refertare esame annullato
mysql> CALL refertaEsame(9090, null);
ERROR 1644 (45000): Referto non inserito
mysql> CALL refertaEsame(9090, 'Negativo per BPCO');

Query OK, 0 rows affected (0.00 sec)
```

Di seguito il risultato delle chiamate della procedura (select su Referta eseguite tramite utente root) andata a buon fine:

```
mysql> SELECT * FROM Referta WHERE idEsame = 9090;
+-----+-----+-----+
| matricolaMR | idEsame | referto      | dataOraReferito |
+-----+-----+-----+
|      3 |    9090 | Negativo per BPCO | 2021-08-21 10:34:11 |
+-----+-----+-----+
```

1 row in set (0.00 sec)

Gli esempi precedenti sono stati effettuati sui dati di prova e si considera come giorno lavorativo in cui vengono eseguiti il 21-08-2021.

# 14 Query

Le seguenti query sono state effettuate su dati di prova, prima dell'inserimento massivo di oltre 200000 record. I dati di prova inseriti per le richieste e gli esami sono relativi ai mesi di Luglio 2021 e Agosto 2021. Per tutte le query effettuate si considera come giorno lavorativo in cui vengono eseguite il 21-08-2021.

## 14.1 Tecnici di turno

“Si vogliono sapere matricola, nome, cognome dei tecnici di radiologia che hanno lavorato nella stanza 3 dalle 8:30 alle 13:30 il “07-08-2021”

*La query è usata qualora si riscontrassero problemi di vario genere in uno o più esami eseguiti in un certo arco di tempo, e si vuole risalire ai tecnici in turno al momento dell'esecuzione per approfondire la questione.*

```
mysql> SELECT Impiegato.matricola, Impiegato.nome, Impiegato.cognome
   > FROM Impiegato, Esame, Richiesta
   > WHERE Esame.dataOraEsame
   > BETWEEN '2021-08-07 8:30:00' AND '2021-08-07 13:30:00'
   > AND richiesta.numeroStanza = 3
   > AND Impiegato.matricola = Esame.matricolaTSRM
   > AND Richiesta.idRichiesta = Esame.idRichiesta
   > GROUP BY Impiegato.matricola;
+-----+-----+
| matricola | nome      | cognome    |
+-----+-----+
|       23 | Caterina  | Ceravolo   |
|       17 | Natalia   | Marino     |
|       16 | Virginia  | Caracciolo |
|       12 | Virginia  | Chinnici   |
|       15 | Mariana   | Colletti   |
+-----+-----+
5 rows in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
5 rows in set (0.02 sec)
```

## 14.2 Disponibilità

“Si vogliono sapere i primi 5 posti disponibili, con il numero della stanza la data e orario, per la prenotazione di una richiesta proveniente dall'esterno della struttura per un RX.

*La query è utile per gestire le prenotazioni.*

```
mysql> SELECT data, ora, Sala.numeroStanza
-> FROM Calendario INNER JOIN Orario INNER JOIN Sala
-> LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta
-> WHERE statoRichiesta != 'Cancellato') AS agenda
-> ON agenda.dataAppuntamento = Calendario.data
-> AND agenda.oraAppuntamento = Orario.ora
-> AND agenda.numeroStanza = Sala.numeroStanza
-> WHERE Sala.Riservata = 'Esterne'
-> AND Sala.metodoImaging = 'RX'
-> AND timestamp(data, ora) >= current_timestamp()
-> AND ora >= '8:00'
-> AND ora <= '19:00'
-> AND Festivo(data) = 0
-> AND agenda.idRichiesta is null
-> ORDER BY data, ora
-> limit 5;
+-----+-----+-----+
| data      | ora      | numeroStanza |
+-----+-----+-----+
| 2021-08-21 | 09:00:00 |      2 |
| 2021-08-21 | 10:00:00 |      2 |
| 2021-08-21 | 10:30:00 |      2 |
| 2021-08-21 | 12:00:00 |      2 |
| 2021-08-21 | 13:30:00 |      2 |
+-----+-----+-----+
5 rows in set (0.04 sec)
```

Tempo dopo inserimento massivo di record

```
5 rows in set (1.50 sec)
```

## 14.3 Prenotazioni a seguire

“Si vogliono sapere data e ora di appuntamento di tutte le prestazioni prenotate in ordine di appuntamento con il loro nome, compresa eventualmente la data di oggi, ma non le date passate, per un determinato paziente sapendone il codice fiscale”

*La query serve qualora un paziente richieda informazioni a riguardo.*

```
mysql> SELECT codiceFiscale, Prestazione.nomePrestazione, Richiesta.dataAppuntamento, Richiesta.oraAppuntamento
-> FROM Prestazione, Richiesta, Paziente
-> WHERE Paziente.codiceFiscale = 'BMBMRZ08S26A662R'
-> AND Richiesta.statoRichiesta = 'Prenotato'
-> AND timestamp(Richiesta.dataAppuntamento, Richiesta.oraAppuntamento) >= current_timestamp()
-> AND Richiesta.idPaziente = Paziente.idPaziente
-> AND Prestazione.codicePrestazione = Richiesta.codicePrestazione
-> ORDER BY Richiesta.dataAppuntamento, Richiesta.oraAppuntamento;
```

codiceFiscale	nomePrestazione	dataAppuntamento	oraAppuntamento
BMBMRZ08S26A662R	Ecografia Addome Inferiore	2021-08-21	16:00:00
BMBMRZ08S26A662R	Angio-TC Aorta Addominale	2021-08-22	04:30:00
BMBMRZ08S26A662R	RM Colonna Cervicale	2021-08-22	07:00:00
BMBMRZ08S26A662R	Eco-color-doppler-grafia Cardiaca	2021-08-23	23:00:00
BMBMRZ08S26A662R	RM Ginocchio Sinistro	2021-08-25	08:30:00
BMBMRZ08S26A662R	RM Ginocchio Destro	2021-08-25	09:00:00
BMBMRZ08S26A662R	RM Colonna Cervicale	2021-08-26	03:00:00
BMBMRZ08S26A662R	TC Cranio con e senza mdc	2021-08-26	09:30:00
BMBMRZ08S26A662R	TC Cranio con e senza mdc	2021-08-26	11:30:00
BMBMRZ08S26A662R	Ecografia Tiroide	2021-08-26	12:00:00
BMBMRZ08S26A662R	RM Addome Superiore con mdc	2021-08-27	08:00:00
BMBMRZ08S26A662R	RM Addome Superiore	2021-08-31	13:00:00

12 rows in set (0.01 sec)

Tempo dopo inserimento massivo di record:

12 rows in set (0.02 sec)

## 14.4 Storico Esami

“Si vuole sapere lo storico degli esami del paziente con id 55. In particolare si vuole sapere il nome della prestazione, il quesito diagnostico e il referto.”

*La query è utile nel caso siano stati fatti esami inerenti lo stesso quesito diagnostico la cui refertazione risulti utile all'esecuzione del nuovo studio e/o alla sua refertazione.*

```
mysql> SELECT Esame.idEsame, Prestazione.nomePrestazione, Richiesta.quesitoDiagnostico, Referta.referto
-> FROM Richiesta, Esame, Prestazione, Referta
-> WHERE richiesta.idPaziente = '55'
-> AND Esame.idEsame = Referta.idEsame
-> AND Richiesta.idRichiesta = Esame.idRichiesta
-> AND Richiesta.codicePrestazione = Prestazione.codicePrestazione;
```

idEsame   nomePrestazione	quesitoDiagnostico	referto
1214   RX Mammografia Monolaterale Sinistra	Screening Tumore Mammella	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista
1239   RX Ortopanoramica	Valutazione mezzo di sintesi	Impianto non in sede. Livre dislocazione anatomica
1427   TC Cranio con e senza mdc	Emarginata celobrale	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista
1488   RX Mammografia Bilaterale	Trauma	Assenza di fratture e/o versamenti
1572   Angio-TC Aorta Toracica	Anuramia Aorta Toracica	Non Apprezzabili segni riferibili alla patologia
1573   TC Torace con e senza mdc	Anuramia Aorta Toracica	Non Apprezzabili segni riferibili alla patologia
1838   RM Encefalo	Epilessia	Non Apprezzabili segni riferibili alla patologia
2386   RM Encelalo	Epilessia	Non Apprezzabili segni riferibili alla patologia
2312   RM Encelalo con mdc	Naufragio Ipotisico	Non Apprezzabili segni riferibili alla patologia
2332   RX Mammografia Monolaterale Sinistra	Screening Tumore Mammella	Non Apprezzabili segni riferibili alla patologia
2764   RX Braccio Destro	Valutazione mezzo di sintesi	Impianto in sede. Rapporti osteoarticolari mantenuti
3381   RX Spalla Destra	Trauma	Assenza di fratture e/o versamenti
3382   RX Avantbraccio Destro	Trauma	Presenza di fratture e versamento circoscritto
3383   RX Polso Destro	Trauma	Presenza di fratture e versamento circoscritto
3384   RX Braccio Destro	Trauma	Presenza di fratture e versamento circoscritto
3385   TC Cranio con e senza mdc	Malattia o Sindrome di Cushing	Non Apprezzabili segni riferibili alla patologia
3443   Ecografia Addome Superiore	Reno Policistico Autosomico Coniante	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista
3573   TC Cranio	Epilessia	Non Apprezzabili segni riferibili alla patologia
4536   RX Endorale	Valutazione mezzo di sintesi	Impianto non in sede. Rapporti osteoarticolari mantenuti
4883   TC Colonna Cervicale	Artrite Ruanoido	Assenza di segni riferibili come causa della patologia
5126   RX Mammografia Monolaterale Sinistra	Prevenzione Tumore Mammella	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista
5137   RM Colonna Cervicale	Anterolistesi	Presenza di anterolistesi di lievi dimensioni. Si consigliano ulteriori approfondimenti diagnostici
5413   TC Cranio con e senza mdc	Merito di Bassow	Non Apprezzabili segni riferibili alla patologia
5581   TC Cranio con e senza mdc	Herbo di Parkinson	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista
6237   RX Mammografia Monolaterale Sinistra	Trauma	Presenza di fratture e versamento diffuso
6279   Ecografia Mammella Monolaterale Destra	Prevenzione Tumore Mammella	Non Apprezzabili segni riferibili alla patologia
6312   Ecografia Tiroside	Tiroidite di Hashimoto	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista
6884   Ecografia Addome Superiore	Lupus	Non Apprezzabili segni riferibili alla patologia
6854   Ecografia Mammella Monolaterale Destra	Prevenzione Tumore Mammella	Non Apprezzabili segni riferibili alla patologia
7790   RM Encelalo con mdc	Anuramia Cerebrale	Presenza di anuramie di lievi dimensioni. Si consigliano ulteriori approfondimenti diagnostici
7857   RM Colonna Cervicale	Cefalea	Presenza di segni riferibili come causa patologica. Si consiglia visione da parte di uno specialista
8181   RM Encelalo con mdc	Anuramia Cerebrale	Presenza di anuramie di severe dimensioni. Si consigliano ulteriori approfondimenti diagnostici

```
| 8222 | Angio-TG Aorta Toracica          | Anurisma Aorta           | Non Apprezzabili segni riferibili alla patologia |  
|  
33 rows in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
33 rows in set (0.02 sec)
```

## 14.5 Ammontare del costo totale per esenzioni

“Si vuole l’ammontare del costo totale per le prestazioni richieste in esenzione erogate nell’anno 2021”

*La query è utile per capire l’ammontare dei costi delle prestazioni per cui i cittadini erano esenti dal pagamento: essendo una struttura pubblica, pertanto convenzionata con il Servizio Sanitario Nazionale, la differenza fra l’imponibile e il netto a liquidare viene coperta dallo Stato.*

```
mysql> SELECT sum(Prestazione.costo) as totale  
    -> FROM PacchettoPrestazionale, Prestazione, Richiesta, Associata  
    -> WHERE Richiesta.codicePrestazione = Prestazione.codicePrestazione  
    -> AND PacchettoPrestazionale.codicePrestazione = Prestazione.codicePrestazione  
    -> AND PacchettoPrestazionale.codiceEsenzione = Associata.codiceEsenzione  
    -> AND Richiesta.idRichiesta = Associata.idRichiesta  
    -> AND year(Richiesta.dataAppuntamento) = 2021;  
+-----+  
| totale |  
+-----+  
| 180814.50 |  
+-----+  
1 row in set (0.04 sec)
```

Tempo dopo inserimento massivo di record:

```
1 row in set (0.28 sec)
```

## 14.6 Mancata disdetta

“Si vogliono sapere gli esami prenotati e non disdetti di ieri, per i quali i pazienti non abbiano fruito la prestazione, mostrando id, nome, cognome del paziente e l’ammontare del costo della ricevuta.”

*La query serve per trovare i pazienti che dovranno corrispondere l’intero importo delle prestazioni.*

```
mysql> SELECT Paziente.idPaziente, Paziente.nome, Paziente.cognome, sum(Prestazione.costo) as totale
-> FROM Richiesta, Paziente, Prestazione
-> WHERE Richiesta.statoRichiesta = 'Prenotato'
-> AND Richiesta.dataAppuntamento = curdate()-1
-> AND Richiesta.nomeDipartimento = 'MC'
-> AND Richiesta.idPaziente = Paziente.idPaziente
-> AND Richiesta.codicePrestazione = Prestazione.codicePrestazione
-> GROUP BY Paziente.idPaziente
-> ORDER BY Paziente.idPaziente;
+-----+-----+-----+
| idPaziente | nome      | cognome   | totale |
+-----+-----+-----+
|      7 | Maurizio | Bembo     |  80.15 |
|     11 | Marta    | Cugia     |  51.15 |
|     12 | Puccio   | Govoni    |  51.15 |
|     15 | Piersanti | Sinisi    |  35.00 |
|     26 | Micheletto| Folliero  |  14.20 |
|     27 | Imelda   | Tamburello|  51.15 |
|     36 | Liana    | Giannotti |  15.49 |
|     40 | Emma     | Barzini   |  49.20 |
|     53 | Livio    | Anichini  | 153.45 |
|     56 | Pina     | Littizzetto|  29.00 |
|     57 | Amico    | Maspero   |  36.15 |
|     67 | Liberto   | Longhena  |  51.15 |
|     68 | Marina   | Renault   |  51.15 |
|     91 | Mariana  | Lippomano |  29.00 |
|     95 | Claudia  | Folliero  |  28.40 |
|     98 | Delfino  | Morricone |  28.40 |
|     99 | Ubaldo   | Marrone   |  51.15 |
|    112 | Michelangelo | Proietti |  51.15 |
|    114 | Lisa     | Cimarosa  |  51.15 |
|   118 | Benito   | Rubbia    |  51.15 |
|   121 | Sandro   | Mersili   |  29.00 |
|   129 | Agostino | Cerutti   |  29.00 |
+-----+-----+-----+
22 rows in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
22 rows in set (0.02 sec)
```

## 14.7 Report Rimborsi

“Si vuole sapere il totale dei pazienti, delle prestazioni e degli importi rimborsati.”

*La query serve a estrapolare i dati da comunicare al reparto che amministra le finanze dell’ospedale per tenere aggiornata la contabilità.*

```
mysql> SELECT count(distinct idPaziente) as totalePazienti, sum(Prestazione.costo) as importo, count(Richiesta.codicePrestazione) as totalePrestazioni
   > FROM Richiesta, Prestazione, Esame
   > WHERE Esame.commento not like 'Paziente rifiuta %'
   > AND Esame.statoEsame = 'Annullato'
   > AND Richiesta.codicePrestazione = Prestazione.codicePrestazione
   > AND Richiesta.idRichiesta = Esame.idRichiesta;
+-----+-----+-----+
| totalePazienti | importo | totalePrestazioni |
+-----+-----+-----+
|      151 | 50012.94 |          1355 |
+-----+-----+-----+
1 row in set (0.03 sec)
```

Tempo dopo inserimento massivo di record:

```
1 row in set (0.19 sec)
```

## 14.8 Dosimetria 1

“Si vuole sapere matricola, nomi, cognomi degli impiegati che abbiano superato la dose soglia annuale stabilita per la propria categoria d'esposizione nell'anno 2021, riportando la dose soglia, la dose totale registrata e di quanto quest'ultima superi la prima.”

*La query è necessaria per la sorveglianza medica degli operatori. Qualora vi siano impiegati la cui dose annua assorbita superi la soglia, avrà luogo un'indagine per poter stabilire quali siano le cause della sovraesposizione (malfunzionamento macchinari, difetti della schermatura delle stanze, negligenza del personale, etc.)*

```
mysql> SELECT Dosimetro.matricola, Impiegato.nome, Impiegato.cognome, sum(Dosimetro.doseRegistrata) as doseRegistrata,
-> CategoriaEsposizione.doseSoglia,
-> sum(Dosimetro.doseRegistrata)-CategoriaEsposizione.doseSoglia as differenzaDose
-> FROM Impiegato, Dosimetro, Qualifica, CategoriaEsposizione
-> WHERE year(Dosimetro.inizioPeriodo) = 2021
-> AND Impiegato.matricola = Dosimetro.matricola
-> AND Impiegato.codiceQualifica = Qualifica.codiceQualifica
-> AND Qualifica.codiceCategoria = CategoriaEsposizione.codiceCategoria
-> GROUP BY Dosimetro.matricola having sum(Dosimetro.doseRegistrata) > CategoriaEsposizione.doseSoglia;
+-----+-----+-----+-----+-----+
| matricola | nome      | cognome   | doseRegistrata | doseSoglia | differenzaDose |
+-----+-----+-----+-----+-----+
|     25 | Emilio    | Sokolov   |       2.75 |    1.00 |      1.75 |
|     26 | Federica  | Porzio    |       1.93 |    1.00 |      0.93 |
|     27 | Atenulif  | Mortati   |       3.30 |    1.00 |      2.30 |
|     28 | Morena    | Salieri   |       2.70 |    1.00 |      1.70 |
|     29 | Giustino  | Solimena  |       3.52 |    1.00 |      2.52 |
|     30 | Bernardo  | Bruno     |       3.40 |    1.00 |      2.40 |
|      1 | Pier       | Morlacchi |       3.71 |    1.00 |      2.71 |
|      2 | Cassandra  | Cabibbo   |       2.35 |    1.00 |      1.35 |
+-----+-----+-----+-----+-----+
8 rows in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
8 rows in set (0.01 sec)
```

## 14.9 Dosimetria 2

“Si vogliono sapere tutti gli impiegati che il mese di Luglio 2021 hanno registrato una dose superiore a quella dell’anno precedente nello stesso mese, rappresentando entrambi i valori”

*La query serve a compilare report statistici sulla sorveglianza medica degli operatori da comunicare agli organi di competenza.*

```
mysql> SELECT x.matricola, x.doseRegistrata as dose2020, y.doseRegistrata as dose2021
-> FROM dosimetro x, dosimetro y
-> WHERE x.matricola = y.matricola
-> AND year(x.inizioPeriodo) = 2020
-> AND month(x.inizioPeriodo) = 7
-> AND year(y.inizioPeriodo) = 2021
-> AND month(y.inizioPeriodo) = month(x.inizioPeriodo)
-> AND y.doseRegistrata > x.doseRegistrata;
+-----+
| matricola | dose2020 | dose2021 |
+-----+
|      2    |   0.14   |   0.28   |
|      3    |   0.60   |   0.63   |
|      4    |   0.40   |   0.86   |
|      6    |   0.03   |   0.46   |
|      7    |   0.01   |   0.39   |
|      9    |   0.48   |   0.58   |
|     10    |   0.46   |   0.86   |
|     11    |   0.28   |   0.42   |
|     14    |   0.57   |   0.65   |
|     15    |   0.66   |   0.79   |
|     16    |   0.25   |   0.41   |
|     21    |   0.56   |   0.70   |
+-----+
12 rows in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
12 rows in set (0.01 sec)
```

## 14.10 Radioprotezione

“Si vogliono sapere id, nome e cognome dei pazienti che hanno sostenuto un esame RX negli ultimi tre giorni(oggi escluso), in cui la dose assorbita supera gli LDR per la prestazione in questione più del 50%(margini di tolleranza), riportando l'id dell'esame, il codice della prestazione, la dose ricevuta, i livelli di dose di riferimento, la matricola dell'impiegato e la sala in cui è stato effettuato.”

*La query è utile per tenere sotto controllo i dati d'esposizione della popolazione. Qualora un paziente riceva una dose superiore ai limiti stabiliti con un margine di tolleranza verrà effettuata un'indagine per capire le cause della sovraesposizione.*

```
mysql> SELECT Paziente.idPaziente, Paziente.nome, Paziente.cognome,
        -> Esame.idEsame, Esame.dosemGycm, Prestazione.LDR, Esame.matricolaTSRM as TSRM,
        -> Richiesta.codicePrestazione as Prestazione, Richiesta.numeroStanza as Stanza, dataOraEsame
        -> FROM Paziente, Richiesta, Esame, Prestazione
        -> WHERE Paziente.idPaziente = Richiesta.idPaziente
        -> AND Richiesta.dataAppuntamento BETWEEN current_date()-3 AND curdate()-1
        -> AND Richiesta.codicePrestazione like 'RX%'
        -> AND Richiesta.idRichiesta = Esame.idRichiesta
        -> AND Richiesta.codicePrestazione = Prestazione.codicePrestazione
        -> AND (Esame.dosemGycm - Prestazione.LDR) > Prestazione.LDR * 0.50;
```

idPaziente   nome	cognome	idEsame   dosemGycm   LDR   TSRM   Prestazione   Stanza   dataOraEsame
90   Morena	Longhena	3774   9.34   5.00   17   RX31SN   2   2021-08-18 12:17:00
90   Morena	Longhena	3775   1.98   1.00   17   RX22DX   2   2021-08-18 13:56:00
64   Roman	Sismondi	3789   332.42   200.00   22   RX19   2   2021-08-18 17:18:00
96   Natalia	Frescobaldi	3791   3.95   2.00   11   RX04DX   2   2021-08-18 15:54:00
86   Gioachino	Bova	3799   1183.06   600.00   18   RX27   2   2021-08-18 18:02:00
46   Lazzaro	Grisoni	6710   3.48   2.00   23   RX03DX   1   2021-08-18 05:06:00
45   Lazzaro	Grisoni	6711   1.97   1.00   23   RX22DX   1   2021-08-18 07:49:00
41   Ruggero	Puglisi	6723   2.84   1.50   11   RX05DX   1   2021-08-18 08:29:00
15   Piersanti	Sinisi	6737   7.63   4.00   24   RX08SN   1   2021-08-18 14:02:00
24   Gelsomina	Zaccagnini	6766   15.49   8.00   17   RX28DX   1   2021-08-18 21:25:00
62   Rosalia	Cuzzocrea	3844   335.55   200.00   17   RX19   2   2021-08-19 12:31:00
62   Rosalia	Cuzzocrea	3845   8.63   5.00   17   RX07SN   2   2021-08-19 12:33:00
45   Lazzaro	Grisoni	3848   393.65   200.00   20   RX19   2   2021-08-19 12:39:00
12   Serafina	Botticelli	3882   1.80   1.00   23   RX22DX   2   2021-08-19 18:02:00
91   Mariana	Lippomano	3883   8.84   5.00   22   RX29   2   2021-08-19 17:52:00
4   Luciana	Gianetti	3898   3.92   2.00   11   RX04SN   2   2021-08-19 19:23:00
94   Emma	Cilea	6866   305.66   200.00   22   RX19   1   2021-08-19 03:58:00
118   Benito	Rubbia	6820   3.81   2.00   13   RX03DX   1   2021-08-19 06:43:00
118   Benito	Rubbia	6822   3.39   2.00   13   RX03SN   1   2021-08-19 09:16:00
122   Livia	Mancini	6838   3.52   2.00   21   RX03DX   1   2021-08-19 11:53:00
76   Alfio	Pizzoli	6837   7.50   4.00   18   RX08DX   1   2021-08-19 12:36:00
116   Licia	Magnani	6846   354.98   200.00   18   RX19   1   2021-08-19 15:16:00
66   Valentina	Mosiglia	6849   2.56   1.50   20   RX05DX   1   2021-08-19 13:41:00

	138   Pina	Nordio		6852	3.87	2.00	14	RX04DX		1	2021-08-19 16:48:00
	138   Pina	Nordio		6853	1.52	1.00	14	RX10DX		1	2021-08-19 15:16:00
	96   Natalia	Frescobaldi		6857	3.18	2.00	12	RX03DX		1	2021-08-19 17:50:00
	132   Adrie	Venditti		6872	3.48	2.00	17	RX09DX		1	2021-08-19 20:42:00
	23   Isa	Ligorio		6877	3.19	2.00	16	RX04SN		1	2021-08-19 22:41:00
	4   Luciana	Gianetti		3899	1.74	1.00	11	RX18SN		2	2021-08-20 08:58:00
	59   Rosa	Cossiga		3963	13.01	8.00	16	RX28SN		2	2021-08-20 10:14:00
	37   Massimiliano	Pavarotti		3919	6.64	4.00	15	RX08DX		2	2021-08-20 12:00:00
	101   Simone	Gualandi		3967	1.67	1.00	19	RX18SN		2	2021-08-20 21:53:00
	27   Imelda	Tamburello		6891	3.34	2.00	18	RX04SN		1	2021-08-20 04:53:00
	111   Marina	Armani		6910	4.98	3.00	13	RX21DX		1	2021-08-20 10:01:00
	111   Marina	Armani		6911	6.02	4.00	13	RX08DX		1	2021-08-20 10:53:00
	111   Marina	Armani		6912	5.87	3.00	13	RX21SN		1	2021-08-20 12:27:00
	111   Marina	Armani		6915	2.94	1.60	13	RX05SN		1	2021-08-20 13:49:00
	11   Marta	Cugia		6923	1.62	1.00	11	RX22SN		1	2021-08-20 16:42:00

38 rows in set (0.01 sec)

## Tempo dopo inserimento massivo di record:

38 rows in set (0.02 sec)

## 14.11 Non idoneità alla somministrazione di contrasto

“Si vogliono sapere quali esami siano stati annullati per mancanza di analisi del sangue recenti o non idoneità alla somministrazione del mezzo di contrasto negli ultimi 3 giorni, mostrando l'id del paziente e il numero di telefono.”

*La query serve poiché in questo caso l'esame deve essere riprogrammato.*

```
mysql> SELECT Esame.idEsame, Paziente.idPaziente, Paziente.telefono
   > FROM Esame, Paziente, Richiesta
   > WHERE Esame.statoEsame = 'Annullato'
   > AND Esame.commento like '%analisi%mdc'
   > AND Richiesta.dataAppuntamento BETWEEN current_date()-3 AND current_date()
   > AND Paziente.idPaziente = Richiesta.idPaziente
   > AND Richiesta.idRichiesta = Esame.idRichiesta;
+-----+-----+-----+
| idEsame | idPaziente | telefono |
+-----+-----+-----+
| 3822 | 99 | 1282327 |
| 4451 | 105 | 1523188 |
| 4481 | 70 | 1198598 |
| 4648 | 96 | 1728790 |
| 5838 | 81 | 1399566 |
| 6111 | 84 | 1026036 |
| 6112 | 84 | 1026036 |
| 6375 | 14 | 1113468 |
| 6377 | 30 | 1639946 |
| 6398 | 73 | 1755791 |
| 6502 | 25 | 1173105 |
| 6514 | 108 | 1377398 |
| 6516 | 46 | 1946069 |
| 6578 | 110 | 1617776 |
| 6611 | 5 | 1154252 |
| 6639 | 24 | 1451492 |
| 6682 | 84 | 1026036 |
| 6712 | 45 | 1811927 |
| 6782 | 38 | 1463604 |
| 6819 | 118 | 1281755 |
| 6875 | 132 | 1569925 |
| 8372 | 21 | 1687083 |
| 8614 | 104 | 1525715 |
| 8623 | 113 | 1736834 |
| 8654 | 59 | 1815237 |
| 8940 | 136 | 1169524 |
| 8948 | 15 | 1689200 |
| 9014 | 84 | 1026036 |
| 9038 | 153 | 1950705 |
| 9058 | 130 | 1725387 |
| 9069 | 93 | 1928659 |
```

```

| 9078 |      143 | 1624883 |
| 9087 |      132 | 1509925 |
+-----+
33 rows in set (0.01 sec)

```

Tempo dopo inserimento massivo di record:

```
33 rows in set (0.02 sec)
```

## 14.12 Possibile Contagio

“Si vogliono sapere id dei pazienti, con nome, cognome e telefono, che hanno sostenuto un esame il ’19-08-2021’ successivamente all’esame del paziente con id 45 e nelle stesse stanze dove sono avvenuti i suoi esami (comprendere anche esami annullati).”

*La query può essere utile nel caso in cui il dipartimento venga informato di una patologia infettiva del paziente e si vogliano avvisare i pazienti possibilmente coinvolti in un contagio.*

```

mysql> SELECT x.idPaziente, Paziente.nome, Paziente.cognome, Paziente.telefono
-> FROM Paziente, Richiesta x, Richiesta y, Esame w, Esame z
-> WHERE y.idPaziente = 45
-> AND y.numeroStanza = x.numeroStanza
-> AND y.idPaziente != x.idPaziente
-> AND y.idRichiesta = w.idRichiesta
-> AND date(w.dataOraEsame) = '2021-08-19'
-> AND date(z.dataOraEsame) = date(w.dataOraEsame)
-> AND time(w.dataOraEsame) < time(z.dataOraEsame)
-> AND z.idRichiesta = x.idRichiesta
-> AND x.idPaziente = Paziente.idPaziente
-> GROUP BY x.idPaziente;
+-----+-----+
| idPaziente | nome     | cognome   | telefono |
+-----+
|    75 | Griselda | Parisi    | 1577828 |
|    74 | Luchino   | Impastato  | 1799147 |
|    70 | Lilla     | Riccati    | 1198598 |
|    13 | Serafina  | Botticelli | 1551428 |
|    91 | Mariana   | Lippomano  | 1378901 |
|     4 | Luciana   | Gianetti   | 1310717 |
+-----+
6 rows in set (0.01 sec)

```

Tempo dopo inserimento massivo di record:

```
6 rows in set (0.13 sec)
```

## 14.13 Valutazione di produttività

“Si vuole sapere matricola, nomi, cognomi degli impiegati che hanno eseguito meno esami del mese precedente, coi rispettivi totali”

*La query è utile a stimare la produttività del reparto.*

```
mysql> SELECT Impiegato.matricola, Impiegato.nome, Impiegato.cognome, count(DISTINCT IF( month(dataOraEsame) = month(curdate())-1, Esame.idEsame, null)) mese1,
```

```
>>> count(DISTINCT IF(month(dataOraEsame) = month(curdate()), Esame.idEsame, null)) mese2
-> FROM Impiegato, Esame, Richiesta
-> WHERE Esame.statoEsame != 'Annullato'
-> AND Esame.idRichiesta = Richiesta.idRichiesta
-> AND year(dataOraEsame) = 2021
-> AND Esame.matricolaTSRM = Impiegato.matricola
-> GROUP BY Impiegato.matricola
-> HAVING mese1 > mese2;
```

matricola	nome	cognome	mese1	mese2
11	Eva	Federico	330	223
12	Virginia	Chinnici	297	201
13	Lauretta	Casalodi	296	216
14	Maria	Pincherle	309	208
15	Mariana	Colletti	318	196
16	Virginia	Caracciolo	350	195
17	Natalia	Marino	278	209
18	Antonietta	Camanni	336	187
19	Marta	Antonelli	279	205
20	Morena	Ossola	327	193
21	Aria	Pavone	318	204
22	Orazio	Morucci	317	215
23	Caterina	Ceravolo	341	210
24	Benvenuto	Troisi	324	201

14 rows in set (0.04 sec)

Tempo dopo inserimento massivo di record:

14 rows in set (0.22 sec)

## 14.14 Valutazione di efficienza

“Si vuole sapere quale sia il tempo di refertazione medio degli esami svolti nel mese di Luglio 2021”

*La query è utile a stimare l'efficienza dei lavoratori nel reparto.*

```
mysql> SELECT avg(timestampdiff(day, Esame.dataOraEsame, Referta.dataOraReferito)) as mediaInGiorni
-> FROM Referta, Esame
-> WHERE year(dataOraEsame)= 2021
-> AND month(dataOraEsame) = 7
-> AND Esame.idEsame = Referta.idEsame;
+-----+
| mediaInGiorni |
+-----+
|      3.7585 |
+-----+
1 row in set (0.02 sec)
```

Tempo dopo inserimento massivo di record:

```
1 row in set (0.32 sec)
```

## 14.15 Gestione Emergenze

“Si vogliono sapere il tempo medio con cui sono state soddisfatte le richieste di codice rosso dal Pronto Soccorso nel 2021.”

*La query è utile a valutare l'efficienza del reparto nella gestione delle emergenze.*

```
mysql> SELECT sec_to_time(avg(timestampdiff(second, timestamp(Richiesta.dataOraRichiesta), timestamp(Esame.dataOraEsame)))) as media
-> FROM Richiesta, Esame
-> WHERE Richiesta.nomeDipartimento = 'PS'
-> AND Richiesta.priorita = 3
-> AND year(Richiesta.dataOraRichiesta) = 2021
-> AND Esame.idRichiesta = Richiesta.idRichiesta;
+-----+
| media      |
+-----+
| 02:56:42.8571 |
+-----+
1 row in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
1 row in set (0.06 sec)
```

## 14.16 Indagine Statistica 1

“Si vuole sapere quanti pazienti, a oggi, della regione Puglia hanno effettuato una richiesta con un’esonzione presso la struttura”

*La query è utile per indagini che comparino la natura, la qualità e l’unicità dei servizi che vengono forniti rispetto a strutture pubbliche di un’altra regione.*

```
mysql> SELECT count(Associata.idRichiesta) totale
-> FROM Associata, Richiesta, Paziente, Zona, ComuneResidenza
-> WHERE ComuneResidenza.Regione = 'Puglia'
-> AND Zona.comune = ComuneResidenza.comune
-> AND Paziente.cap = Zona.cap
-> AND Richiesta.idPaziente = Paziente.idPaziente
-> AND Richiesta.idRichiesta = Associata.idRichiesta;
+-----+
| totale |
+-----+
|    940 |
+-----+
1 row in set (0.02 sec)
```

Tempo dopo inserimento massivo di record:

```
1 row in set (0.52 sec)
```

## 14.17 Indagine Statistica 2

“Si vogliono sapere le percentuali di richiesta di pazienti provenienti dalle diverse regioni, divisi per regione.”

*La query serve a quantificare l'affluenza di pazienti da varie regioni per generare statistiche utili a comparare la natura, la qualità e l'unicità dei servizi che vengono forniti rispetto a strutture pubbliche di altre regioni.*

```
mysql> SELECT ComuneResidenza.regione, count(idRichiesta)*100/richieste.totale percentuale_per_regione
   > FROM (SELECT count(idRichiesta) totale FROM Richiesta) as richieste,
   > Richiesta, Paziente, Zona, ComuneResidenza
   > WHERE Richiesta.idPaziente = Paziente.idPaziente
   > AND Paziente.cap = Zona.cap
   > AND Zona.comune = ComuneResidenza.comune
   > GROUP BY ComuneResidenza.regione;
+-----+-----+
| regione | percentuale_per_regione |
+-----+-----+
| Puglia | 16.0127 |
| Emilia Romagna | 15.8489 |
| Toscana | 12.1159 |
| Lombardia | 10.1566 |
| Campania | 12.1323 |
| Sicilia | 11.8548 |
| Lazio | 12.3997 |
| Piemonte | 9.4799 |
+-----+
8 rows in set (0.02 sec)
```

Tempo dopo inserimento massivo di record:

```
8 rows in set (0.81 sec)
```

## 14.18 Indagine statistica 3

“Si vogliono sapere le 10 prestazioni più richieste, riportando il nome della prestazione e il numero di richieste”

*La query è utile a fini statistici.*

```
mysql> SELECT Prestazione.nomePrestazione, count(Richiesta.idRichiesta) numeroRichieste
   -> FROM Prestazione, richiesta
   -> WHERE Prestazione.codicePrestazione = Richiesta.codicePrestazione
   -> GROUP BY Richiesta.codicePrestazione
   -> ORDER BY count(Richiesta.idRichiesta) desc limit 10;
+-----+-----+
| nomePrestazione | numeroRichieste |
+-----+-----+
| TC Cervello con e senza mdc | 1595 |
| Ecografia Addome Superiore | 1278 |
| RM Colonna Cervicale | 1007 |
| TC Cranio | 875 |
| RX Endorale | 745 |
| RM Encefalo con mdc | 606 |
| Eco-color-doppler-grafia Cardiaca | 571 |
| Ecografia Tiroide | 541 |
| RX Mammografia Bilaterale | 405 |
| RX Ortopanoramica | 397 |
+-----+-----+
10 rows in set (0.03 sec)
```

Tempo dopo inserimento massivo di record:

```
10 rows in set (0.33 sec)
```

## 14.19 Esami per patologie croniche

“Si vuole sapere id, nomi, cognomi dei pazienti che hanno usufruito di un’esonzione per Artrite Reumatoide e il numero di esami effettuati per ognuno”

*La query è utile a generare statistiche sulle patologie croniche che affliggono la popolazione.*

```
mysql> SELECT Paziente.idPaziente, Paziente.nome, Paziente.cognome, count(idEsame) totaleEsami
       -> FROM Richiesta, Paziente, Esame, Associaata, Esenzione
       -> WHERE Eserzione.condizione = 'Artrite Reumatoide'
       -> AND Esame.statoEsame != 'ANNULLATO'
       -> AND Richiesta.idPaziente = Paziente.idPaziente
       -> AND Associaata.codiceEsenzione = Esenzione.codiceEsenzione
       -> AND Esame.idRichiesta = Richiesta.idRichiesta
       -> AND Associaata.idRichiesta = Richiesta.idRichiesta
       -> GROUP BY Paziente.idPaziente;
+-----+-----+-----+
| idPaziente | nome   | cognome | totaleEsami |
+-----+-----+-----+
|      22 | Imelda | Soranzo |          1 |
+-----+-----+-----+
1 row in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
29 rows in set (0.01 sec)
```

## 14.20 Indagine demografica

“Si vogliono sapere tutti gli id, nome, cognome, età e genere dei pazienti di età compresa fra i 30 e i 60 anni che vivono nella stessa zona del paziente 103 per cui siano stati richiesti RX del torace per sospetta BPCO.”

*La query è utile per l'indagine sulle cause di una certa patologia.*

```
mysql> SELECT x.idPaziente, x.nome, x.cognome, year(current_date()) - year(x.dataNascita) età, x.genere
-> FROM Paziente x, Paziente y, Richiesta, Prestazione
-> WHERE year(current_date())- year(x.dataNascita) >= 30
-> AND year(current_date())- year(x.dataNascita) <= 60
-> AND y.idPaziente = 103
-> AND y.cap = x.cap
-> AND Richiesta.idPaziente = x.idPaziente
-> AND Richiesta.quesitoDiagnostico = 'BPCO'
-> AND Richiesta.codicePrestazione = Prestazione.codicePrestazione
-> AND Prestazione.nomePrestazione = 'RX Torace'
-> AND y.idPaziente != x.idPaziente
-> GROUP BY x.idPaziente;
+-----+-----+-----+-----+
| idPaziente | nome      | cognome   | età    | genere |
+-----+-----+-----+-----+
|       26 | Micheletto | Folliero |   45 | M     |
+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

Tempo dopo inserimento massivo di record:

```
7 rows in set (0.01 sec)
```

# 15 Query in algebra relazionale

## 15.1 Storico Esami

“Si vuole sapere lo storico degli esami del paziente con id 55. In particolare si vuole sapere il nome della prestazione, il quesito diagnostico e il referto.”

$$\begin{aligned} & \pi_{idEsame, nomePrestazione, quesitoDiagnostico, referto} ( \\ & \quad PRESTAZIONE | X | codicePrestazione = codicePrestazione (( \\ & \quad ESAME | X | idEsame = idEsame REFERTA) | X | idPaziente = idPaziente ( \\ & \quad \sigma_{idPaziente=55} (RICHIEDA))) ) \end{aligned}$$

## 15.2 Dosimetria 2

“Si vogliono sapere tutti gli impiegati che il mese di Luglio 2021 hanno registrato una dose superiore a quella dell’anno precedente nello stesso mese, rappresentando entrambi i valori”

$$\begin{aligned} & \pi_{xmatricola, xdoseRegistrata, ydoseRegistrata} ( \\ & \rho_{xidDosimetro, xmatricola, xdoseRegistrata, xinizioPeriodo, xfinePeriodo \leftarrow (} \\ & \quad idDosimetro, matricola, doseRegistrata, inizioPeriodo, finePeriodo \\ & \quad \sigma_{year(inizioPeriodo=2020 \text{ and } month(inizioPeriodo=7))} ( \\ & \quad DOSIMETRO) | X | ydoseRegistrata > xdoseRegistrata ( \\ & \quad \rho_{yidDosimetro, ymatricola, ydoseRegistrata, yinizioPeriodo, yfinePeriodo \leftarrow (} \\ & \quad idDosimetro, matricola, doseRegistrata, inizioPeriodo, finePeriodo \\ & \quad \sigma_{year(inizioPeriodo=2021 \text{ and } month(inizioPeriodo=7))} (DOSIMETRO)))) \end{aligned}$$

# 16 Ottimizzazione Query

Per ottimizzare le prestazioni del database in prima istanza nella formulazione delle query è stato evitato l'uso di select nidificate, usando invece query con alias dove possibile e select derived;

A seguire verrà presentato un esempio di ottimizzazione.

## 16.1 Disponibilità

In questo caso per ottimizzare i tempi di esecuzione è stato preso in considerazione il piano di esecuzione dell'ottimizzatore.

La query dopo l'inserimento di oltre 200000 record riporta il seguente risultato:

*(Per ottenere risultati validi prima di eseguire ogni query è stato riavviato il server in modo da svuotare la "query\_cache")*

```
mysql> SELECT data, ora, Sala.numeroStanza
   > FROM Calendario INNER JOIN Orario INNER JOIN Sala
   > LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta
   > WHERE statoRichiesta != 'Cancellato') AS agenda
   > ON agenda.dataAppuntamento = Calendario.data
   > AND agenda.oraAppuntamento = Orario.ora
   > AND agenda.numeroStanza = Sala.numeroStanza
   > WHERE Sala.Riservata = 'Esterni'
   > AND Sala.metodoImaging = 'RX'
   > AND timestamp(data, ora) >= current_timestamp()
   > AND ora >= '8:00'
   > AND ora <= '19:00'
   > AND Festivo(data) = 0
   > AND agenda.idRichiesta is null
   > ORDER BY data, ora
   > LIMIT 5;

+-----+-----+-----+
| data      | ora      | numeroStanza |
+-----+-----+-----+
| 2021-08-21 | 09:00:00 |          2 |
| 2021-08-21 | 10:00:00 |          2 |
| 2021-08-21 | 10:30:00 |          2 |
| 2021-08-21 | 12:00:00 |          2 |
| 2021-08-21 | 13:30:00 |          2 |
+-----+-----+-----+
5 rows in set (1.46 sec)
```

Chiedendo di mostrare tutti i risultati della tabella si ottengono i seguenti tempi di esecuzione:

```
mysql> SELECT data, ora, Sala.numeroStanza
   > FROM Calendario INNER JOIN Orario INNER JOIN Sala
   > LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta
   > WHERE statoRichiesta != 'Cancellato') AS agenda
   > ON agenda.dataAppuntamento = Calendario.data
   > AND agenda.oraAppuntamento = Orario.ora
   > AND agenda.numeroStanza = Sala.numeroStanza
   > WHERE Sala.Riservata = 'Esterini'
   > AND Sala.metodoImaging = 'RX'
   > AND timestamp(data, ora) >= current_timestamp()
   > AND ora >= '8:00'
   > AND ora <= '19:00'
   > AND AND Festivo(data) = 0
   > AND agenda.idRichiesta is null
   > ORDER BY data, ora;
+-----+-----+-----+
| data      | ora       | numeroStanza |
+-----+-----+-----+
| 2021-08-21 | 09:00:00 |      2 |
| 2021-08-21 | 10:00:00 |      2 |
| 2021-08-21 | 10:30:00 |      2 |
| 2021-08-21 | 12:00:00 |      2 |
| 2021-08-21 | 13:30:00 |      2 |
| 2021-08-21 | 14:30:00 |      2 |
|
|
|
| 2022-08-19 | 13:30:00 |      2 |
| 2022-08-19 | 14:00:00 |      2 |
| 2022-08-19 | 16:00:00 |      2 |
| 2022-08-19 | 16:30:00 |      2 |
| 2022-08-19 | 17:30:00 |      2 |
| 2022-08-19 | 18:30:00 |      2 |
+-----+-----+-----+
4838 rows in set (1.48 sec)
```

Usando il comando explain otteniamo il seguente risponso:

```
mysql> EXPLAIN SELECT data, ora, Sala.numeroStanza
   > FROM Calendario INNER JOIN Orario INNER JOIN Sala
   > LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta
   > WHERE statoRichiesta != 'Cancellato') AS agenda
   > ON agenda.dataAppuntamento = Calendario.data
```

## RADB

```
--> AND agenda.oraAppuntamento = Orario.ora
--> AND agenda.numeroStanza = Sala.numeroStanza
--> WHERE Sala.Riservata = 'Esterni'
--> AND Sala.metodoImaging = 'RX'
--> AND timestamp(data, ora) >= current_timestamp()
--> AND ora >= '8:00'
--> AND ora <= '19:00'
--> AND Festivo(data) = 0
--> AND agenda.idRichiesta is null
--> ORDER BY data, ora;
```

id   select_type   table   partitions   type   possible_keys	key   key_len   ref
1   SIMPLE   Sala   NULL   ALL   NULL	NULL   NULL   NULL
1   SIMPLE   Orario   NULL   range   PRIMARY	PRIMARY   3   NULL
1   SIMPLE   Calendario   NULL   index   NULL	PRIMARY   3   NULL
1   SIMPLE   Richiesta   NULL   <b>ref</b>   dataAppuntamento,OraAppuntamento,numeroStanza	dataAppuntamento   3   radb2.Calendario.data

rows   filtered   Extra
10   10.00   Using where; Using temporary; Using filesort
23   100.00   Using where; Using index; Using join buffer (hash join)
1146   100.00   Using where; Using index; Using join buffer (hash join)
232   10.00   Using where

4 rows in set, 1 warning (0.02 sec)

Osserviamo che per la tabella Richiesta viene eseguita una ricerca con indice “dataAppuntamento”. Poiché comunque verrà scandita tutta la tabella forziamo l’ottimizzatore a ignorare gli indici “dataAppuntamento” e “oraAppuntamento” con un hint:

```
mysql> SELECT data, ora, Sala.numeroStanza
--> FROM Calendario INNER JOIN Orario INNER JOIN Sala
--> LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta /*!ignore index(dataAppuntamento,
oraAppuntamento)/*)
--> WHERE statoRichiesta != 'Cancellato' AS agenda
--> ON agenda.dataAppuntamento = Calendario.data
--> AND agenda.oraAppuntamento = Orario.ora
--> AND agenda.numeroStanza = Sala.numeroStanza
--> WHERE Sala.Riservata = 'Esterni'
--> AND Sala.metodoImaging = 'RX'
--> AND timestamp(data, ora) >= current_timestamp()
--> AND ora >= '8:00'
--> AND ora <= '19:00'
--> AND Festivo(data) = 0
```

```

-> AND agenda.idRichiesta is null
-> ORDER BY data, ora
-> LIMIT 5;

+-----+-----+-----+
| data      | ora       | numerostanza |
+-----+-----+-----+
| 2021-08-21 | 09:00:00 |      2 |
| 2021-08-21 | 10:00:00 |      2 |
| 2021-08-21 | 10:30:00 |      2 |
| 2021-08-21 | 12:00:00 |      2 |
| 2021-08-21 | 13:30:00 |      2 |
+-----+-----+-----+
5 rows in set (0.48 sec)

```

```

mysql> SELECT data, ora, Sala.numerostanza
-> FROM Calendario INNER JOIN Orario INNER JOIN Sala
-> LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta /*ignore index(dataAppuntamento,
oraAppuntamento)*/)
-> WHERE statoRichiesta != 'Cancellato') AS agenda
-> ON agenda.dataAppuntamento = Calendario.data
-> AND agenda.oraAppuntamento = Orario.ora
-> AND agenda.numeroStanza = Sala.numeroStanza
-> WHERE Sala.Riservata = 'Esterini'
-> AND Sala.metodoImaging = 'RX'
-> AND timestamp(data, ora) >= current_timestamp()
-> AND ora >= '08:00'
-> AND ora <= '19:00'
-> AND Festivo(data) = 0
-> AND agenda.idRichiesta is null
-> ORDER BY data, ora;

+-----+-----+-----+
| data      | ora       | numerostanza |
+-----+-----+-----+
| 2021-08-21 | 09:00:00 |      2 |
| 2021-08-21 | 10:00:00 |      2 |
| 2021-08-21 | 10:30:00 |      2 |
| 2021-08-21 | 12:00:00 |      2 |
| 2021-08-21 | 13:30:00 |      2 |
| 2021-08-21 | 14:30:00 |      2 |
|
|
|
| 2022-08-19 | 13:30:00 |      2 |
| 2022-08-19 | 14:00:00 |      2 |
| 2022-08-19 | 16:00:00 |      2 |

```

## RADB

```
| 2022-08-19 | 16:30:00 |      2 |
| 2022-08-19 | 17:30:00 |      2 |
| 2022-08-19 | 18:30:00 |      2 |
+-----+
4838 rows in set (0.53 sec)
```

Notiamo un miglioramento dei tempi d'esecuzione. Di seguito il piano d'esecuzione che è stato forzato a eseguire con l'hint /\*!ignore index (dataAppuntamento, oraAppuntamento)\*/

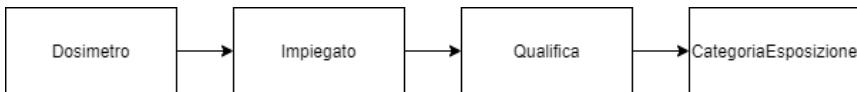
```
mysql> EXPLAIN SELECT data, ora, Sala.numerostanza
   > FROM Calendario INNER JOIN Orario INNER JOIN Sala
   > LEFT JOIN (SELECT idRichiesta, dataAppuntamento, oraAppuntamento, numeroStanza FROM Richiesta /*!ignore index(dataAppuntamento,
oraAppuntamento)*/
   > WHERE statoRichiesta != 'Cancellato') AS agenda
   > ON agenda.dataAppuntamento = Calendario.data
   > AND agenda.oraAppuntamento = Orario.ora
   > AND agenda.numeroStanza = Sala.numeroStanza
   > WHERE Sala.Riservata = 'Esterni'
   > AND Sala.metodoImaging = 'RX'
   > AND timestamp(data, ora) >= current_timestamp()
   > AND ora >= '8:00'
   > AND ora <= '19:00'
   > AND Festivo(data) = 0
   > AND agenda.idRichiesta is null
   > ORDER BY data, ora;
+-----+
| id | select_type | table      | partitions | type    | possible_keys | key     | key_len | ref   |
+-----+
|  1 | SIMPLE      | Sala        | NULL       | ALL     | NULL         | NULL    | NULL    | NULL  |
|  1 | SIMPLE      | Orario      | NULL       | range   | PRIMARY      | PRIMARY | 3      | NULL  |
|  1 | SIMPLE      | Calendario | NULL       | index   | NULL         | PRIMARY | 3      | NULL  |
|  1 | SIMPLE      | Richiesta | NULL       | ALL    | numeroStanza | NULL    | NULL    | NULL  |
+-----+
-----+
rows | filtered | Extra
-----+
 10 |    10.00 | Using where; Using temporary; Using filesort          |
 23 |  100.00 | Using where; Using index; Using join buffer (hash join) |
1146 |  100.00 | Using where; Using index; Using join buffer (hash join) |
283517 |    10.00 | Using where; Using join buffer (hash join)           |
-----+
4 rows in set, 1 warning (0.01 sec)
```

# 17 MongoDB

Per eseguire il confronto con MongoDB sono stati scelti due rami del database e sono state create due Collection in Mongo. Per le Collection è stato scelto di inserire dati denormalizzati.

## 17.1 Ramo Dosimetria

Il ramo scelto è strutturato nello schema fisico nel seguente modo:



per strutturare il documento denormalizzato in MongoDB è stato scelto di inglobare in CategoriaEsposizione tutte le qualifiche inquadrate in ogni categoria come sotto documento, in Qualifica tutti gli impiegati inquadrati in ogni titolo professionale come sotto documento, in Impiegato tutti i propri dosimetri come sotto documento. La struttura del documento risultante è la seguente:

```
{
    "_id" : ObjectId("61221955ab030404b7f881d6"),
    "codiceCategoria" : "C",
    "doseSoglia" : 1,
    "descrizione" : "Supera 1 mSv dose efficace in un anno solare",
    "inquadramentoInCategoria" : [
        {
            "_id" : ObjectId("61221954c013e1dba9a6bc82"),
            "codiceQualifica" : "AS",
            "codiceCategoria" : "C",
            "descrizione" : "Amministratore Sistema",
            "qualificati" : [
                {
                    "_id" : ObjectId("61221954a6a8124554553919"),
                    "matricola" : 2,
                    "nome" : "Cassandra",
                    "cognome" : "Cabibbo",
                    "dataNascita" : "1958-09-26",
                    "cittaNascita" : "Bari",
                    "codiceFiscale" : "CBBCSN58P66A662A",
                    "indirizzo" : "Canale Calcedonio, 35",
                    "cap" : 118,
                    "telefono" : "1639998",
                    "codiceQualifica" : "AS",
                }
            ]
        }
    ]
}
```

## RADB

```
    "username" : "CabibboCassandra@localhost",
    "dosimetri" : [
        {
            "_id" : ObjectId("61221954aa132535abb6b18a"),
            "idDosimetro" : 20202,
            "matricola" : 2,
            "doseRegistrata" : 0.48,
            "inizioPeriodo" : ISODate("2020-01-01T00:00:00Z"),
            "finePeriodo" : ISODate("2020-01-31T00:00:00Z")
        },
        ...
    ]
```

A seguire alcuni esempi di possibili query da eseguire sul ramo.

### 17.1.1 Categoria d'appartenenza

“Si vogliono sapere quali siano le categorie d'appartenenza degli Impiegati dall'11 al 30 (numero di matricola), con la descrizione e la dose soglia.”

```
> db.CategoriaEsposizione.find({"inquadramentoInCategoria.qualificati.matricola":{$gte:11, $lte:30},  
{"inquadramentoInCategoria":0}).pretty()  
{  
    "_id" : ObjectId("61221955ab030404b7f881d6"),  
    "codiceCategoria" : "C",  
    "doseSoglia" : 1,  
    "descrizione" : "Supera 1 mSv dose efficace in un anno solare"  
}  
  
{  
    "_id" : ObjectId("61221955ab030404b7f881d7"),  
    "codiceCategoria" : "A",  
    "doseSoglia" : 20,  
    "descrizione" : "Supera 6 mSv dose efficace in un anno solare"  
}
```

La rispettiva query in MySQL è:

```
mysql> SELECT CategoriaEsposizione.codiceCategoria, CategoriaEsposizione.doseSoglia, CategoriaEsposizione.descrizione  
        -> FROM CategoriaEsposizione, Qualifica, Impiegato  
        -> WHERE Impiegato.matricola BETWEEN 11 AND 30  
        -> AND CategoriaEsposizione.codiceCategoria = Qualifica.codiceCategoria  
        -> AND Qualifica.codiceQualifica = Impiegato.codiceQualifica  
        -> GROUP BY CategoriaEsposizione.codiceCategoria;  
+-----+-----+-----+  
| codiceCategoria | doseSoglia | descrizione          |  
+-----+-----+-----+  
| A             |     20.00 | Supera 6 mSv dose efficace in un anno solare |  
| C             |      1.00 | Supera 1 mSv dose efficace in un anno solare |  
+-----+-----+-----+  
2 rows in set (0.02 sec)
```

## 17.1.2 Inquadramento in Categoria

“Si vogliono sapere le matricole degli impiegati inquadrati in categoria C con la rispettiva qualifica.”

```
> db.CategoriaEsposizione.find({“codiceCategoria”:”C”},
{“codiceCategoria”:1,”inquadramentoInCategoria.codiceQualifica”:1,”inquadramentoInCategoria.qualificati.matricola”:1}).pretty()
{
  “_id” : ObjectId(“6122195ab030404b7f801d6”),
  “codiceCategoria” : “C”,
  “inquadramentoInCategoria” : [
    {
      “codiceQualifica” : “AS”,
      “qualificati” : [
        {
          “matricola” : 2
        },
        {
          “matricola” : 1
        }
      ]
    },
    {
      “codiceQualifica” : “AMM”,
      “qualificati” : [
        {
          “matricola” : 25
        },
        {
          “matricola” : 26
        },
        {
          “matricola” : 27
        },
        {
          “matricola” : 28
        },
        {
          “matricola” : 29
        },
        {
          “matricola” : 30
        }
      ]
    }
  ]
}
```

La rispettiva query in MySQL è:

```
mysql> SELECT Impiegato.matricola, Impiegato.codiceQualifica
-> FROM CategoriaEsposizione, Qualifica, Impiegato
-> WHERE CategoriaEsposizione.codiceCategoria = 'C'
-> AND CategoriaEsposizione.codiceCategoria = Qualifica.codiceCategoria
-> AND Qualifica.codiceQualifica = Impiegato.codiceQualifica;
+-----+-----+
| matricola | codiceQualifica |
+-----+-----+
|      25 | AMM          |
|      26 | AMM          |
|      27 | AMM          |
|      28 | AMM          |
|      29 | AMM          |
|      30 | AMM          |
|       1 | AS           |
|       2 | AS           |
+-----+-----+
8 rows in set (0.1 sec)
```

### 17.1.3 Dosimetria 1

“Si vuole sapere matricola, nomi, cognomi degli impiegati che abbiano superato la dose soglia annuale stabilita per la propria categoria d'esposizione nell'anno 2021, riportando la dose soglia la dose totale registrata e di quanto quest'ultima superi la prima.”

```
> db.CategoriaEsposizione.aggregate([
... {$unwind:{path: "$inquadramentoInCategoria"}},
... {$unwind:{path: "$inquadramentoInCategoria.qualificati"}},
... {$unwind:{path: "$inquadramentoInCategoria.qualificati.dosimetri"}},
... {$match:{"inquadramentoInCategoria.qualificati.dosimetri.inizioPeriodo":
...     {$gte:ISODate('2021-01-01T00:00:00Z'),$lt:ISODate('2022-01-01T00:00:00Z')}},
... {$group:{_id: {"matricolaImp": "$inquadramentoInCategoria.qualificati.dosimetri.matricola",
...     "nomeImp": "$inquadramentoInCategoria.qualificati.nome",
...     "cognomeImp": "$inquadramentoInCategoria.qualificati.cognome",
...     "doseSogliaImp": "$doseSoglia"}, 
...     "totDoseRegistrata":{$sum:"$inquadramentoInCategoria.qualificati.dosimetri.doseRegistrata"}},
... {$project:{"totDoseRegistrata":1,"doseDiff":{$subtract:[ "$totDoseRegistrata", "$_id.doseSogliaImp"]}}},
... {$match: {"doseDiff": {$gt: 0}}},
... {$sort:{_id.matricolaImp:1}},
... ]).pretty()
{
    "_id" : {
        "matricolaImp" : 1,
        "nomeImp" : "Pier",
        "cognomeImp" : "Morlacchi",
```

## RADB

```
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 3.71,
    "doseDiff" : 2.71
}
{
    "_id" : {
        "matricolaImp" : 2,
        "nomeImp" : "Cassandra",
        "cognomeImp" : "Cabibbo",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 2.35,
    "doseDiff" : 1.35
}
{
    "_id" : {
        "matricolaImp" : 25,
        "nomeImp" : "Emilio",
        "cognomeImp" : "Sokolov",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 2.75,
    "doseDiff" : 1.75
}
{
    "_id" : {
        "matricolaImp" : 26,
        "nomeImp" : "Federica",
        "cognomeImp" : "Porzio",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 1.93,
    "doseDiff" : 0.9299999999999999
}
{
    "_id" : {
        "matricolaImp" : 27,
        "nomeImp" : "Atenulf",
        "cognomeImp" : "Mortati",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 3.3,
    "doseDiff" : 2.3
}
```

```

{
    "_id" : {
        "matricolaImp" : 28,
        "nomeImp" : "Morena",
        "cognomeImp" : "Salieri",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 2.7,
    "doseDiff" : 1.700000000000002
}
{
    "_id" : {
        "matricolaImp" : 29,
        "nomeImp" : "Giustino",
        "cognomeImp" : "Solimena",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 3.52,
    "doseDiff" : 2.52
}
{
    "_id" : {
        "matricolaImp" : 30,
        "nomeImp" : "Bernardo",
        "cognomeImp" : "Bruno",
        "doseSogliaImp" : 1
    },
    "totDoseRegistrata" : 3.4,
    "doseDiff" : 2.4
}

```

La rispettiva query in MySQL è stata illustrata nel capitolo 14 “Query”, paragrafo 14.8 “Dosimetria 1”

### 17.1.4 Dosimtria 2

“Si vogliono sapere tutti gli impiegati che il mese di Luglio 2021 hanno registrato una dose superiore a quella dell’anno precedente nello stesso mese, rappresentando entrambi i valori?”

```
> db.CategoriaEsposizione.aggregate([
... {$unwind:{path: "$inquadramentoInCategoria"}},
... {$unwind:{path: "$inquadramentoInCategoria.qualificati"}},
... {$unwind:{path: "$inquadramentoInCategoria.qualificati.dosimetri"}},
... {$match:{"inquadramentoInCategoria.qualificati.dosimetri.inizioPeriodo":
...     {$in:[ISODate('2020-07-01T00:00:00Z'),ISODate('2021-07-01T00:00:00Z')]}}},
... {$group:{_id: {"matricolaImp": "$inquadramentoInCategoria.qualificati.dosimetri.matricola"},
... "dosInfo":{$push:{"inizioPeriodoDos": "$inquadramentoInCategoria.qualificati.dosimetri.inizioPeriodo",
... "doseReg": "$inquadramentoInCategoria.qualificati.dosimetri.doseRegistrata"}}}},
... {$addFields:{"doseInfo1":{$arrayElemAt: [ "$dosInfo", 0 ]}}},
... {$addFields:{"doseReg1": "$doseInfo1.doseReg"}},
... {$addFields:{"doseInfo2":{$arrayElemAt: [ "$dosInfo", 1 ]}}},
... {$addFields:{"doseReg2": "$doseInfo2.doseReg"}},
... {$addFields:{"doseDiff":{$subtract:["$doseReg2", "$doseReg1"]}}},
... {$match:{"doseDiff":{$gt:0}}},
... {$project:{"_id.matricolaImp":1,"dosInfo":1}},
... {$sort:{"_id.matricolaImp":1}},
... ]).pretty()
{
    "_id" : {
        "matricolaImp" : 2
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.14
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.28
        }
    ]
}
{
    "_id" : {
        "matricolaImp" : 3
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),

```

```

        "doseReg" : 0.6
    },
    {
        "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
        "doseReg" : 0.63
    }
]
}
{
    "_id" : {
        "matricolaImp" : 4
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.4
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.86
        }
    ]
}
{
    "_id" : {
        "matricolaImp" : 6
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.03
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.46
        }
    ]
}
{
    "_id" : {
        "matricolaImp" : 7
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),

```

## RADB

```
        "doseReg" : 0.01
    },
    {
        "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
        "doseReg" : 0.39
    }
]
}
{
    "_id" : {
        "matricolaImp" : 9
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.48
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.58
        }
    ]
}
{
    "_id" : {
        "matricolaImp" : 10
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.46
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.86
        }
    ]
}
{
    "_id" : {
        "matricolaImp" : 11
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
```

```

        "doseReg" : 0.28
    },
    {
        "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
        "doseReg" : 0.42
    }
}
],
{
    "_id" : {
        "matricolaImp" : 14
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.57
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.65
        }
    ]
},
{
    "_id" : {
        "matricolaImp" : 15
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.66
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.79
        }
    ]
},
{
    "_id" : {
        "matricolaImp" : 16
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),

```

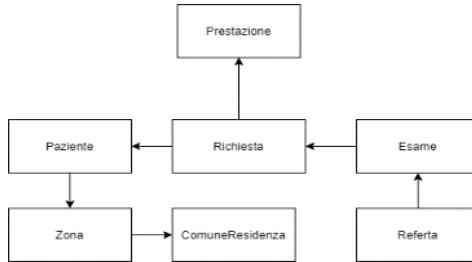
## RADB

```
        "doseReg" : 0.25
    },
    {
        "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
        "doseReg" : 0.41
    }
]
}
{
    "_id" : {
        "matricolaImp" : 21
    },
    "dosInfo" : [
        {
            "inizioPeriodoDos" : ISODate("2020-07-01T00:00:00Z"),
            "doseReg" : 0.56
        },
        {
            "inizioPeriodoDos" : ISODate("2021-07-01T00:00:00Z"),
            "doseReg" : 0.7
        }
    ]
}
```

La rispettiva query in MySQL è stata illustrata nel capitolo 14 “Query”, paragrafo 14.9 “Dosimetria 2”.

## 17.2 Ramo Info Pazienti

Il ramo scelto è strutturato nello schema fisico nel seguente modo:



per strutturare il documento denormalizzato in MongoDB è stato scelto di unire le informazioni di zona e comune di Residenza con l'anagrafica del Paziente, le informazioni delle prestazioni con le richieste e tutte le informazioni sulla refertazione con gli esami. Successivamente si è strutturato il documento in modo che a ogni Paziente afferissero le relative richieste come sotto documento, e che a sua volta ogni Richiesta inglobasse l'esame(se eseguito) come sotto documento.

```
> db.PazientiRichieste.find().limit(1).pretty()
{
  "_id" : ObjectId("6125053566f812118d86a012"),
  "idPaziente" : "4",
  "nome" : "Luciana",
  "cognome" : "Gianetti",
  "genere" : "F",
  "dataNascita" : "1977-05-07",
  "cittaNascita" : "Roma",
  "codiceFiscale" : "GNILCN77E47H501P",
  "indirizzo" : "Strada Mario 9",
  "cap" : "70121",
  "comune" : "Bari",
  "regione" : "Puglia",
  "telefono" : "1310717",
  "note" : "Soggetto a mobilità ridotta",
  "Richieste" : [
    ...
  ],
  {
    "_id" : ObjectId("6125058166f812118d86a0d5"),
    "idRichiesta" : "9",
    "idPaziente" : "4",
    "quesitoDiagnostico" : "Connettività Indifferenziate",
    "dataOraRichiesta" : ISODate("2021-06-30T00:30:00Z"),
    "esame" : [
      ...
    ]
  }
}
```

## RADB

```
    "codicePrestazione" : "RX08SN",
    "tipoPrestazione" : "RX",
    "nomePrestazione" : "RX Gamba Sinistra",
    "costo" : NumberDecimal("21.17"),
    "LDR" : NumberDecimal("4.00"),
    "priorita" : "3",
    "nomeDipartimento" : "W7",
    "dataOraAppuntamento" : ISODate("2021-07-01T00:30:00Z"),
    "numeroStanza" : "1",
    "statoRichiesta" : "Accettato",
    "matricolaAmministrativo" : "25",
    "Esame" : [
        {
            "_id" : ObjectId("6125050e66f812118d867c95"),
            "idEsame" : "4",
            "idRichiesta" : "9",
            "dataOraEsame" : ISODate("2021-07-01T02:41:00Z"),
            "matricolaTSRM" : "21",
            "statoEsame" : "Completato",
            "immaginiAccettate" : 3,
            "immaginiRifiutate" : 2,
            "dosemGycm" : NumberDecimal("5.10"),
            "commento" : "Paziente non deambulante"
        }
    ],
    ...
}, ...
```

A seguire alcuni esempi di possibili query da eseguire sul ramo.

### 17.2.1 Ricerca dati pazienti per età

“Si vogliono sapere id, nome, cognome, data di nascita e codice fiscale dei pazienti d’età inferiore o uguale a due anni”

```
> db.PazientiRichieste.find({"dataNascita": {$gt:"2018-08-21"}}, {"_id":0,"idPaziente":1, "nome":1, "cognome":1, "dataNascita":1, "codiceFiscale":1}).pretty()

{
    "idPaziente" : 94,
    "nome" : "Emma",
    "cognome" : "Cilea",
    "dataNascita" : "2018-11-16",
    "codiceFiscale" : "CLIMME18S56FB839B"

}
{
    "idPaziente" : 110,
    "nome" : "Ramona",
    "cognome" : "Carocci",
    "dataNascita" : "2020-03-20",
    "codiceFiscale" : "CRCRMN20C60A944W"

}
{
    "idPaziente" : 87,
    "nome" : "Berenice",
    "cognome" : "Ruberto",
    "dataNascita" : "2020-08-04",
    "codiceFiscale" : "RBRBNC20M44L219K"

}
{
    "idPaziente" : 109,
    "nome" : "Ivo",
    "cognome" : "Brunello",
    "dataNascita" : "2019-11-19",
    "codiceFiscale" : "BRNVIO19S19A662E"

}
{
    "idPaziente" : 53,
    "nome" : "Livio",
    "cognome" : "Anichini",
    "dataNascita" : "2019-09-03",
    "codiceFiscale" : "NCHLVII9P03A662J"

}
{
    "idPaziente" : 54,
    "nome" : "Vincenzo",
    "cognome" : "Costalonga",
    "dataNascita" : "2020-04-08",
```

## RADB

```
"codiceFiscale" : "CSTVCN20D08H501X"  
}
```

La rispettiva query in MySQL è:

```
mysql> SELECT Paziente.idPaziente, Paziente.nome, Paziente.cognome, Paziente.dataNascita,  
Paziente.codiceFiscale  
    -> FROM Paziente  
    -> WHERE Paziente.dataNascita > '2018-08-21';  
+-----+-----+-----+-----+  
| idPaziente | nome      | cognome   | dataNascita | codiceFiscale |  
+-----+-----+-----+-----+  
|      53 | Livio     | Anichini   | 2019-09-03 | NCHLV119P03A662J |  
|      54 | Vincentio | Costalonga | 2020-04-08 | CSTVCN20D08H501X |  
|      87 | Berenice   | Ruberto    | 2020-08-04 | RBRBNC20M44L219K |  
|      94 | Emma       | Cilea      | 2018-11-16 | CLIMME18S56F839B |  
|     109 | Ivo        | Brunello   | 2019-11-19 | BRNVI019S19A662E |  
|     110 | Ramona    | Carocci   | 2020-03-20 | CRCRMN20C60A944W |  
+-----+-----+-----+-----+  
6 rows in set (0.01 sec)
```

### 17.2.2 Selezione Pazienti

“Si vogliono sapere tutti i pazienti che hanno avuto un appuntamento per una TC con contrasto in stanza 8 il giorno “11-08-2021” dalle ore 8:00 alle ore 12:00 con il relativo quesito diagnostico e il referto .

```
> db.PazientiRichieste.aggregate([  
... {$unwind:{path:"$Richieste"}},  
... {$match:{$and:[{"Richieste.nomePrestazione":/.*TC.*mdc*/}, {"Richieste.numeroStanza":8}, {"Richieste.statoRichiesta": "Accettato"},  
... {"Richieste.dataOraAppuntamento": {$gte:ISODate("2021-08-11T08:00:00Z"), $lte:ISODate("2021-08-11T12:00:00Z")}}]},  
... {$project:{idPaziente:1, nome:1, cognome:1, Richieste.quesitoDiagnostico:1, "Richieste.Esame.referto":1}}]).pretty()  
{  
    "_id" : ObjectId("612253d84ceb1f2dd6a0399d"),  
    "idPaziente" : 115,  
    "nome" : "Simonetta",  
    "cognome" : "Verdone",  
    "Richieste" : [  
        {"quesitoDiagnostico" : "Epilessia",  
        "Esame" : [  
            {}  
        ],  
        "referto" : "Presenza di segni riferibili alla patologia. Si consiglia  
visione da parte di uno specialista"  
    }  
}  
}  
{  
    "_id" : ObjectId("612253d84ceb1f2dd6a039e3"),  
    "idPaziente" : 48,
```

```

        "nome" : "Agnolo",
        "cognome" : "Satriani",
        "Richieste" : [
            {
                "quesitoDiagnostico" : "Morbo di Basedow",
                "Esame" : [
                    {
                        "referto" : "Non Apprezzabili segni riferibili alla patologia"
                    }
                ]
            }
        ]
    }

{
    "_id" : ObjectId("612253d84ceb1f2dd6a039f2"),
    "idPaziente" : 119,
    "nome" : "Flora",
    "cognome" : "Rapisardi",
    "Richieste" : [
        {
            "quesitoDiagnostico" : "Aneurisma Cerebrale",
            "Esame" : [
                {
                    "referto" : "Presenza di aneurisma di severe dimensioni. Si consigliano ulteriori approfondimenti diagnostici"
                }
            ]
        }
    ]
}

```

La rispettiva query in MySQL è:

```

mysql> SELECT Paziente.idPaziente, Paziente.nome, Paziente.cognome, Richiesta.quesitoDiagnostico, Referta.referto
-> FROM Paziente, Richiesta, Esame, Referta, Prestazione
-> WHERE Richiesta.dataAppuntamento = '2021-08-11'
-> AND Richiesta.oraAppuntamento BETWEEN '8:00:00' AND '12:00:00'
-> AND Prestazione.nomePrestazione like 'TC%mdc'
-> AND Richiesta.idPaziente = Paziente.idPaziente
-> AND Richiesta.codicePrestazione = Prestazione.codicePrestazione
-> AND Richiesta.idRichiesta = Esame.idRichiesta
-> AND Richiesta.numeroStanza = 8
-> AND Esame.idEsame = Referta.idEsame;

```

idPaziente	nome	cognome	quesitoDiagnostico	referto
48	Agnolo	Satriani	Morbo di Basedow	Non Apprezzabili segni riferibili alla patologia
119	Flora	Rapisardi	Aneurisma Cerebrale	Presenza di aneurisma di severe dimensioni. Si consigliano ulteriori approfondimenti diagnostici
118	Simonetts	Verdone	Epilessia	Presenza di segni riferibili alla patologia. Si consiglia visione da parte di uno specialista

3 rows in set (0.02 sec)

### 17.2.3 Indagine Statistica 1

“Si vuole sapere quanti pazienti, a oggi, della regione Puglia hanno effettuato una richiesta con un’esonzione presso la struttura”

```
> db.PazientiRichieste.aggregate([
... {$unwind:{path:"$Richieste"}},
... {$match:{$and:[{"regione": "Puglia"}, {"Richieste.codiceEsenzione":{$exists: true}}]}},
... {$group:{_id: "$regione", "totRich": { $sum : 1}}},
... ]).pretty()

{ "_id" : "Puglia", "totRich" : 940 }
```

La rispettiva query in MySQL è stata illustrata nel capitolo 14 “Query”, paragrafo 14.16 “Indagine Statistica 1”.

### 17.2.4 Indagine Statistica 2

“Si vogliono sapere le percentuali di richiesta di pazienti provenienti dalle diverse regioni, divisi per regione.”

```
> db.PazientiRichieste.aggregate([
... {$facet:{"totaleRichieste": [{"group:{_id: null,"totRich": {$sum : {$size:"$Richieste"}}}},,
... {"addFields:{"fattoreMul":{$divide:[100,$totRich]}}}],,
... "richiestePerRegione": [
... {"$unwind:{path: "$Richieste"},,
... {"group:{_id: {"nomeRegione": "$regione"}, "quantitaRichieste": {$sum : 1}}}}],,
... {"$unwind:{path:'$richiestePerRegione'},,
... {"$unwind:{path:'$totaleRichieste'}},,
... {"$addFields:{"percentuale":{$multiply:[{$richiestePerRegione.quantitaRichieste}, "$totaleRichieste.fattoreMul"]}}},,
... {"$project:{"richiestePerRegione._id.nomeRegione":1,
... {"percentuale":1}}}
... ]).pretty()
{

    "richiestePerRegione" : {
        "_id" : {
            "nomeRegione" : "Emilia Romagna"
        }
    },
    "percentuale" : 15.848933034983354
}
{
    "richiestePerRegione" : {
        "_id" : {
            "nomeRegione" : "Lombardia"
        }
    }
}
```

```

        },
        "percentuale" : 10.156633739016536
    },
    {
        "richiestePerRegione" : {
            "_id" : {
                "nomeRegione" : "Toscana"
            },
            "percentuale" : 12.115919882115373
        }
    },
    {
        "richiestePerRegione" : {
            "_id" : {
                "nomeRegione" : "Sicilia"
            },
            "percentuale" : 11.853954046826392
        }
    },
    {
        "richiestePerRegione" : {
            "_id" : {
                "nomeRegione" : "Puglia"
            },
            "percentuale" : 16.012661682038967
        }
    },
    {
        "richiestePerRegione" : {
            "_id" : {
                "nomeRegione" : "Piemonte"
            },
            "percentuale" : 9.47988866452
        }
    },
    {
        "richiestePerRegione" : {
            "_id" : {
                "nomeRegione" : "Campania"
            },
            "percentuale" : 12.132292746820934
        }
    },
    {
        "richiestePerRegione" : {

```

## RADB

```
_id" : {  
    "nomeRegione" : "Lazio"  
}  
},  
"percentuale" : 12.399716203678436  
}
```

La rispettiva query in MySQL è stata illustrata nel capitolo 14 “Query”, paragrafo 14.17 “Indagine Statistica 2”.

## 17.3 Confronto tempi di Esecuzione

A seguire verranno confrontati tempi di esecuzione di due tra le query d'esempio illustrate per MongoDB e le corrispettive query eseguite in MySQL. Prima del confronto sono stati inseriti in entrambi i database oltre 200000 record sul ramo Info Pazienti. Per avere un quadro completo verrà scelta una query sul ramo Dosimetria che anche dopo l'inserimento di massa mantiene lo stesso numero di record, e una sul ramo Info Pazienti coinvolto nell'inserimento massivo di record, in modo da confrontare le performance nei due casi.

Le query selezionate sono le seguenti:

- Dosimetria 1
- Indagine Statistica 1

### 17.3.1 Dosimetria 1

La query in MySQL, già illustrata nel capitolo 14 “Query”, viene riportata di seguito:

```
mysql> SELECT Dosimetro.matricola, Impiegato.nome, Impiegato.cognome, sum(Dosimetro.doseRegistrata) as doseRegistrata,
-> CategoriaEsposizione.doseSoglia,
-> sum(Dosimetro.doseRegistrata)-CategoriaEsposizione.doseSoglia as differenzaDose
-> FROM Impiegato, Dosimetro, Qualifica, CategoriaEsposizione
-> WHERE year(Dosimetro.inizioPeriodo) = 2021
-> AND Impiegato.matricola = Dosimetro.matricola
-> AND Impiegato.codiceQualifica = Qualifica.codiceQualifica
-> AND Qualifica.codiceCategoria = CategoriaEsposizione.codiceCategoria
-> GROUP BY Dosimetro.matricola having sum(Dosimetro.doseRegistrata) > CategoriaEsposizione.doseSoglia;
+-----+-----+-----+-----+
| matricola | nome      | cognome   | doseRegistrata | doseSoglia | differenzaDose |
+-----+-----+-----+-----+
|    25 | Emilio    | Sokolov   |      2.75 |     1.00 |      1.75 |
|    26 | Federica  | Porzio    |      1.93 |     1.00 |      0.93 |
|    27 | Atenulf   | Mortatti  |      3.30 |     1.00 |      2.30 |
|    28 | Morena    | Salieri   |      2.70 |     1.00 |      1.70 |
|    29 | Giustino  | Solimena  |      3.52 |     1.00 |      2.52 |
|    30 | Bernardo  | Bruno     |      3.40 |     1.00 |      2.40 |
|     1 | Pier       | Morlacchi |      3.71 |     1.00 |      2.71 |
|     2 | Cassandra  | Cabibbo   |      2.35 |     1.00 |      1.35 |
+-----+-----+-----+-----+
8 rows in set (0.01 sec)
```

La query in MySQL è molto performante, su una piccola quantità di dati impiega 0.01 s. Osserviamo ora i tempi di esecuzione della query corrispondente in MongoDB:

## RADB

```
> db.CategoriaEsposizione.explain('executionStats').aggregate([
... {$unwind:{path: "$inquadramentoInCategoria"}},
... {$unwind:{path: "$inquadramentoInCategoria.qualificati"}},
... {$unwind:{path: "$inquadramentoInCategoria.qualificati.dosimetri"}},
... {$match:{"inquadramentoInCategoria.qualificati.dosimetri.inizioPeriodo":
...     {$gte:ISODate('2021-01-01T00:00:00Z'),$lt:ISODate('2022-01-01T00:00:00Z')}},
... {$group:{_id: {"matricolaImp": "$inquadramentoInCategoria.qualificati.dosimetri.matricola",
...     "nomeImp": "$inquadramentoInCategoria.qualificati.nome",
...     "cognomeImp": "$inquadramentoInCategoria.qualificati.cognome",
...     "doseSogliaImp": "$doseSoglia"}, "totDoseRegistrata":{$sum:"$inquadramentoInCategoria.qualificati.dosimetri.doseRegistrata"}}},
... {$project:{"totDoseRegistrata":1,"doseDiff":{$subtract:["$totDoseRegistrata", "$_id.doseSogliaImp"]}}},
... {$match: {"doseDiff": {$gt: 0}}},
... {$sort:{"_id.matricolaImp":1}},
... ])
{
...
    "executionStats" : {
        "executionSuccess" : true,
        "nReturned" : 3,
        "executionTimeMillis" : 8,
        "totalKeysExamined" : 0,
        "totalDocsExamined" : 3,
        ...
    }
}
```

Osserviamo che il tempo di esecuzione è di 8 ms, ovvero 0.008 s, lievemente inferiore rispetto al tempo impiegato da MySQL.

### 17.3.2 Indagine Statistica 1

La query in MySQL, già illustrata nel capitolo 14 “Query”, viene riportata di seguito:

```
mysql> SELECT count(Associata.idRichiesta) totale
   -> FROM Associata, Richiesta, Paziente, Zona, comuneResidenza
   -> WHERE comuneResidenza.Regione = 'Puglia'
   -> AND Zona.comune = ComuneResidenza.comune
   -> AND Paziente.cap = Zona.cap
   -> AND Richiesta.idPaziente = Paziente.idPaziente
   -> AND Richiesta.idRichiesta = Associata.idRichiesta;
+-----+
| totale |
+-----+
| 10715 |
+-----+
1 row in set (0.52 sec)
```

La query in MySQL su un' ingente quantità di dati impiega 0.52 s. Osserviamo ora i tempi di esecuzione della query corrispondente in MongoDB:

```
db.PazientiRichieste.explain('executionStats').aggregate([
  {$unwind:{path:"$Richieste"},},
  {$match:{$and:[{"regione": "Puglia"}, {"Richieste.codiceEsenzione":{$exists: true}}]}},
  {$group:{_id: "$regione", "totRich": { $sum : 1}}},
  {}]

> db.PazientiRichieste.explain('executionStats').aggregate([
  ... {$unwind:{path:"$Richieste"},},
  ... {$match:{$and:[{"regione": "Puglia"}, {"Richieste.codiceEsenzione":{$exists: true}}]}},
  ... {$group:{_id: "$regione", "totRich": { $sum : 1}}},
  ... {}]
{
  ...
    "executionStats" : {
      "executionSuccess" : true,
      "nReturned" : 30488,
      "executionTimeMillis" : 719,
      "totalKeysExamined" : 0,
      "totalDocsExamined" : 244112,
      ...
    }
}
```

La query impiega 719 ms, 0,72 s, rispetto la corrispondente query in MySQL è quindi meno performante.

I precedenti risultati sono stati confrontati evitando l'uso della cache per MySQL e del file mappato in memoria che mantiene in RAM i risultati delle query per MongoDB. È stata eseguita un'ulteriore analisi eseguendo le query una seconda volta per vedere come variassero i tempi di esecuzione:

```
mysql> SELECT count(Associata.idRichiesta) totale
   > FROM Associata, Richiesta, Paziente, Zona, comuneResidenza
   > WHERE comuneResidenza.Regione = 'Puglia'
   > AND Zona.comune = ComuneResidenza.comune
   > AND Paziente.cap = Zona.cap
   > AND Richiesta.idPaziente = Paziente.idPaziente
   > AND Richiesta.idRichiesta = Associata.idRichiesta;
+-----+
| totale |
+-----+
| 10715 |
+-----+
1 row in set (0.13 sec)
```

## RADB

```
> db.PazientiRichieste.explain('executionStats').aggregate([
... {$unwind:{path:"$Richieste"}},
... {$match:{$and:[{"regione": "Puglia"}, {"Richieste.codiceEsenzione":{$exists: true}}]}},
... {$group:{_id: "$regione", "totRich": {$sum : 1}}},
... ])
{
  ...
  "executionStats" : {
    "executionSuccess" : true,
    "nReturned" : 30488,
    "executionTimeMillis" : 304,
    "totalKeysExamined" : 0,
    "totalDocsExamined" : 244112,
    ...
  }
}
```

Anche in questo caso si è dimostrato più performante MySQL con i suoi 0.15 s verso i 0.30 s in MongoDB.

## 17.4 Conclusioni

Sulla base dei precedenti risultati si è scelto di confrontare le misurazioni per tutte le query proposte sul Ramo Info Paziente sia per MySQL sia per MongoDB ottenendo i seguenti risultati:

Query	MySQL	MongoDB
Ricerca dati pazienti per età	0.20 s	0.99 s
Selezione Pazienti	0.15 s	1.31 s
Indagine Statistica 1	0.52 s	0.72 s
Indagine Statistica 2	0.81 s	2.23 s

MongoDB si è dimostrato lievemente più performante su un esiguo numero di record rispetto MySQL, mentre al crescere del numero di record sono stati registrati tempi di esecuzione migliori da parte di MySQL.

Il database gestito mediante MySQL è una struttura normalizzata, mentre per MongoDB è stato scelto di usare dati denormalizzati.

Si è giunti quindi alla conclusione che, sulla base del tipo di strutture scelte, per il tipo di operazioni richieste e sulla base dei dati registrati dopo l'inserimento massivo di record, MySQL si è dimostrato più performante.