

CORSO DI LAUREA IN TECNICHE DI RADIOLOGIA MEDICA
PER IMMAGINI E RADIOTERAPIA

**CORSO DI: SISTEMI DI ELABORAZIONE DELLE
INFORMAZIONI I**

Anno Accademico 2017/2018

Dott. Silvio Pardi

Lezione N°5

Che cos'è un Database

Definiamo un **Database (DB)** un **insieme di dati tra loro correlati** che possono essere utilizzati e gestiti da più applicativi.

Un **Data Base Management System (DBMS)** è un software che realizza database mettendo a disposizione comandi e strumenti per l'aggiornamento, la manutenzione e la consultazione dei dati di un DB.

relational database management system (RDBMS)

Esempio di Database

Un Data Base è una collezione strutturata di record o dati

Un insieme di dati, utilizzati per rappresentare le informazioni per qualche scopo specifico:

- L'insieme dei dati anagrafici dei pazienti di un ospedale
- L'insieme delle prenotazioni degli esami da svolgere in un centro radiologico
- L'insieme delle immagini radiologiche prodotte da un centro

DBMS: esempio

Si consideri una Ospedale che registri le informazioni anagrafiche relative ai pazienti, le informazioni sulle patologie e le informazioni su ingressi e uscite.

- Il DBMS dovrà disporre comandi per: Gestire i pazienti (aggiungere/togliere)
 - Creare nuovi Pazienti
 - Estrarre le informazioni su uno o più pazienti
 - Aggiornare le informazioni
 - Generare report
- Man mano che si manifestano nuove esigenze, può essere necessario modificare le applicazioni esistenti o disporre di nuovi programmi
- Ad esempio si supponga che venga aperto un nuovo reparto

Operazioni effettuabili su un DB

Le operazioni fondamentali che possiamo svolgere su un DataBase sono le seguenti:

- **immissione** dati
- **cancellazione** di dati
- **Modifica**
- **Ricerca** di dati attraverso criteri definiti dall'utente
- **Ordinamento e classificazione** dei dati singolarmente o secondo vari criteri
- **Stampa** di rapporti o relazioni.

DataBase Relazionali

Si parla di Database Relazionali delle particolari strutture di database costituiti da Insieme di tabelle messe in relazione tra loro tramite i campi.

I campi comuni tra tabelle diverse stabiliscono le relazioni della base di dati

I software che gestiscono database relazionali vengono indicati come **RDBMS (Relational Data Base Management System)**

RDBMS

Esempi di software disponibili per la realizzazione di RDBMS sono:

- Software Commerciali commerciali
 - Oracle/MySQL
 - IBM DB2
 - Microsoft Access
- Software open source
 - OpenOffice Base
 - MariaDB
 - PostgreSQL

Gli elementi principali di un database

- Tabelle
- Maschere
- Query
- Report

1. Le tabelle

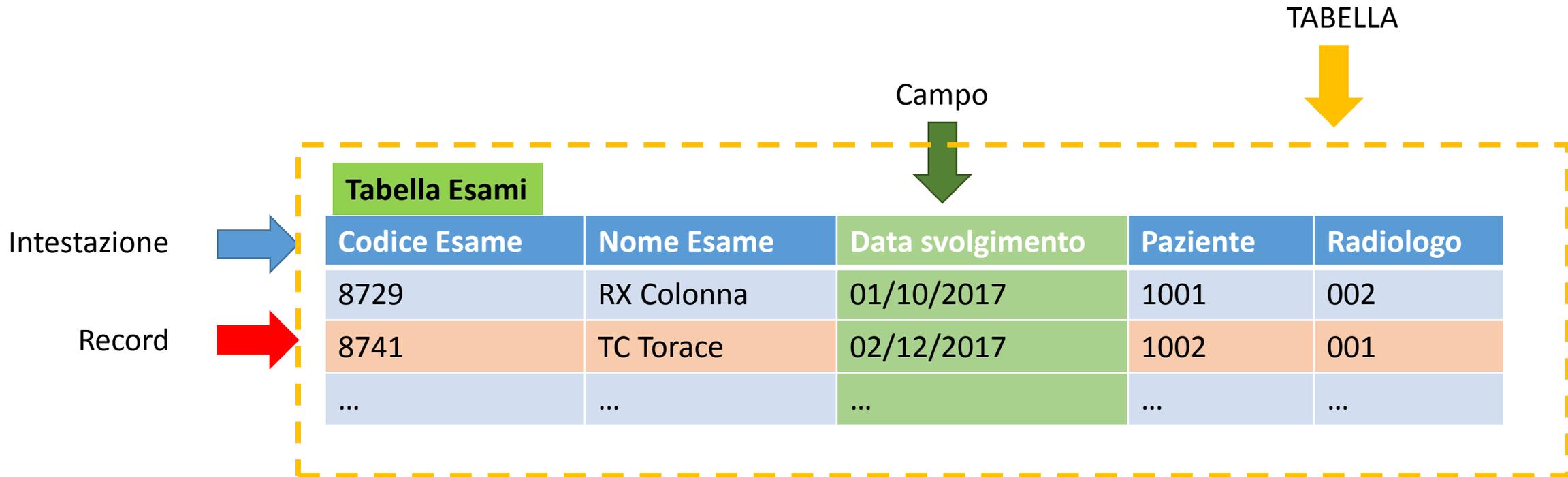
All'interno di un database le informazioni vengono raggruppate sotto forma di tabelle:

Una tabella è una griglia suddivisa in righe e colonne:

- Le colonne rappresentano i **campi ovvero le categorie**
- Le righe sono chiamate **record o entry** contengono i **dati veri e propri**

In una tabella la prima riga viene detta **intestazione** e contiene il nome dei campi.

Esempio di DataBase Relazionale



Chiave primaria di una Tabella

- **Definizione:**
 - La chiave primaria è un campo che identifica in modo univoco i record (righe) della tabella.
- **Esempi:**
 - Un codice fiscale consente di identificare in modo univoco la persona: non esistono due codici fiscali identici
 - Il Codice Esame identifica in modo univoco un tipo di esame
 - Il numero di Badge identifica in modo univoco un lavoratore
- La chiave primaria è l'elemento fondamentale per stabilire relazioni tra tabelle. Essa rappresenta infatti il **campo di collegamento che identifica in maniera univoca la riga di una tabella in un'altra tabella.**
- Per ogni tabella deve essere indicato esplicitamente quale campo funge da chiave primaria

Esempio di Tabelle

Tabella Pazienti

ID	Cognome	Nome	Data di Nacita	Genere
1001	Rossi	Antonio	01/12/1938	M
1002	Bianchi	Maria	12/09/1940	F
...

Tabella Esami

Codice Esame	Nome Esame	Sala
8729	RX Colonna	Room A
8741	TC Torace	Room B
...

Tabella Radiologi

ID	Cognome	Nome	Data di Nacita	Città
001	Conte	Lucia	23/04/1982	Napoli
002	Barone	Mario	02/03/1985	Napoli
...

Quali sono le chiavi primarie delle tabelle precedenti?

Le relazioni: esempio

Pensiamo ad un Centro Radiologico che colleziona al proprio interno tutte le informazioni relative ai pazienti, ad esempio: Nome, Cognome, indirizzo, data di nascita, recapiti e le informazioni relative ai dipendenti, Medici, Radiologi.

Gli stessi dati vengano reperiti dal Centro di Radiologia della struttura, dal Laboratorio di Analisi e da altri dipartimenti con un evidente ridondanza e spreco di spazi.

Inoltre un cambiamento (per esempio il recapito di un paziente o la disponibilità di un operatore) dovrebbero essere cambiati in tutte le tabelle che li contengono

Per risolvere questo inconveniente, ci viene in aiuto la possibilità che hanno i programmi di gestione dei database, di organizzare il database su più tabelle mettendole in **RELAZIONE (collegamento logico)** tra di loro.

Le relazioni: esempio

- Si potrebbero quindi suddividere le nostre informazioni in **Quattro tabelle**:
 - La prima riporta le informazioni dei pazienti
 - La seconda le informazioni sui Radiologi
 - La terza informazioni sugli esami e sulle sale nelle quali sono presenti le strumentazioni per eseguirli
 - La quarta tabella una WorkList che comprende Paziente, Esame e Sala ognuno con il suo codice identificativo.
- In questo modo i dati non sono più ripetuti e se dovesse essere necessario modificare qualche informazione su un paziente o su un esame basterà fare una modifica nella tabella relativa e non per tutte le volte in cui compare lo specifico paziente o lo specifico operatore.

Esempio di Tabele

Pazienti

ID	Cognome	Nome	Data di Nacita	Genere
4001	Rossi	Antonio	01/12/1938	M
4002	Bianchi	Maria	12/09/1940	F
...

Radiologi

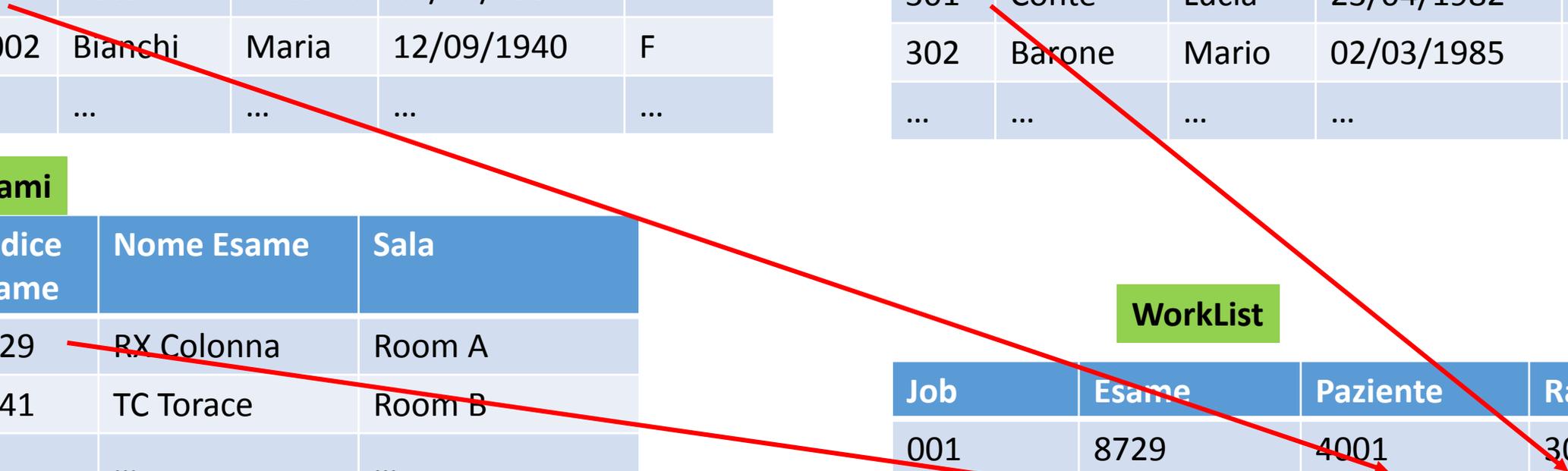
ID	Cognome	Nome	Data di Nacita	Città
301	Conte	Lucia	23/04/1982	Napoli
302	Barone	Mario	02/03/1985	Napoli
...

Esami

Codice Esame	Nome Esame	Sala
8729	RX Colonna	Room A
8741	TC Torace	Room B
...

WorkList

Job	Esame	Paziente	Radiologo
001	8729	4001	301
002	8741	4002	302
...

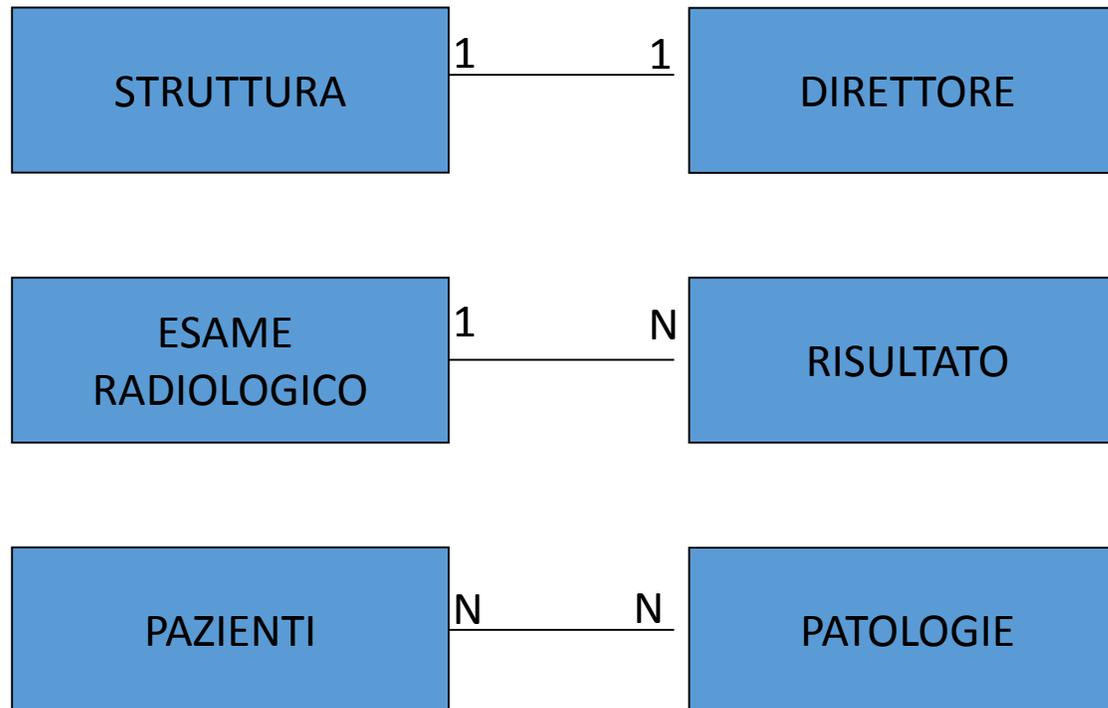


Le relazioni

Si possono stabilire tre tipi di relazione:

- **uno a uno**: si tratta di relazioni tra elementi che hanno una corrispondenza univoca: ad un elemento di una tabella ne corrisponde uno soltanto in un'altra e viceversa;
- **uno a molti**: sono relazioni che si stabiliscono tra un record di una tabella e più records di un'altra tabella, ma non il contrario;
- **molti a molti**: un record può essere relazionato a più di un record di un'altra tabella e viceversa; questo tipo di relazione è normalmente definita tramite una terza tabella che costituisce un "ponte" tra le due da relazionare.

Le relazioni



2. Le maschere

- In una tabella i dati possono essere inseriti:
 - Utilizzando i comandi dello specifico Database
 - Linguaggio SQL
 - Scrivendo nelle Caselle di un documento EXCEL

oppure:

- utilizzando delle maschere di inserimento

Le **maschere di inserimento** dati sono dei sistemi per facilitare la creazione di tabella e per l'inserimento di dati in maniera guidata

Le maschere sono utili anche per la visualizzazione ordinata del contenuto delle tabelle.

Nelle maschere si possono introdurre anche elementi grafici e i campi possono essere formattati secondo stili diversi.

Le maschere: esempio

Image from
<https://customer.first-insight.com/MaximEyesSqlEHR/HtmlPages/Images/MaximEyesEditPatientRecord.jpg>

maximEyes SQL - Electronic Health Record

maximEyes Find Common Tasks Office Admin Windows

Shortcuts

- Home
- Find Patient
- New Patient
- New Encounter
- Today's Encounters
- Task Manager

Open Windows

- Home
- Find Patient
- Edit Patient Record

Patient Navigation

- Adult Contacts's
 - Patient Record
 - Encounters
 - Optical
 - Patient Medications

Edit Patient Record

Patient Info Phone Sheet Electronic Files

(P) Allergic to peanuts; Ibuprofen causes nausea; use another anti-inflammatory. [Add Manage](#)

Patient ID **101008** Chart # Patient Since 04/12/2010 Is Active?

Patient Details

Last Name * **Contacts**

First Name * **Adult**

Middle Initial Suffix

Salutation

Preferred Name [Load Photo](#)

Date of Birth * 06/25/1968 Age 42 Weight 134 lbs kg

SSN Guardian

Gender Male Female Emergency Contact

Ethnic Background Employer

Location * School

Appointments & Recalls

Last Appt	<input type="text"/>	For	<input type="text"/>
Next Appt	<input type="text"/>	For	<input type="text"/>
Next Recall	<input type="text"/>	For	<input type="text"/>
Last Exam Visit	<input type="text"/>	For	<input type="text"/>
Last Annual Exam	<input type="text"/>	For	<input type="text"/>

Other

HIPAA Privacy Statement Signed? Yes No Date

Notes

Main Address

Line 1

Line 2

City, State OR

Zip Code -

Phone & Email

Home * Day x

Fax Other

Cellular Pager

Email Permission to send emails?

Physicians

Provider

Primary Care Physician

Current Referring Physician

Insurances

1.	<input type="text" value="Northwest Optical Insuranc"/>	<input type="text" value="Primary Vision"/>
2.	<input type="text" value="Northwest Optical Insuranc"/>	<input type="text" value="Secondary Vision"/>
3.	<input type="text" value="-- Select --"/>	<input type="text" value="-- Select Type --"/>
4.	<input type="text" value="-- Select --"/>	<input type="text" value="-- Select Type --"/>

Main

Adult Contacts Patient Room | Owner | Task - Status Next priority EMR_1.0.1.0 User: admin

3. Le Query

Viene definita Query una richiesta per interrogare un database per conoscere i contenuti delle tabelle.

Le Query possono essere richieste semplici ma anche molto complesse contenenti filtri ed altresì operazioni sui dati ottenuti.

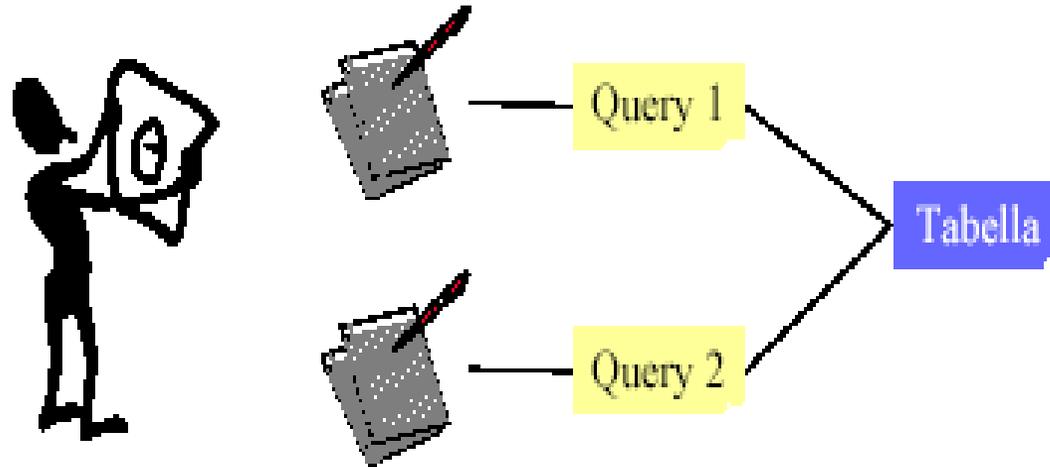
Il linguaggio SQL

- E' un linguaggio utilizzato per interagire con i database relazionali
- **SQL = Structured Query Language** (linguaggio di interrogazione strutturato)
- È universalmente accettato dai produttori di sistemi per database
 - Microsoft Access, Microsoft SQL server
 - Oracle
 - IBM DB2
 - MariaDB
 - MySQL
 - Postgress
- Può essere inglobato in linguaggi di programmazione tradizionali come C, C++, Java
- È in grado di effettuare le diverse operazioni per l'interazione con un database: **interrogazione**, **inserimento** di nuovi dati, **modifica** di dati esistenti, **eliminazione** di dati esistenti

4. I report

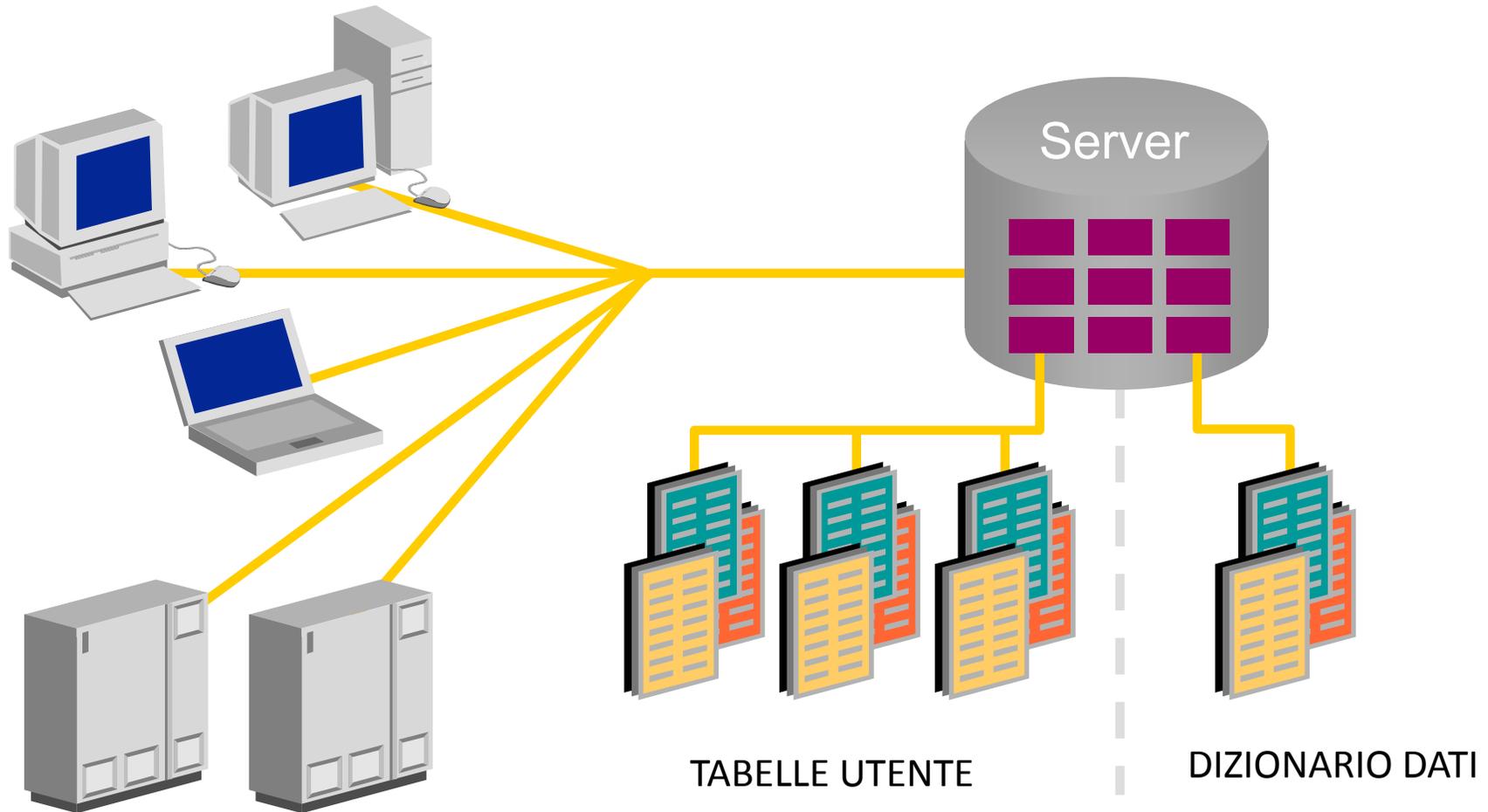
- Un report consente di visualizzare i dati memorizzati in tabelle o query del DB
- Tale visualizzazione non ha lo scopo di consentire inserimenti o modifiche, ma di PRESENTARE i dati in un formato efficace su carta o su video
- I **report** consentono di visualizzare in anteprima o stampare i dati secondo uno specifico formato.

I report



- I report visualizzano efficacemente i risultati di una o più query o i dati di una o più tabelle

RDBMS



Connessione al server MySQL

Accesso alla macchina: Passwd unicompania123

```
[spardi.osio] ➤ ssh 80.158.16.196 -l studenti
studenti@80.158.16.196's password:
X11 forwarding request failed on channel 0
[studenti@ecs-ad0e ~]$
```

Accesso al server mysql (unicampania)

```
[studenti@ecs-ad0e ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

Prime interazioni con il server

```
mysql> select version(),current_date;
```

```
+-----+-----+  
| version() | current_date |  
+-----+-----+  
| 5.1.73    | 2017-10-26   |  
+-----+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> select 4+3;
```

```
+-----+  
| 4+3 |  
+-----+  
|    7 |  
+-----+
```

```
1 row in set (0.00 sec)
```

Visiona l'elenco degli utenti

```
mysql> select user();  
+-----+  
| user() |  
+-----+  
| root@localhost |  
+-----+  
1 row in set (0.00 sec)
```

Selezione di un database

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| test |
+-----+
3 rows in set (0.00 sec)

mysql>
```

Creazione di un database

```
mysql> CREATE DATABASE RIS_Pardi;
```

```
mysql> show databases;
```

```
+-----+
```

```
| Database |
```

```
+-----+
```

```
| information_schema |
```

```
| RIS_Pardi |
```

```
| mysql |
```

```
| test |
```

```
+-----+
```

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

Creazione della Tabella Pazienti

```
mysql> use RIS_Pardi;  
Database changed  
  
mysql> create table Pazienti (  
-> ID INT NOT NULL AUTO_INCREMENT,  
-> Cognome VARCHAR(45) NOT NULL,  
-> Nome VARCHAR(45) NOT NULL,  
-> DataDiNascita DATE NOT NULL,  
-> Genere ENUM('M','F') NULL,  
-> PRIMARY KEY (ID));  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> show tables;  
+-----+  
| Tables_in_RIS_Pardi |  
+-----+  
| Pazienti             |  
+-----+  
1 row in set (0.00 sec)
```

Creazione di una tabella

```
mysql> describe Pazienti;
```

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	auto_increment
Cognome	varchar(45)	NO		NULL	
Nome	varchar(45)	NO		NULL	
DataDiNascita	date	NO		NULL	
Genere	enum('M','F')	YES		NULL	

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

Creazione di una tabella

```
mysql> use RIS_Pardi;  
Database changed  
  
mysql> create table Radiologi (  
-> ID INT NOT NULL AUTO_INCREMENT,  
-> Cognome VARCHAR(45) NULL,  
-> Nome VARCHAR(45) NULL,  
-> DataDiNascita DATE NULL,  
-> Citta VARCHAR(45) NULL,  
-> PRIMARY KEY (ID));
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> show tables;
```

```
+-----+  
| Tables_in_RIS_Pardi |  
+-----+  
| Pazienti           |  
| Radiologi          |  
+-----+
```

2 rows in set (0.00 sec)

Creazione di una tabella

```
mysql> describe Radiologi;
```

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	auto_increment
Cognome	varchar(45)	YES		NULL	
Nome	varchar(45)	YES		NULL	
DataDiNascita	date	YES		NULL	
Citta	varchar(45)	YES		NULL	

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

Creazione di una tabella

```
mysql> create table Esami (  
  -> CodiceEsame INT NOT NULL AUTO_INCREMENT,  
  -> NomeEsame VARCHAR(45) NULL,  
  -> Sala VARCHAR(45) NULL,  
  -> PRIMARY KEY (CodiceEsame));
```

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

```
mysql> show tables;
```

```
+-----+  
| Tables_in_RIS_Pardi |  
+-----+  
| Esami                |  
| Pazienti             |  
| Radiologi            |  
+-----+
```

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

Creazione di una tabella

```
mysql> describe Esami;
```

Field	Type	Null	Key	Default	Extra
CodiceEsame	int(11)	NO	PRI	0	
NomeEsame	varchar(45)	YES		NULL	
Sala	varchar(45)	YES		NULL	

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

Creazione di una tabella

```
mysql> create table WorkList (  
    -> Job INT NOT NULL AUTO_INCREMENT,  
    -> Esame INT NULL,  
    -> Paziente INT NULL,  
    -> Radiologo INT,  
    -> PRIMARY KEY (Job));
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> show tables;
```

```
+-----+  
| Tables_in_RIS_Pardi |  
+-----+  
| Esami                |  
| Pazienti             |  
| Radiologi            |  
| WorkList             |  
+-----+
```

4 rows in set (0.00 sec)

Creazione di una tabella

Cambio di nome

```
ALTER TABLE Radiologi RENAME Tecnici;
```

Aggiunta di un Campo

```
ALTER TABLE Radiologi ADD Presenza INT;
```

```
ALTER TABLE Radiologi DROP Presenza;
```

Inserire Entry nelle Tabelle

```
insert Pazienti (Cognome, Nome, DataDiNascita, Genere) values
('Napoli', 'Anna', '1958-07-03', 'F');

insert Pazienti (Cognome, Nome, DataDiNascita, Genere) values
('Bianchi', 'Maria', '1987-05-05', 'F');

insert Pazienti (Cognome, Nome, DataDiNascita, Genere) values
('Rossi', 'Antonio', '1948-02-13', 'M');

insert Pazienti (Cognome, Nome, DataDiNascita, Genere) values
('Roma', 'Luca', '1990-10-21', 'M');
```

Inserire Entry nelle Tabelle

```
mysql> select * from Pazienti;
```

ID	Cognome	Nome	DataDiNascita	Genere
1	Napoli	Anna	1958-07-03	F
2	Bianchi	Maria	1987-05-05	F
3	Rossi	Antonio	1948-02-13	M
4	Roma	Luca	1990-10-21	M

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

Inserire Entry nelle Tabelle

```
insert Radiologi (Cognome, Nome, DataDiNascita, Citta) values  
('Conte', 'Anna', '1985-07-03', 'Napoli');
```

```
insert Radiologi (Cognome, Nome, DataDiNascita, Citta) values  
('Barone', 'Maria', '1983-05-05', 'Caserta');
```

```
insert Radiologi (Cognome, Nome, DataDiNascita, Citta) values  
('Marchese', 'Antonio', '1988-02-13', 'Salerno');
```

```
insert Radiologi (Cognome, Nome, DataDiNascita, Citta) values  
('Principe', 'Luca', '1985-10-21', 'Pozzuoli');
```

Inserire Entry nelle Tabelle

```
mysql> select * from Radiologi;
```

```
+-----+-----+-----+-----+-----+
| ID | Cognome | Nome | DataDiNascita | Citta |
+-----+-----+-----+-----+-----+
| 1 | Conte | Anna | 1985-07-03 | Napoli |
| 2 | Barone | Maria | 1983-05-05 | Caserta |
| 3 | Marchese | Antonio | 1988-02-13 | Salerno |
| 4 | Principe | Luca | 1985-10-21 | Pozzuoli |
+-----+-----+-----+-----+-----+
```

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

Inserire Entry nelle Tabelle

```
insert Esami (CodiceEsame, NomeEsame, Sala) values ('8729', 'RX  
Colonna', 'Room A');  
insert Esami (CodiceEsame, NomeEsame, Sala) values ('8741', 'TC  
Torace', 'Room B');
```

Inserire Entry nelle Tabelle

```
mysql> select * from Esami;  
+-----+-----+-----+  
| CodiceEsame | NomeEsame | Sala |  
+-----+-----+-----+  
|          8729 | RX Colonna | Room A |  
|          8741 | TC Torace  | Room B |  
+-----+-----+-----+  
2 rows in set (0.00 sec)
```

Inserire Entry nelle Tabelle

```
insert WorkList (Esame, Paziente, Radiologo) values ('8729',  
  '2', '1');  
insert WorkList (Esame, Paziente, Radiologo) values ('8729',  
  '3', '1');  
insert WorkList (Esame, Paziente, Radiologo) values ('8729',  
  '4', '2');
```

Query

Le Query si esprimono in un linguaggio natural e si avvalgono dei campi from where e posson essere composte:

```
select Nome,Cognome from Pazienti;
```

```
select Nome,Cognome from Pazienti where Genere='M';
```

```
select Nome,Cognome from Pazienti where ID=(select Paziente  
from WorkList where Job='1');
```

Contare quanti esami di tipo 8729 sono in programma.

```
select count(*) from WorkList where Esame='8729';
```

```
select count(*) from WorkList where Esame=(select CodiceEsame  
from Esami where NomeEsame='RX Colonna');
```