

Corso di Base Di Dati

Università e 5° anno ITIS

In questo corso:

- Introduzione alle base di dati (terminologia, modello, utilizzo, ecc...)
- Come progettare, rappresentare e ristrutturare le Base Di Dati (ER e UML)
- Algebra relazionale
- SQL usando il DBMS PostgreSQL
- Funzioni e Trigger
- Normalizzazione



Argomenti di questa lezione

In questa lezione:

- Introduzione alle Base Di Dati (con un esempio)
- Cosa sono le base di dati (terminologia e organizzazione)
- I vari modelli

Pre-requisiti e contatti

Pre-requisiti:

- Avere familiarità con un linguaggio di programmazione qualsiasi ed aver visto i FILES all'interno di quest'ultimo.
- **OPPURE**
- Aver seguito la **playlist del C o di Java** (ancora non disponibile) sul mio canale YouTube

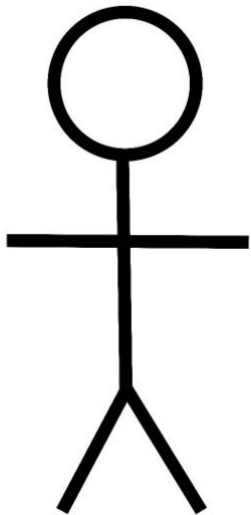
Informazioni di contatto:

- @specter915 su Telegram
- Commenti di YouTube

Materiale del corso

Tutto il materiale visto su YouTube è disponibile su Github incluse le slides: https://github.com/Hicortab/Playlist_Base_Dati

Esempio di quando usare un database



Giovanni

Giovanni dovrebbe creare un'applicazione per gestire una libreria. In particolare l'applicazione dovrà immagazzinare informazioni riguardo:

- Chi prende in prestito i libri e per quanto
- Chi restituisce i libri
- ecc...

Problema:

- Dove dovrebbe salvare queste informazioni?

Soluzione 1 (terribile)



Soluzione (gestione_libreria_soluzione_1.c):

Fare in modo che le informazioni siano gestite da delle variabili

Contro:

- Se il programma dovesse crashare per qualunque ragione perdi ogni tipo di informazione che risiedeva nelle variabili
- Se più computer dovessero utilizzare il programma le modifiche del 1° non sarebbero sincronizzare con il 2° ecc...
- **Probabilmente non otterrai mai più un lavoro in ambito informatico**

Soluzione 2 (meno terribile)

Soluzione (gestione_libreria_soluzione_2.c):

Salvare ogni singola informazione su un file di testo e recuperare le informazioni all'occorrenza

Contro:

- Il file che avresti in locale dovrebbe essere condiviso da altri computer se la libreria ha bisogno di avere più computer che allo stesso momento immagazzinano informazioni



ATTENZIONE



- **Politiche di scheduling:** se nello stesso momento 2 computer tentano di accedere al file l'applicazione crasherà creando disservizi

Soluzione 3 (seria)



Soluzione: Utilizzare le base di dati!

Risolvendo i problemi precedenti:

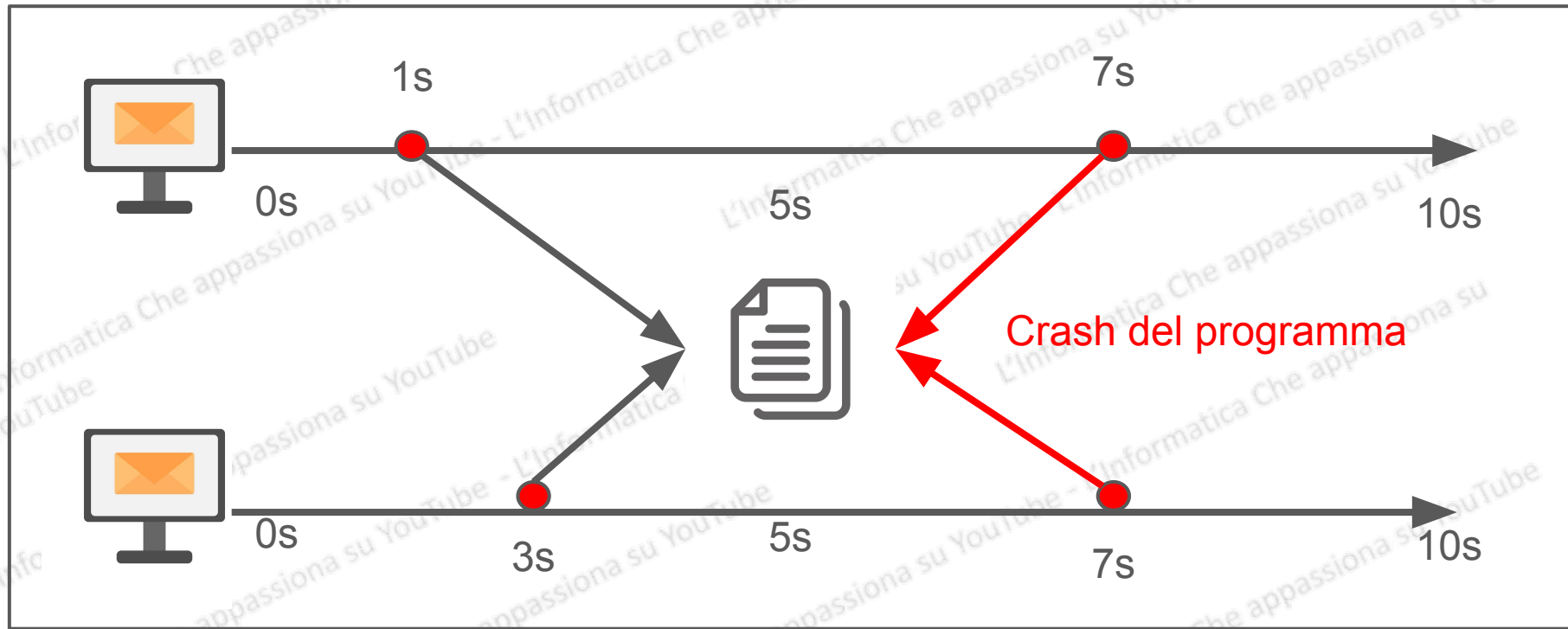
- Il problema che diversi computer non potevano condividere e accedere alle stesse informazioni (**caratteristica 1**)
- Non devi gestire direttamente gli accessi ai file che potrebbe essere complicato (**caratteristica 2**)
- Se chiudi il programma (o crasha) non perdi le informazioni

Caratteristica 1



Grazie alle proprietà della base di dati vedremo
che le informazioni saranno **sempre disponibili a
tutti i computer, contemporaneamente**

Caratteristica 2



Se un computer X richiede nello stesso momento la stessa risorsa del computer Y e il programma non prevede ciò, **potrebbe crashare**

Cosa sono le base di dati?



La base di dati (o **database**) altro non è che un insieme di informazioni.

Simbolo grafico per indicare i DataBase

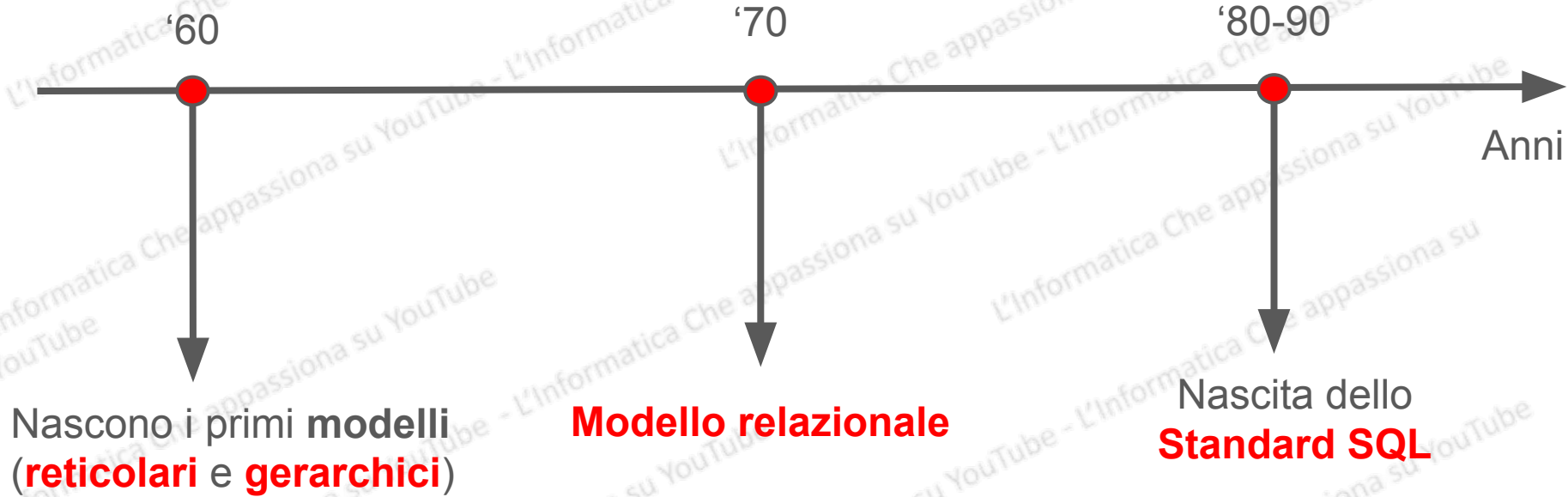
Una base dati in particolare, serve a:

- Immagazzinare
- Organizzare
- Condividere



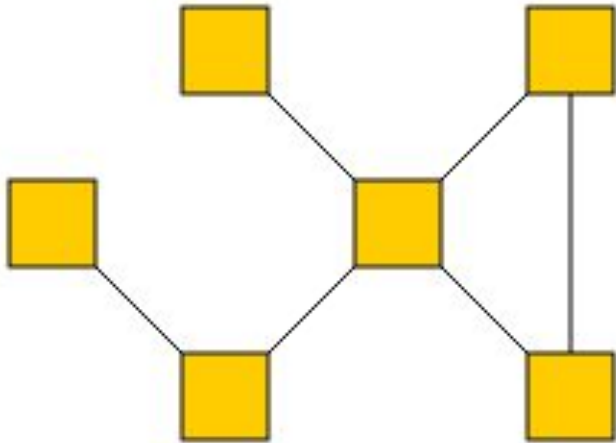
MOLTI
dati

I vari modelli

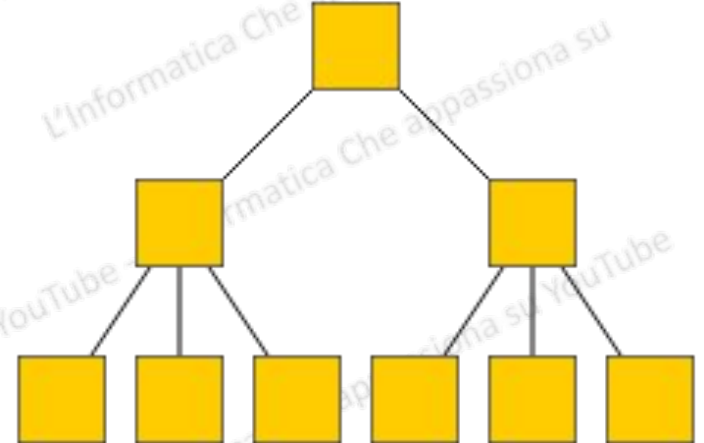


A cosa serve un modello?

Un **modello nella base di dati** serve principalmente a definire **come i dati sono organizzati, rappresentati e gestiti** all'interno di un sistema di database.

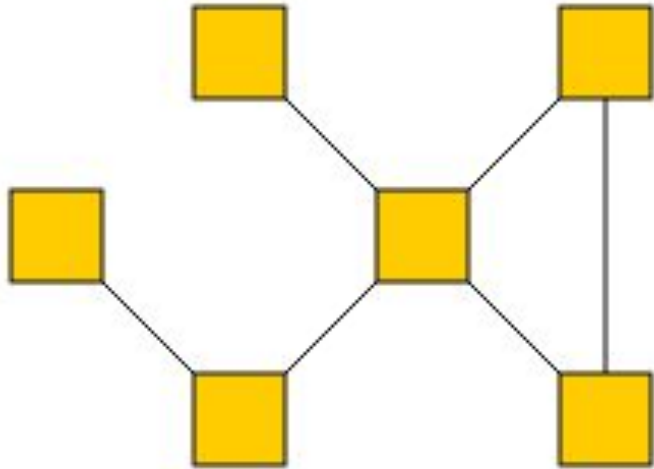


Reticolare



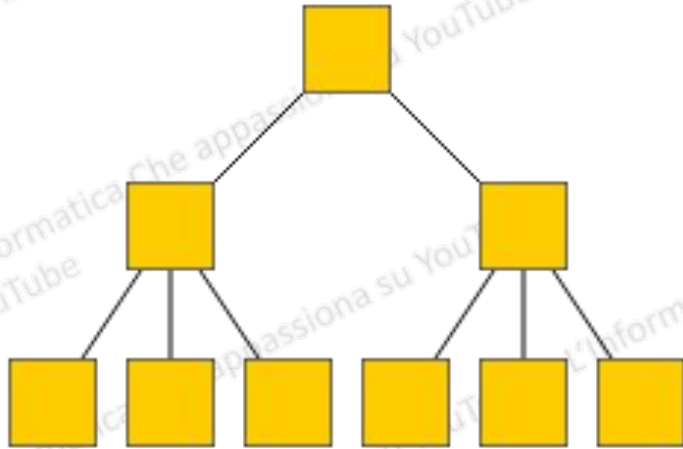
Gerarchico

Modello Reticolare



- Le strutture di collegamento devono essere **definite a priori**.
- **Modificare lo schema** (ad esempio aggiungere nuove relazioni) può essere **difficile e costoso**.

Modello Gerarchico



Gerarchico

- Organizza le informazioni come padre-figlio
- Non tutte le informazioni possono essere organizzate così però

Modello Relazionale



- Organizza i dati in **tabelle bidimensionali**
- Schema basato su **chiavi** (vedremo più tardi)
- **Relazioni tra varie informazioni** in maniera granulare e flessibile

Struttura di un database

	A	B	C	D	E	F
1		www.valterborsato.it				
2		data	Venditore	Regione	Settore	Fatturato
3		venerdì 15 gennaio 2016	Verdi	Friuli	Cancelleria	€ 1.650,00
4		venerdì 15 gennaio 2016	Bianchi	Lombardia	Cancelleria	€ 750,00
5		venerdì 15 gennaio 2016	Neri	Veneto	Cancelleria	€ 280,00
6		lunedì 18 gennaio 2016	Rossi	Lombardia	Informatica	€ 10.160,00
7		lunedì 18 gennaio 2016	Verdi	Veneto	Cancelleria	€ 302,00
8		lunedì 18 gennaio 2016	Bianchi	Lombardia	Informatica	€ 2.240,00
9		martedì 19 gennaio 2016	Verdi	Lombardia	Cancelleria	€ 840,00
10		mercoledì 20 gennaio 2016	Bianchi	Trentino	Informatica	€ 6.420,00
11		giovedì 21 gennaio 2016	Rossi	Veneto	Cancelleria	€ 2.840,00
12		venerdì 22 gennaio 2016	Bianchi	Friuli	Cancelleria	€ 1.420,00
13		lunedì 25 gennaio 2016	Verdi	Friuli	Cancelleria	€ 350,00
14		lunedì 25 gennaio 2016	Bianchi	Veneto	Cancelleria	€ 210,00
15		lunedì 25 gennaio 2016	Rossi	Lombardia	Cancelleria	€ 2.900,00

Nel **database** le nostre informazioni verranno **organizzate sotto forma di tabelle** come quella affianco.

Esempio di database

Scuola

Studente

Professore

...

Matricola	Nome	Cognome
12345	Luca	Bianchi
23456	Maria	Rossi
34567	Giulia	Verdi

IDProfessore	Nome	Cognome
1	Marco	Ferri
2	Anna	Conti
3	Paolo	Neri

Terminologia

Scuola



Questo è il **database** che conterrà le nostre tabelle

Studente

Professore

...

Tabelle con all'interno le informazioni

