

Database (principi teorici)

Liceo G.B. Brocchi

Classi quarte Scientifico - opzione scienze applicate

Bassano del Grappa, Febbraio 2023

Prof. Giovanni Mazzocchin

Cos'è un *database*

- 1) Tutti interagiamo con i database quotidianamente: operazioni quali *visualizzare il proprio estratto conto tramite un sistema home banking, o prenotare un biglietto aereo, o fare acquisti online* richiedono sicuramente degli accessi ad uno o più database. Un esempio più «scolastico» potrebbe essere il *registro elettronico*
- 2) L'evoluzione dei database ha costituito una parte fondamentale nel progresso dell'informatica: immaginate quanto potesse essere difficile e scomodo gestire i dati anagrafici di un comune in forma cartacea...
- 3) Gli esempi elencati sopra possono essere considerati **applicazioni tradizionali** dei database, in quanto l'informazione memorizzata nei database è prevalentemente testuale o numerica
- 4) Ci sono altri esempi di applicazioni più moderne dei database: la gestione dei **big data**, i **sistemi informativi geografici**, le **data warehouse** ... li vedrete all'università

Cos'è un *database*

1. Un **database** è una raccolta di dati logicamente correlati
2. Esempio: una rubrica telefonica.
3. Al giorno d'oggi, una rubrica telefonica può essere memorizzata dal database interno al telefono, su un computer tramite un foglio di calcolo *Excel*, oppure tramite *Microsoft Access* etc...
4. Un database, per essere considerato tale, deve avere queste proprietà:
 1. non deve essere una raccolta di dati casuali privi di significato
 2. deve rappresentare un aspetto della **realtà**: ad esempio, i dati relativi alle analisi del sangue effettuate in un determinato ospedale
 3. deve essere progettato per un utilizzo da parte di qualcuno (persona) o qualcosa (applicazione software)
 4. i cambiamenti che avvengono nella realtà devono riflettersi nel database: ad esempio, l'arrivo di un nuovo dipendente in un'azienda deve riflettersi nell'inserimento dei dati relativi al dipendente all'interno di un database aziendale

Cos'è un *database*

1. La dimensione di un database può andare dai pochi *kilobyte* di una rubrica telefonica al numero di *byte* inimmaginabile necessario per la gestione di tutti i dati di *Amazon*
2. Chiaramente, i dati di *Google*, *Facebook* o *Amazon* sono memorizzati su centinaia/migliaia di computer (*server*) sparsi nei loro *data center* in giro per il mondo
3. I database che ci interessano non sono gestiti manualmente, ma da *applicazioni software* particolari

DBMS

- Un **DBMS** è un sistema software estremamente complesso e *general-purpose*, che permette la creazione e la manutenzione di uno o più database. È general-purpose perché permette di creare qualsiasi tipo di database
- Un DBMS permette:
 - la **definizione** del database, che consiste nello stabilire quali sono le strutture impiegate per memorizzare i dati e i vincoli che intercorrono tra di essi
 - la **costruzione** del database, ossia l'inserimento dei dati
 - la **manipolazione** del database, ossia la modifica dei dati precedentemente memorizzati
 - la **condivisione** del database tra diversi utenti e programmi
- I programmi interagiscono con un database tramite:
 - **query**: permettono di leggere i dati in base ad alcuni criteri
 - **transazioni**: permettono di scrivere su un database, ossia di modificarne alcuni dati

Modellazione di alcune realtà

- Un database per la gestione delle informazioni relative alla realtà «Liceo Brocchi» necessita sicuramente, tra le altre cose, di:
 - informazioni anagrafiche degli studenti
 - informazioni anagrafiche dei docenti e del personale ATA
 - informazioni relative ai dipartimenti e agli indirizzi
 - l'orario
 - la composizione dei consigli di classe: se, ad esempio il 02/12/2023 è previsto un consiglio straordinario per la classe 3B, la notifica dovrebbe arrivare soltanto ai docenti del consiglio della 3B
 - informazioni relative ai libri di testo adottati nella scuola per materia, per indirizzo, per anno, etc...

Modellazione di alcune realtà

- Possiamo già iniziare a descrivere le «cose» della realtà «scuola». Queste cose sono chiamate **entità**: in questo caso potremmo avere a che fare con le entità **studente, docente, classe, libro di testo, compito in classe**
- E se la realtà fosse un ospedale? Allora, molto probabilmente, le entità sarebbero: **paziente, personale medico, reparto, visita medica, referto**
- È evidente che le entità non saranno oggetti isolati, ma collegati
- Si dice che tra le entità sussistono delle **relazioni**, che vedremo in seguito

Operazioni su un database

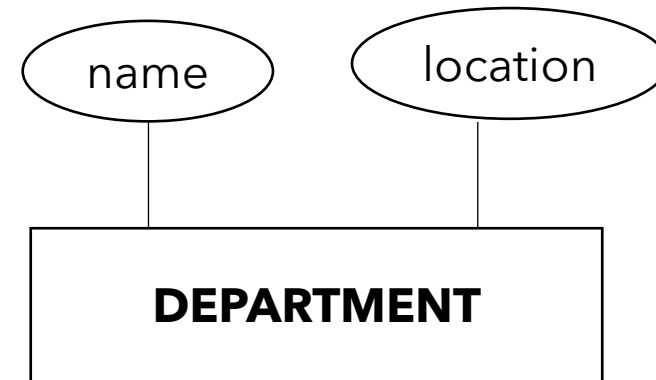
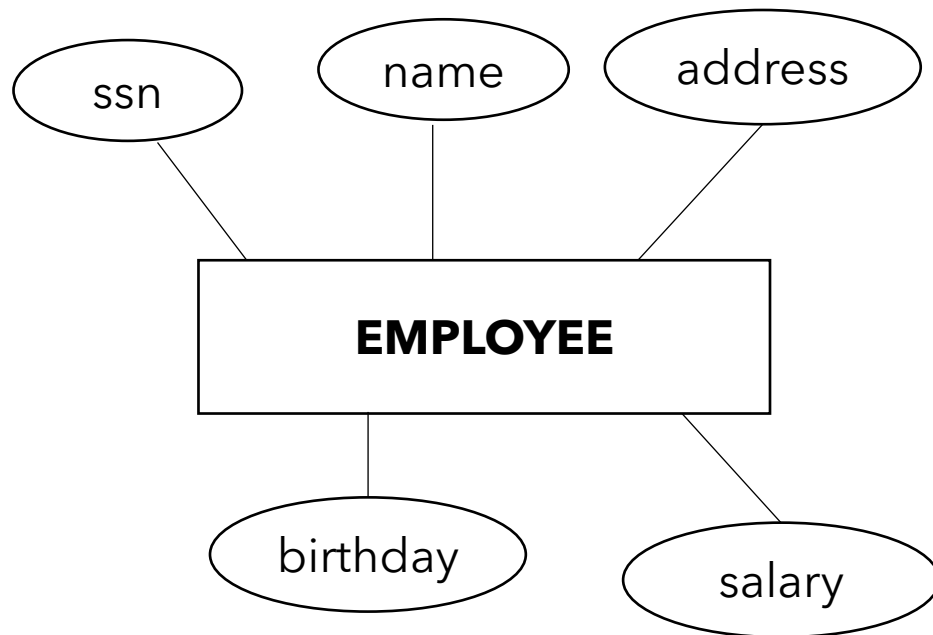
- Quali operazioni potremmo effettuare sul database del Liceo Brocchi?
 - cercare tutti gli studenti di cognome Rossi
 - cercare tutti gli studenti di una determinata classe
 - inserire un nuovo studente in una classe
 - cercare le anagrafiche di tutti i docenti di una determinata classe
 - cercare i compiti in classe di Matematica di una classe il cui voto è superiore alla media della classe
 - calcolare il numero di insufficienze in tutte le materie umanistiche allo Scientifico
 - calcolare il numero di insufficienze in tutte le materie scientifiche al Classico
 - estrarre i dati anagrafici dei docenti che insegnano sia allo Scientifico sia al Classico
 - calcolare il numero di studenti per comune di residenza
 - calcolare il numero di docenti per comune di residenza
 - trovare il comune di residenza con più studenti frequentanti
 - eliminare le informazioni relative ai docenti appena andati in pensione
 - cercare la classe con la media voti di Informatica più alta di tutta la scuola
 - aggiornare il numero di telefono di un docente

Alternative ai database

- Prima della nascita e dello sviluppo della scienza dei database, le informazioni relative ad una realtà venivano gestite da programmi specifici che manipolavano direttamente dei *file* (https://github.com/Cyofanni/high-school-cs-class/blob/main/C/file_IO/hospital/hospital_management.c)
- Probabilmente succedeva una cosa del genere:
 - l'ufficio «didattica» utilizzava un programma che scriveva e leggeva dei file contenenti le informazioni relative ai risultati scolastici degli studenti e ai docenti
 - l'ufficio «statistiche» utilizzava un programma che calcolava statistiche di vario tipo
 - chiaramente, entrambi gli uffici necessitavano delle informazioni relative agli studenti
 - quindi, l'ufficio didattica e l'ufficio statistiche avevano entrambi un file *studenti* contenente le stesse cose
- Questo approccio spreca risorse di *storage* e può portare a inconsistenze molto gravi. C'è infatti il rischio che l'ufficio didattica cambi un dato di uno studente e l'altro ufficio si dimentichi di effettuare la stessa modifica!
- Inoltre, la struttura dei file veniva specificata nel codice del programma. Per cambiare la struttura del file era dunque necessario cambiare il programma

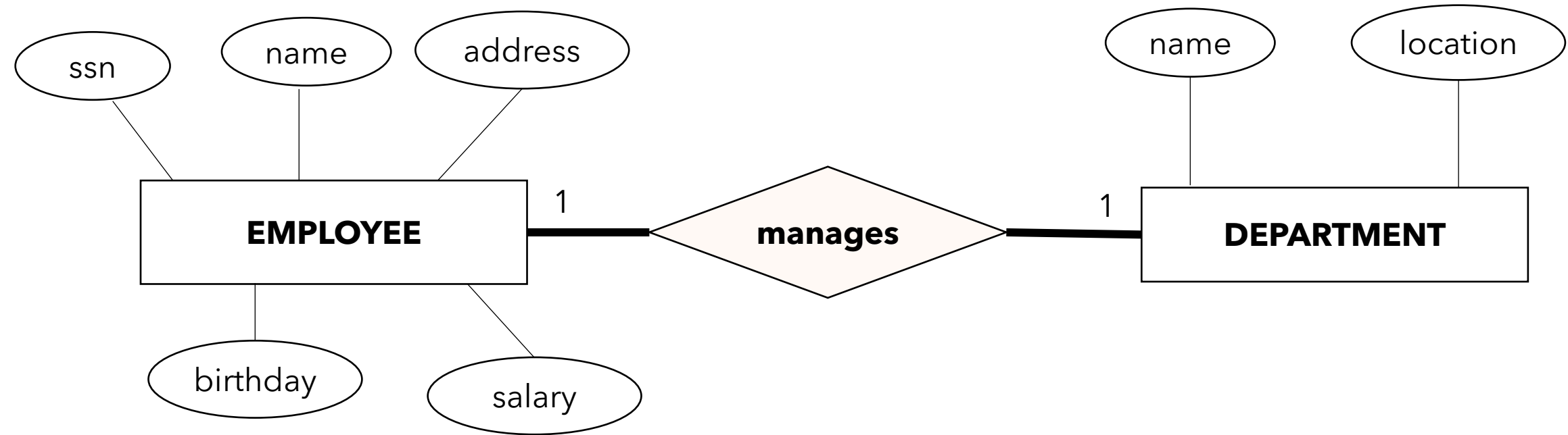
Diagrammi ER - Entità

- Possiamo descrivere una realtà utilizzando un diagramma **ER** (*Entity-Relationship*)
- Le **entità** sono le «cose» della realtà, mentre le **associazioni** (*relationship*) sono dei collegamenti logici tra le entità
- La progettazione dei diagrammi ER è chiamata **progettazione concettuale**
- Ecco degli esempi di entità per la realtà «azienda». Le proprietà delle entità sono dette «attributi»



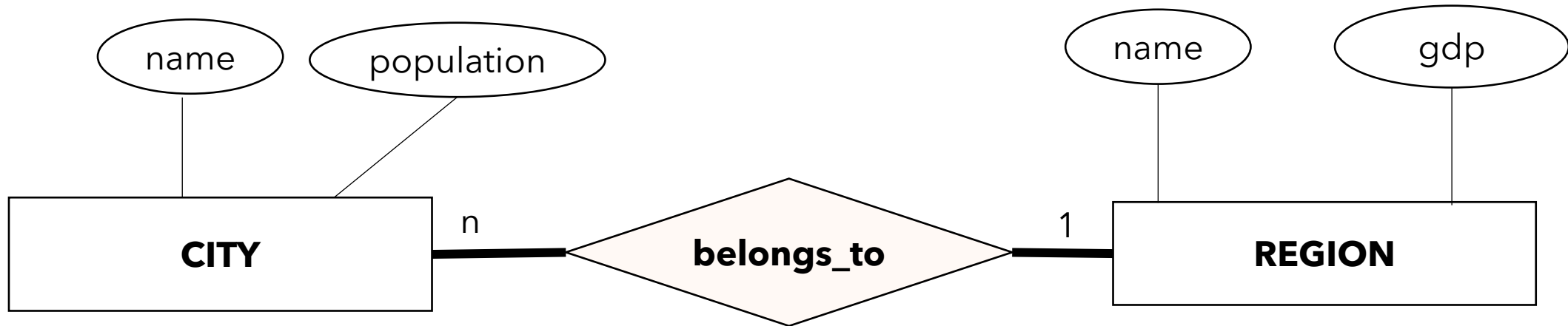
Diagrammi ER – associazione di cardinalità 1:1

- Entità: EMPLOYEE e DEPARTMENT
- Chiamiamo l'associazione con un verbo inglese alla terza persona singolare
- Intuitivamente: un impiegato «dirige» (*manages*) al più 1 dipartimento, e un dipartimento è diretto da minimo 1 impiegato e massimo 1 impiegato
- Questo tipo di associazione viene chiamato **1 a 1**



Diagrammi ER – associazione di cardinalità 1:n

- Entità: CITY, REGION
- Intuitivamente: una città fa parte di una regione, e una regione può contenere diverse città
- Questo tipo di associazione viene chiamato **1 a molti**



Diagrammi ER – associazione di cardinalità n:n

- Entità: DOCTOR, PATIENT
- Intuitivamente: un medico può seguire più pazienti, un paziente può essere seguito da più medici
- Questo tipo di associazione viene chiamato **molti a molti**

