LE BANCHE DATI

- Struttura di una banca dati biologica
- I database relazionali
- I database non relazionali
- API
- Tipi di dati
- Formato dei dati
- Sincronizzazione dei dati

LE BANCHE DATI

- Struttura di una banca dati biologica

Una banca dati biologica è organizzata in modo da permettere la raccolta, la gestione e la condivisione di **grandi quantità** di informazioni biologiche in modo strutturato e standardizzato. In genere, la struttura di una banca dati comprende diverse sezioni, tra cui:

1. **Entry**: ogni informazione biologica presente nella banca dati è rappresentata da una "entry" o record. Ogni entry è unica e identificata da un identificatore univoco, ad esempio un codice numerico o alfanumerico.

LE BANCHE DATI

- Struttura di una banca dati biologica

- Ogni **entry** in una banca dati deve avere un identificatore univoco che lo distingue dagli altri record nella stessa banca dati.

- Gli **identificatori univoci** sono utili perché consentono di riferirsi a un record specifico in modo inequivocabile, anche se ci sono altri record nella banca dati con informazioni simili.

LE BANCHE DATI

- Struttura di una banca dati biologica

2. **Annotazioni**: ogni entry è accompagnata da una serie di annotazioni, ovvero informazioni aggiuntive sulla natura dell'oggetto rappresentato dalla entry, come ad esempio la sua funzione biologica, la sua struttura molecolare, la sua localizzazione cellulare, la sua sequenza di nucleotidi o di amminoacidi, e così via.

Ad esempio, una banca dati di proteine potrebbe avere annotazioni che descrivono la funzione della proteina, la sua struttura tridimensionale e le sue interazioni con altre proteine.
Le annotazioni sono importanti perché forniscono **contesto** e aiutano a interpretare le informazioni contenute in un record.

LE BANCHE DATI

- Struttura di una banca dati biologica

3. **Cross-link:** le entry possono essere collegate tra loro attraverso le cross-link, ovvero riferimenti incrociati che permettono di stabilire relazioni tra diverse entry, come ad esempio l'associazione tra un gene e una proteina, o tra una malattia e un polimorfismo genetico.

- Le **relazioni** tra le entry in una banca dati si riferiscono al modo in cui i record sono correlati tra loro.

- I cross-link sono dei collegamenti tra le diverse entry di una banca dati che consentono di stabilire le relazioni tra di esse.

  Ad esempio, una proteina può essere associata a un gene attraverso un cross-link, o due proteine possono essere collegate tra loro in base alle loro interazioni note. I cross-link possono essere utilizzati per effettuare ricerche incrociate tra le entry e recuperare informazioni correlate.

LE BANCHE DATI

- Struttura di una banca dati biologica

Esistono cross-link sia all'interno della stessa banca dati che tra **diverse** banche dati.

- All'interno della stessa banca dati, i cross-link possono connettere le diverse voci o record, ad esempio collegando le sequenze alle relative annotazioni o metadati.

- Tra diverse banche dati, i cross-link possono collegare informazioni correlate tra loro, come ad esempio collegare una sequenza di DNA presente in una banca dati a una proteina corrispondente presente in un'altra banca dati. I

I cross-link possono essere utilizzati per integrare e combinare informazioni da diverse fonti e migliorare l'accessibilità dei dati.

LE BANCHE DATI

- Struttura di una banca dati biologica

4. **Struttura del database**: la struttura della banca dati è progettata per permettere l'accesso efficiente alle informazioni in essa contenute.
In genere, la banca dati è organizzata in modo **gerarchico**, con sezioni e sottosezioni in cui le informazioni sono suddivise in modo logico e coerente.

L'**organizzazione gerarchica** di una banca dati prevede una struttura a **livelli** che va dalla classe generale a quella più specifica.
In particolare, si parte da un livello superiore che contiene le categorie più generali di dati, per poi scendere a livelli sempre più specifici e dettagliati.

LE BANCHE DATI

- Struttura di una banca dati biologica

Ad esempio, una banca dati biologica potrebbe essere organizzata in questo modo:

- Livello 1: Categoria generale, ad esempio "Organismi"
- Livello 2: Sotto-categorie più specifiche, ad esempio "Animali", "Piante", "Funghi"
- Livello 3: Sotto-categorie ancora più specifiche, ad esempio "Mammiferi", "Rettili", "Angiosperme"
- Livello 4: Voci singole, ad esempio "Homo sapiens", "Mus musculus", "Arabidopsis thaliana"

In questo esempio, si parte dalla classe generale degli organismi, per poi scendere a livelli sempre più specifici, fino ad arrivare alle voci singole che rappresentano gli elementi di base della banca dati.

LE BANCHE DATI

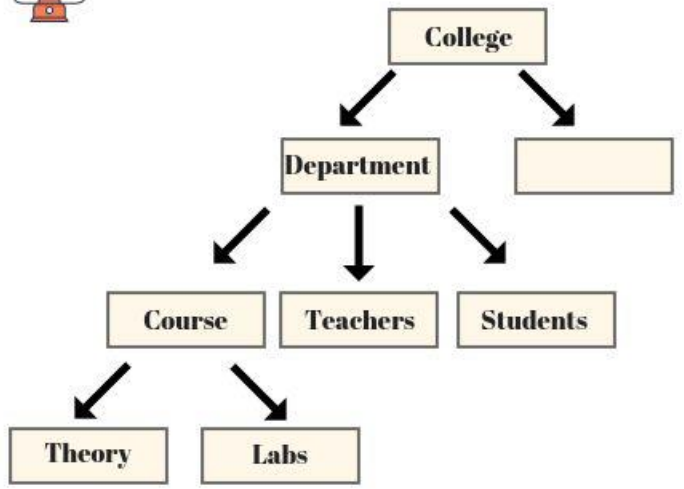
- Struttura di una banca dati biologica

- L'organizzazione gerarchica permette di suddividere i dati in modo logico e coerente, facilitando la ricerca e la consultazione delle informazioni all'interno della banca dati stessa.

Inoltre, consente di espandere e aggiornare la banca dati in modo semplice ed efficiente, aggiungendo nuove categorie e voci singole quando necessario.

- I database gerarchici hanno una **struttura semplice**, tipo schedari, i cui dati sono del tutto indipendenti tra loro.
- La modifica di un dato non ha nessuna influenza sui dati inseriti in altre tabelle/schedari. I db gerarchici non permettono di implementare procedure avanzate di interrogazione dei dati.

Esempio di DB gerarchico



LE BANCHE DATI

- Struttura di una banca dati biologica

5. **Interfaccia utente:** l'accesso alla banca dati avviene attraverso un'interfaccia utente, ovvero una piattaforma software che permette agli utenti di interrogare la banca dati, visualizzare le informazioni e scaricarle in vari formati.

L'interfaccia utente può essere personalizzata a seconda delle esigenze degli utenti e delle funzioni della banca dati.

LE BANCHE DATI

- I database relazionali

I database **gerarchici** sono considerati una forma di **database relazionale**, ma con una struttura gerarchica anziché tabellare.

I database relazionali sono un tipo di database basati sulla teoria delle relazioni matematiche.

Essi offrono una struttura organizzata e flessibile per l'archiviazione e l'accesso ai dati, attraverso l'uso di tabelle collegate tra loro in modo logico e coerente.

LE BANCHE DATI

- I database relazionali

Le caratteristiche principali dei database relazionali includono:

- Struttura tabellare: i dati sono organizzati in tabelle, che sono composte da **righe** e **colonne**.

- Relazioni tra le tabelle: le tabelle sono collegate tra loro attraverso **chiavi primarie** e **chiavi esterne**, che creano relazioni tra i dati delle diverse tabelle.

- Integrità dei dati: i database relazionali sono progettati per garantire **l'integrità** dei dati, attraverso l'uso di vincoli che impediscono l'inserimento di dati non validi o la modifica di dati esistenti in modo non corretto.

- Accesso ai dati: i database relazionali consentono l'accesso ai dati tramite il linguaggio di interrogazione strutturato (**SQL**).

LE BANCHE DATI

- I database relazionali

Nell'esempio in figura le chiavi sono i valori dei campi "Roll_no" (1,2,3) nelle tabelle "Studenti" e "Voti", e i valori del campo "Course_ID" (IT_101, IT_102, IT_103) nella tabella Corsi.

Modello relazionale in un DBMS

Studenti

Roll_No	Name
1	Arya
2	Bran
3	Jon

Corsi

Course_ID	Sub_Name
IT_101	Java
IT_102	C++
IT_103	DBMS

Voti

Roll_No	Course_ID	Marks
1	IT_101	100
1	IT_102	99
3	IT_103	95

LE BANCHE DATI

- I database relazionali

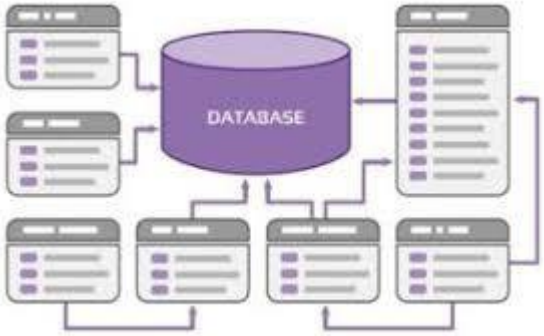
Il modello relazionale è attualmente il modello piu' usato.
La struttura principale di questo tipo di database è una relazione cioè una tabella bidimensionale composta da righe e da colonne. Le tabelle quindi sono le componenti chiave di questo tipo di database.
Una tabella consiste di righe (record). Ogni riga e' divisa in campi (colonne o fields) che hanno un certo formato dei dati.

ID gene	Gene Name	Unigene	RefSeq
101	Gata1	Hs.765	NM_002049
102	Gata2	Hs.367725	NM_032638

righe

campi

LE BANCHE DATI



- I database relazionali

La posizione fisica dei record o dei campi in una tabella e' ininfluente e ogni record di una tabella viene identificato da un campo che contiene un valore univoco (**chiave**).

All'utente quindi non e' richiesta alcuna conoscenza della posizione specifica di un record per poterne recuperare i relativi dati.

Il modello relazionale suddivide le relazioni in:
- uno a uno
- uno a molti
- molti a molti

Una relazione tra due tabelle viene implicitamente stabilita tramite i valori corrispondenti di un
campo condiviso.

LE BANCHE DATI

- I database relazionali

ID gene	Gene Name	Unigene	RefSeq	Chromosomal range
101	Gata1	Hs.765	NM_002049	Chr X: 48529906 - 48537662; strand + ;
102	Gata2	Hs.367725	NM_032638	Chr 3: 129680955 - 129695055; strand - ;
103	tp53	Hs.654481	NM_001126117 NM_001126116 NM_001126115 NM_001126114 NM_001126113 NM_001126112 NM_000546	Chr 17: 7505822 - 7591285; strand - ;

Gene

ID protein	ID gene	Protein Name	Accession
95	101	GATA1 protein	AAH09797
96	102	GATA2 protein	AAH51272
97	103	p53	

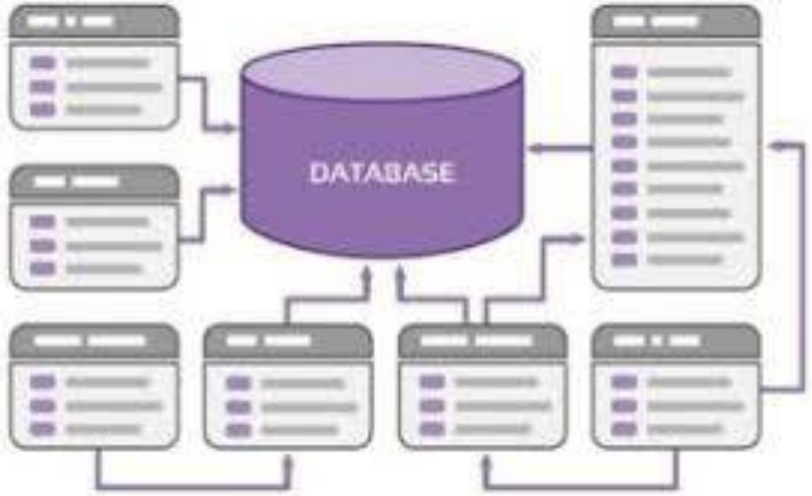
Protein

LE BANCHE DATI

- I database relazionali

Teminologia dei DB relazionali

- Termini relativi ai valori (dati, informazioni, null.)
- Termini relativi alla struttura (tabella, campo, record, chiave.)
- Termini relativi alle relazioni (relazioni uno a uno, uno a molti, ecc.)
- Termini relativi all'integrita` (integrita` di entita`, di dominio, referenziale, ecc.)



LE BANCHE DATI

- I database relazionali

Termini relativi ai valori:

Dati
I valori che archiviate nel database sono detti dati. I dati sono statici nel senso che conservano lo stesso stato finché non li modificate tramite un processo manuale o automatizzato.

Informazioni
Le informazioni sono ottenute dai dati che vengono elaborati in modo da poter essere resi significativi quando vengono utilizzati.
Le informazioni sono dinamiche nel senso che si modificano continuamente in relazione ai dati archiviati nel database o anche perché possono essere elaborate e presentate in un numero illimitato di modi.

I dati sono ciò che archiviate; le informazioni sono ciò che recuperate!!!

LE BANCHE DATI

- I database relazionali

Null

Un null rappresenta un valore sconosciuto o mancante. Quindi non rappresenta né' uno zero ne' una stringa di testo di uno o più' spazi vuoti. Il lato positivo dell'utilizzo dei valori null è che permettono di estrarre record con eventuali valori di campo sconosciuti o mancanti.

Ad esempio dei geni non ancora annotati a livello di Refseq possono essere selezionati per effettuare un aggiornamento con i dati di NCBI. Il principale inconveniente dei null e' che essi hanno un effetto negativo sulle operazioni aritmetiche: un'operazione in cui uno dei termini e' null da un risultato null.

ID protein	ID gene	Protein Name	Accession
95	101	GATA1 protein	AAH09797
96	102	GATA2 protein	AAH51272
97	103	p53	*NULL*

LE BANCHE DATI

- I database relazionali

Tabella

Secondo il modello relazionale i dati sono archiviati in relazioni che l'utente percepisce come tabelle.

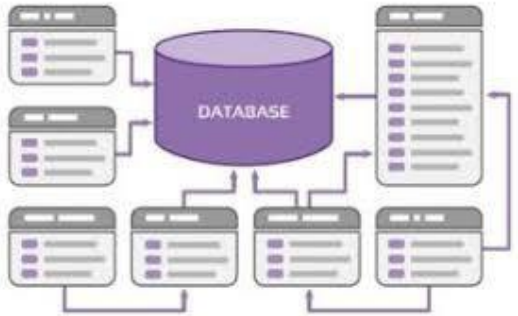
Ogni relazione e' composta da tuple (record) e attributi (campi).

ID gene	Gene Name	Unigene	RefSeq	Chromosomal range
101	Gata1	Hs.765	NM_002049	Chr X: 48529906 - 48537662; strand + ;
102	Gata2	Hs.367725	NM_032638	Chr 3: 129680955 - 129695055; strand - ;
103	tp53	Hs.654481	NM_001126117 NM_001126116 NM_001126115 NM_001126114 NM_001126113 NM_001126112 NM_000546	Chr 17: 7505822 - 7591285; strand - ;

record

campi

LE BANCHE DATI



- I database relazionali

Tabella
Ogni tabella rappresenta uno specifico oggetto (entità: ad es. gene or protein).

• L'ordine logico dei record e dei campi all'interno di una tabella è assolutamente ininfluente;
• Ogni tabella contiene almeno un campo chiamato chiave primaria, che identifica in modo univoco ciascuno dei suoi record (nell'esempio dell'ultima tabella ID gene).

Grazie a queste due caratteristiche i dati di un database relazionale possono esistere indipendentemente dal modo in cui sono fisicamente archiviati nel computer.

LE BANCHE DATI

- I database relazionali

Campo
I campi sono le strutture che effettivamente archiviano i dati.
I dati dei campi possono essere recuperati e presentati come informazioni in un enorme numero di configurazioni.

Ogni campo di un database progettato correttamente contiene un solo valore, il cui tipo viene identificato dal nome del campo.

Quindi sono da evitare:
- i campi compositi, il cui valore contiene due o più voci diverse;
- i campi multivalore, che contengono più' istanze dello stesso tipo di valore;
- i campi calcolati, che contengono un valore di testo concatenato o il risultato di un'espressione aritmetica

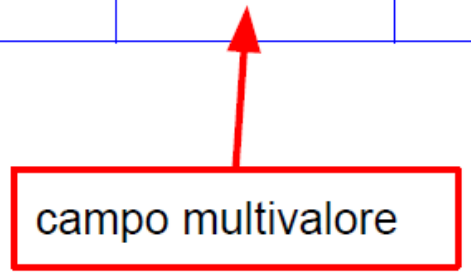
LE BANCHE DATI

- I database relazionali

Quindi sono da evitare:

- i campi multivalore, che contengono piu' istanze dello stesso tipo di valore

ID gene	Gene Name	Gene lenght	RefSeq	Chromosomal range
101	Gata1	7757	NM_002049	Chr X: 48529906 - 48537662; strand + ;
102	Gata2	14099	NM_032638	Chr 3: 129680955 - 129695055; strand - ;
103	tp53	85462	NM_001126117 NM_001126116 NM_001126115 NM_001126114 NM_001126113 NM_001126112 NM_000546	Chr 17: 7505822 - 7591285; strand - ;



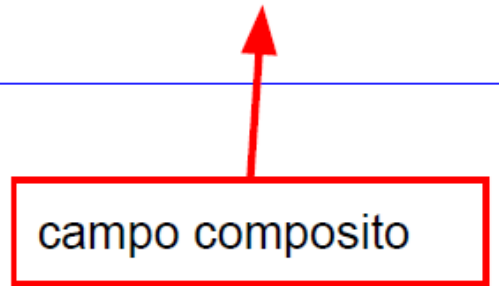
LE BANCHE DATI

- I database relazionali

Sono da evitare

· i campi compositi, il cui valore contiene due o piu voci diverse

ID gene	Gene Name	Gene lenght	RefSeq	Chromosomal range
101	Gata1	7757	NM_002049	Chr X: 48529906 - 48537662; strand + ;
102	Gata2	14099	NM_032638	Chr 3: 129680955 - 129695055; strand - ;
103	tp53	85462	NM_001126117 NM_001126116 NM_001126115 NM_001126114 NM_001126113 NM_001126112 NM_000546	Chr 17: 7505822 - 7591285; strand - ;



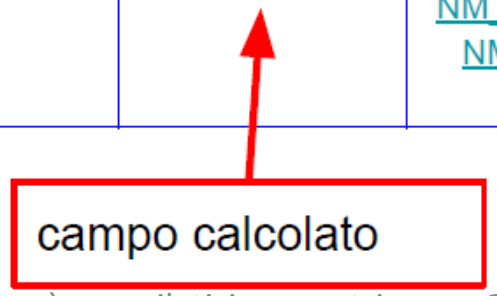
LE BANCHE DATI

- I database relazionali

Sono da evitare

• i campi calcolati, poiché' a seguito di un aggiornamento della banca dati i campi calcolati non si aggiornano automaticamente.

ID gene	Gene Name	Gene lenght	RefSeq	Chromosomal range
101	Gata1	7757	NM_002049	Chr X: 48529906 - 48537662; strand + ;
102	Gata2	14099	NM_032638	Chr 3: 129680955 - 129695055; strand - ;
103	tp53	85462	NM_001126117 NM_001126116 NM_001126115 NM_001126114 NM_001126113 NM_001126112 NM_000546	Chr 17: 7505822 - 7591285; strand - ;



LE BANCHE DATI

- I database relazionali

Termini relativi alla struttura:

Record
Un record rappresenta un'istanza univoca del soggetto di una tabella.
Esso è composto di tutto l'insieme di campi della tabella a prescindere dal fatto che i campi contengano o meno valori.
Ogni record e' identificato in tutto il database da un valore univoco nel campo della chiave primaria di quel record.

chiave primaria

ID gene	Gene Name	Unigene	RefSeq
101	Gata1	Hs.765	NM_002049
102	Gata2	Hs.367725	NM_032638
103	tp53	Hs.654481	NM_001126117

LE BANCHE DATI

- I database relazionali

Chiavi
Le chiavi sono campi speciali che hanno ruoli specifici all'interno di una tabella.
Una tabella può' contenere più' tipi di chiavi, ma i due tipi più' importanti sono la chiave primaria e la chiave esterna.
Una chiave primaria e' un campo o un gruppo di campi, che identifica in modo univoco ciascun record della tabella;

se una chiave primaria e' composta di due o più' campi viene detta composita.

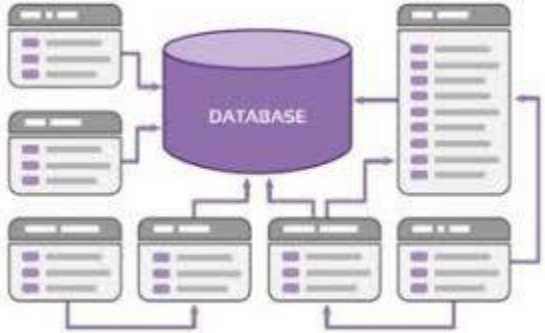
- Il valore di una chiave primaria identifica un record specifico nell'intero database.
- Il campo della chiave primaria identifica una determinata tabella nell'intero database

LE BANCHE DATI

- I database relazionali

Chiavi
Le chiavi esterne servono a stabilire le relazioni tra coppie di tabelle.
I record di entrambi le tabelle saranno correlati sempre correttamente se i valori di una chiave esterna corrispondono ai valori esistenti di una chiave primaria a cui la chiave esterna fa riferimento.

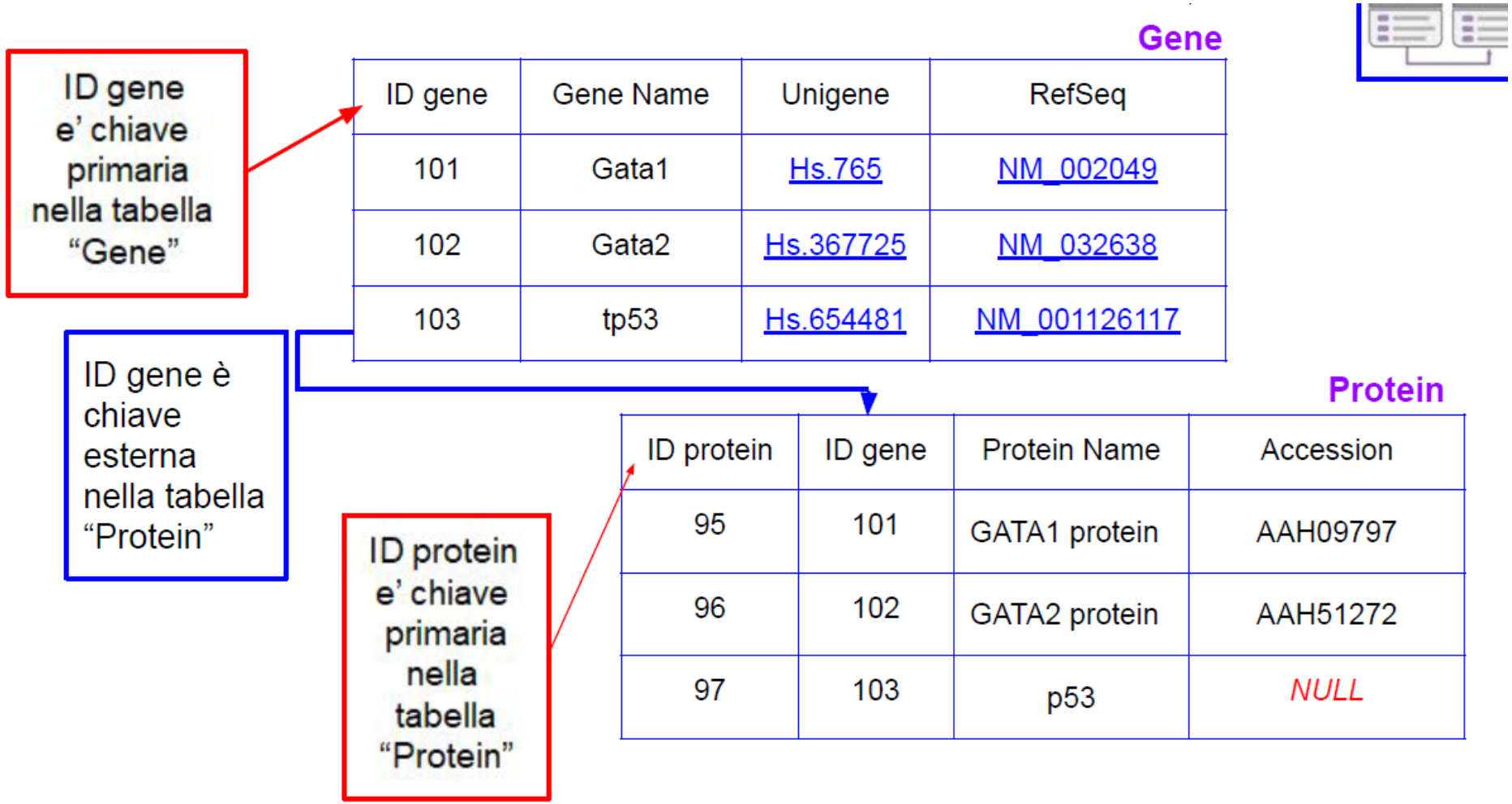
Quindi, quando viene popolato un database, dovranno essere riempite prima le tabelle la cui chiave primaria funge da chiave esterna per un'altra tabella.



LE BANCHE DATI

- I database relazionali

Ogni tabella di un database deve avere una chiave primaria!



Gene

ID gene	Gene Name	Unigene	RefSeq
101	Gata1	Hs.765	NM_002049
102	Gata2	Hs.367725	NM_032638
103	tp53	Hs.654481	NM_001126117

Protein

ID protein	ID gene	Protein Name	Accession
95	101	GATA1 protein	AAH09797
96	102	GATA2 protein	AAH51272
97	103	p53	*NULL*

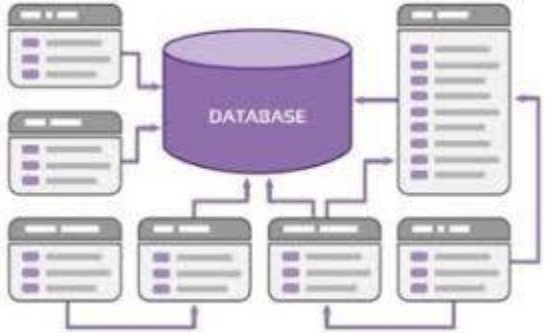
LE BANCHE DATI

- I database relazionali

Indici
Un indice e' una struttura fornita da un RDBMS per migliorare i tempi di ricerca (query) dei dati.
Se una tabella non ha indici, ogni ricerca obbliga il sistema a leggere tutti i dati presenti in essa.
L'indice consente invece di ridurre l'insieme dei dati da leggere per completare la ricerca.

In ogni caso un indice non ha nulla a che fare con la struttura logica del database!

Le chiavi sono strutture logiche utilizzate per identificare i record all'interno di una tabella, mentre **gli indici sono strutture ausiliarie di accesso ai dati utilizzate per ottimizzare l'elaborazione dei dati**.



LE BANCHE DATI

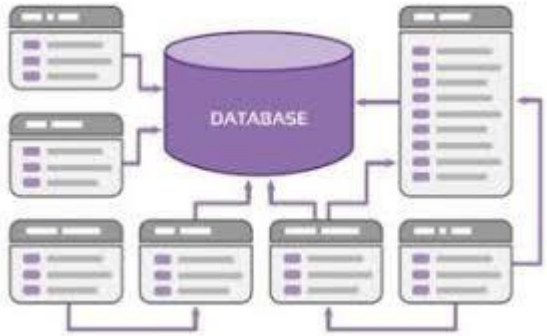
- I database relazionali

Termini relativi alle relazioni:

Una relazione e' una componente importante di un database relazionale.

• Essa vi consente di creare visualizzazioni composte di piu' tabelle.
• E' fondamentale per l'integrità' dei dati, poiché' aiuta a ridurre i dati ridondanti ed elimina i dati duplicati.

Come abbiamo già introdotto tre sono i tipi specifici di relazione (comunemente detti cardinalità') che possono esistere tra due tabelle:
• uno a uno
• uno a molti
• molti a molti



LE BANCHE DATI

- I database relazionali

Relazioni uno a uno:
Due tabelle sono associate da una relazione di tipo uno a uno quando un solo record della prima tabella e' correlato ad un solo record della seconda ed un solo record della seconda tabella e' correlato ad un solo record della prima.

Per stabilire la relazione, bisogna prendere una copia della chiave primaria della tabella padre e incorporarla all'interno della struttura della tabella figlio.

Questo tipo di relazione e' particolare perché è l'unico tipo in cui entrambe le tabelle possono condividere la stessa chiave primaria.

ID gene	Gene Name	Unigene	RefSeq
101	Gata1	Hs.765	NM_002049
102	Gata2	Hs.367725	NM_032638
103	tp53	Hs.654481	NM_001126117

ID gene	Protein Name	Accession
101	GATA1 protein	AAH09797
102	GATA2 protein	AAH51272
103	p53	*NULL*

Esempio di relazione di tipo uno a uno.

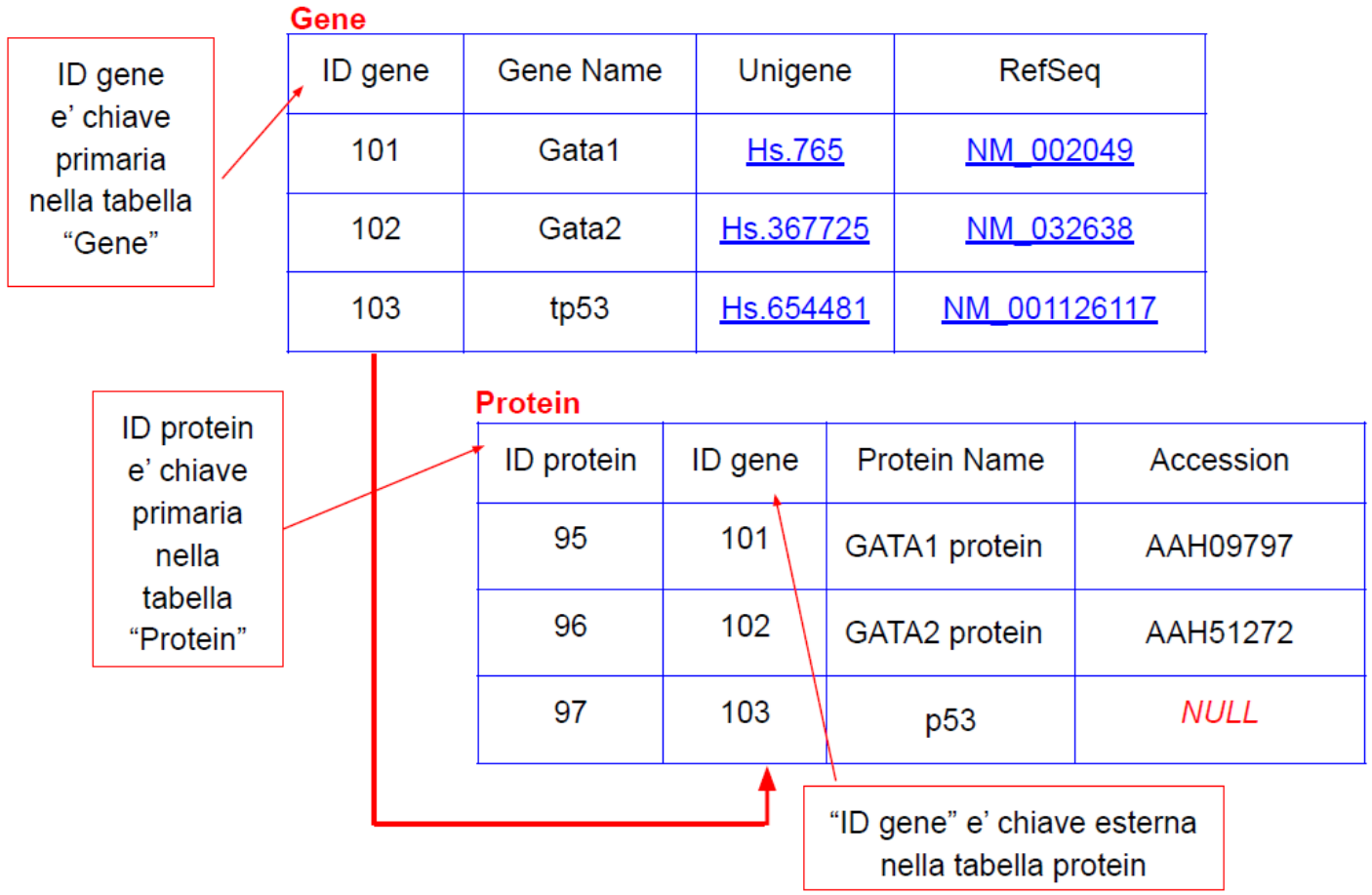
LE BANCHE DATI

- I database relazionali

Relazioni uno a molti

Due tabelle sono associate da una relazione di tipo uno a molti quando un singolo record della prima tabella può' essere collegato a molti record della seconda tabella ma un singolo record della seconda tabella può' essere collegato invece soltanto ad un record della prima tabella.

Per stabilire una relazione di tipo uno a molti dovete prendere una copia della chiave primaria della tabella padre e incorporarla nella struttura della tabella figlio, dove essa diventa chiave esterna.



LE BANCHE DATI

- I database relazionali

Relazioni molti a molti

Due tabelle sono associate da una relazione di tipo molti a molti quando un singolo record della prima tabella puo' essere collegato a molti record della seconda tabella e un singolo record della seconda tabella puo' essere collegato a molti record della prima tabella.

Per stabilire questa relazione occorre una tabella di collegamento. Essa facilita l'associazione dei record di una tabella con quelli dell'altra e aiuta a garantire che non ci siano problemi nell'aggiungere, eliminare o modificare i dati correlati.

Nella tabella di collegamento i due campi chiave hanno due ruoli distinti:

- insieme formano la chiave primaria composita della tabella di collegamento;
- separatamente fungono ciascuno da chiave esterna.

LE BANCHE DATI

- I database relazionali

Gene

ID gene	Gene Name
101	Gata1
102	Gata2
103	tp53

Relazione stabilita tra le due tabelle Gene e Organism con l'aiuto di una tabella di collegamento.

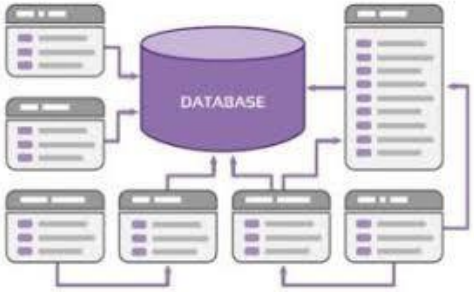
Organism

ID organism	Organism Name
2	Mus musculus
1	Homo sapiens
3	Rattus norvegicus

ID gene	ID organism
101	2
101	1
102	1
103	3

tabella di collegamento tra Gene e Organism

LE BANCHE DATI



- I database relazionali

Termini relativi all'integrità:

**L'integrita`** dei dati fa riferimento alla validita`, coerenza e precisione dei dati di un database. L'integrita' dei dati e' uno dei processi più' importanti nella fase di progettazione di un database.

Vi sono tre tipi di integrita` dei dati che si possono implementare nella progettazione di un database:

❏ l'integrità' a livello di tabella,
❏ l'integrità' a livello di campo,
❏ l'integrità' a livello di relazione.

Essi si basano su vari aspetti della struttura del database e sono denominati secondo il livello in cui operano.

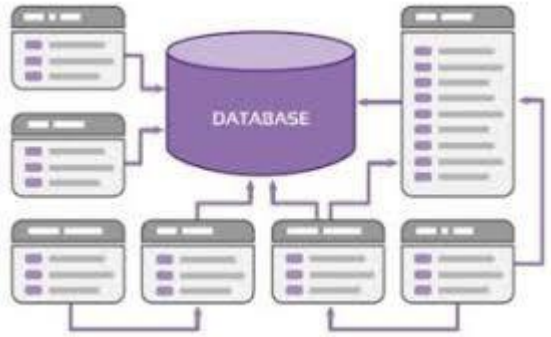
LE BANCHE DATI

- I database relazionali

L'integrità **a livello di tabella** (detta comunemente integrità di entità) assicura che non ci siano record duplicati all'interno della tabella e che il campo che identifica ciascun record sia univoco e mai null.

L'integrità **a livello di campo** assicura che la struttura di ogni campo sia solida; che i valori di ogni campo siano validi, coerenti e precisi, e che i campi dello stesso tipo siano definiti in modo coerente in tutto il database.

L'integrita' **a livello di relazione** (detta comunemente integrita' referenziale) assicura che la relazione tra una coppia di tabelle sia solida e che i record delle tabelle siano sincronizzati ogni qualvolta si inseriscono, aggiornano o eliminano dati da una delle due tabelle



LE BANCHE DATI

- I database relazionali

SQL (Structured Query Language)

Le richieste di informazioni da un database relazionale sono fatte sotto forma di query, che e' una richiesta "stilizzata".

L' SQL è il linguaggio standard utilizzato per creare, modificare, gestire ed eseguire ricerche nei database basati sul modello relazionale (RDBMS).

Di seguito e' riportato un esempio di query SQL per ottenere i dati relativi ad un gene la cui proteina codificata ha un accession ="AAH09797"

```
SELECT Gene Name, Unigene, Refseq
FROM Gene, Protein
WHERE Accession="AAH09797"  ORDER BY Unigene;
```

LE BANCHE DATI

- I database relazionali

```
SELECT Gene Name, Unigene, Refseq
FROM Gene, Protein
WHERE Accession="AAH09797"  ORDER BY Unigene;
```

❏ La clausola SELECT serve ad indicare i campi che si intendono utilizzare nella query.
❏ La clausola FROM indica la tabella o le tabelle a cui i campi appartengono.
❏ Le clausole WHERE e ORDER BY impongono i criteri per filtrare
i record restituiti.