

Approach to database design

Importance of database design

- Crucial to the success of a project
- Contributes to:
 - Integrity of data
 - No redundancy
 - Performance
 - User satisfaction
- Avoids costly problems



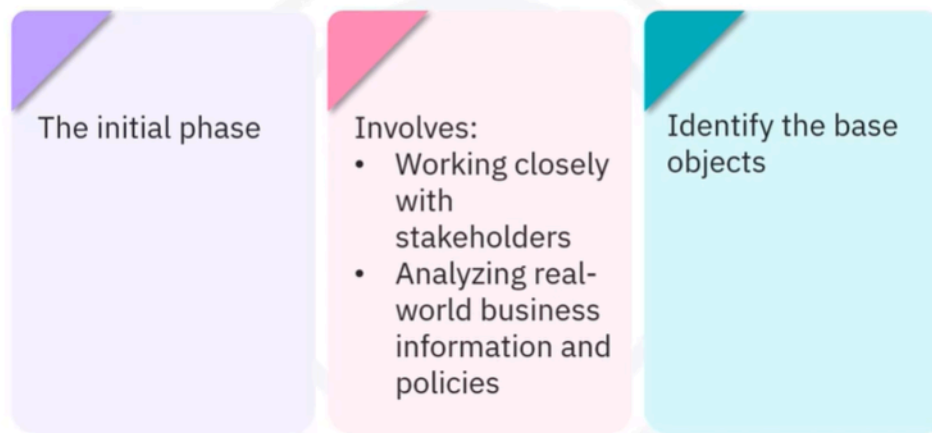
- well-designed database is crucial to the success of any data-driven project. The design of the database contributes to the integrity of the data, the reduction of redundant data, application performance, and user satisfaction, which are all key markers of a successful project. Therefore, it is essential that you spend the time upfront designing your database to avoid costly problems later.

Database design process



- There are three critical phases in the database design process, requirements analysis, logical design, and physical design.

Requirements analysis



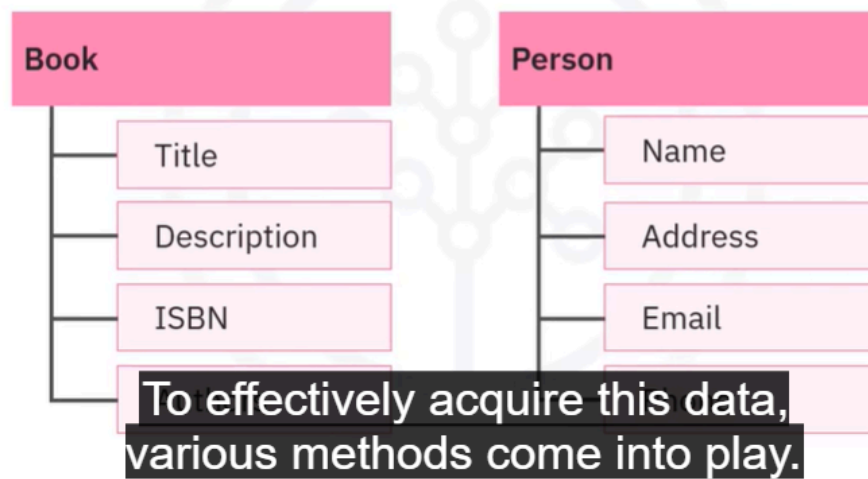
- Let us explore the steps in detail. First is the requirements analysis phase. This step is where you work closely with stakeholders to gather and analyze real-world business information and policies. You need to identify the base objects in the data and the relationships between these objects.

Exmaple

-

Requirements analysis

- Identify information associated with objects
- Example: Details of the book



UML Network

Data acquisition



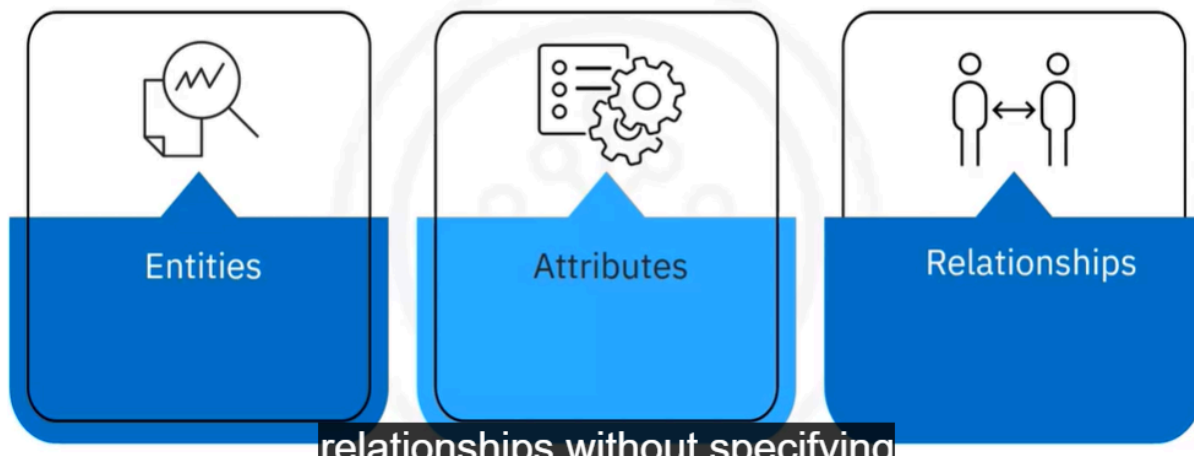
- To effectively acquire this data, various methods come into play. Let us explore a few of these methods. Data acquisition methods may involve reviewing existing data stores, electronic formats, or paper-based systems. When reviewing a current database, use it as a source of information, not a starting template. Interviewing users to determine current data usage, interviewing current and potential users to gather insights for possible improvements.

Requirement Analysis Output



- The output for your requirements analysis could be a report, a data diagram, or a presentation for stakeholder validation.

Logical design



Logical design

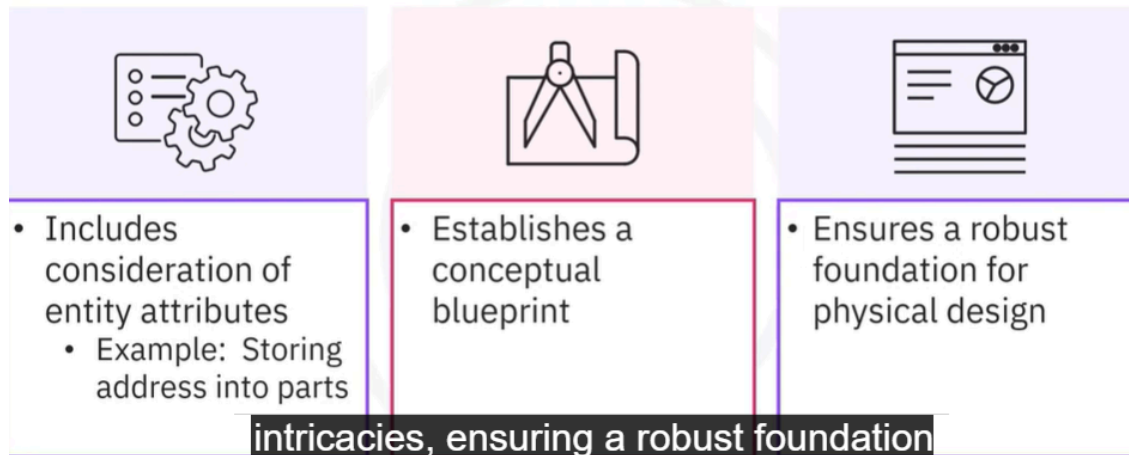
- Objects evolve into entities
- Ensure objects stand as entities
- Object characteristics become attributes

Book	Loan	Person
Title	Book title	First name
Description	Person first name	Last name
ISBN	Person last name	Street address
Authors	Loan date	City
	Return date	Email
		Phone

last name attributes in database terms

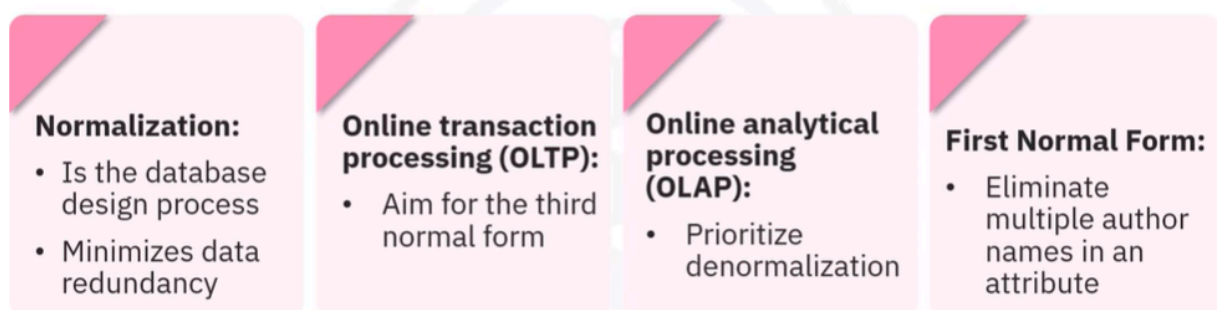
- Let us now discuss the logical design phase. The logical design phase involves transferring the output from the requirements analysis phase into entities, attributes, and relationships without specifying technical implementation details. In the logical design phase, the objects identified in the previous stage, like books and people, evolve into entities representing people, events, locations, or things. So the book object becomes a book entity, and the person object becomes a person entity. It is crucial to ensure that objects truly stand as entities and not characteristics of others. Object characteristics become attributes. For example, representing a person's name is more effective as separate first and last name attributes in database terms

Considering entity attributes

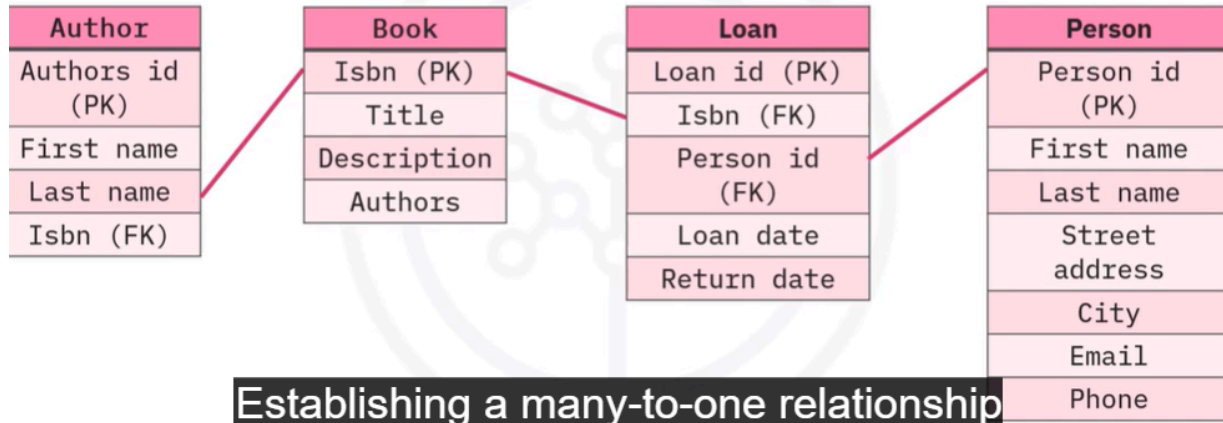


- The logical design also includes consideration of entity attributes. For instance, when dealing with a person's address, breaking it down into its constituent parts such as street name, city, state, and zip code can enhance the overall efficiency of data storage. By thinking about first and last names and breaking down addresses, the logical design phase establishes a conceptual blueprint without delving into technical intricacies, ensuring a robust foundation for subsequent physical design decisions.

Normalization

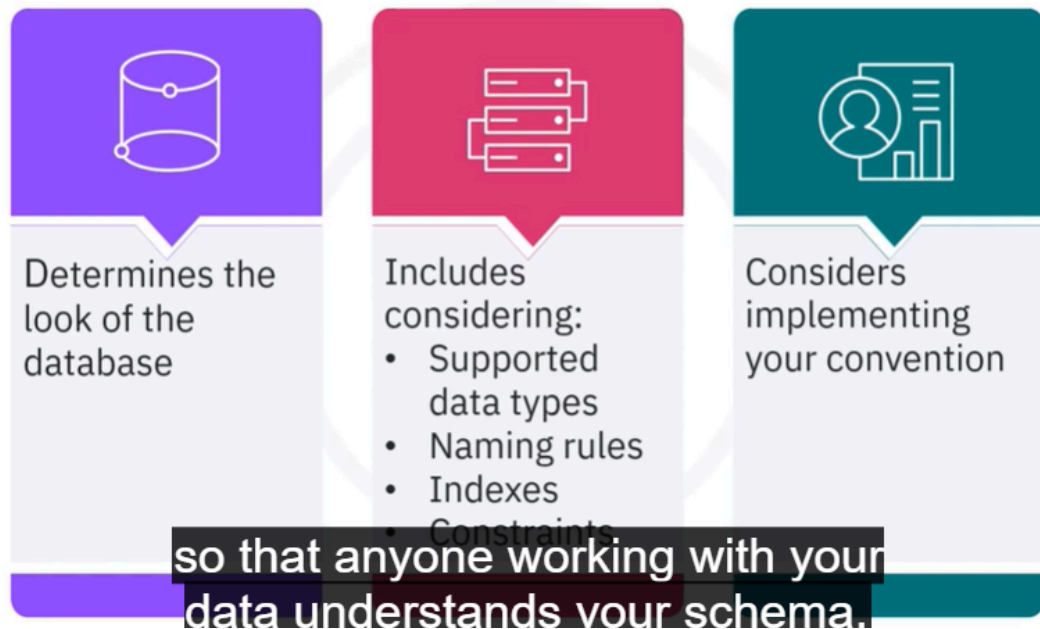


Normalization



- Normalization as a part of the database design process aims to reduce data redundancy. In online transaction processing (OLTP) systems, achieving the third normal form ensures transactional efficiency. Conversely, online analytical processing (OLAP) systems prioritize denormalization for enhanced read performance. Adhering to the first normal form involves eliminating the possibility of multiple author names listed in a single attribute, ensuring a more organized and efficient database structure.
- Additionally, creating a distinct author entity, instead of splitting into authors one and two, which violates the second normal form, is a critical step. Establishing a many-to-one relationship between the authors and book entities is key to ensuring a more flexible and normalized database structure.

Physical design



- When completing normalizing your entities, you can move on to the physical design stage, how your database will look. At this stage, consider the impact of your chosen database management system, including supported data types, naming rules, indexes, and constraints. When considering naming rules, you should also consider implementing your convention so that anyone working with your data understands your schema.

Let's talk about the importance of good database design.

Good database design is like **building a strong foundation for a house**. Just as a sturdy foundation keeps a house safe and stable, **a well-designed database**

ensures that the data is organized, reliable, and easy to access. When you take the time to design your database properly, it helps **prevent problems like data duplication and confusion**, which can lead to costly mistakes later on. Think of it as planning a road map before embarking on a journey; it makes the trip smoother and more efficient.

Now, imagine you are organizing a library. You need to keep track of books, authors, and borrowers. If you don't design your database well, you might end up with a messy system where it's hard to find out who borrowed which book or what books are available. By carefully planning how to categorize and relate this information, you create a clear and effective system that everyone can use easily.

What challenges might arise from poor database design?

Poor database design can lead to several challenges, including:

- **Data Redundancy:** This occurs when the same data is stored in multiple places, leading to inconsistencies. For example, if a book's information is duplicated in different records, updating one record may not update the others.
- **Data Integrity Issues:** Inaccurate or inconsistent data can arise, making it difficult to trust the information stored in the database. For instance, if a borrower's contact details are incorrect, it can lead to communication problems.
- **Performance Problems:** A poorly designed database can slow down data retrieval and processing. If the database structure is not optimized, queries may take longer to execute, affecting user experience.
- **Complexity in Data Management:** Managing and maintaining a poorly designed database can become complicated and time-consuming. It may require more effort to perform updates, backups, and data migrations.
- **Difficulty in Scalability:** As the amount of data grows, a poorly designed database may struggle to handle increased loads, making it challenging to

scale the system effectively.

- **Inability to Meet Business Needs:** If the database does not accurately reflect the relationships and requirements of the business, it may fail to support necessary operations and decision-making processes.

Addressing these challenges often requires significant time and resources, which is why investing in good database design upfront is crucial.