A screenshot of a whiteboard with text and numbers

Description automatically generated

3 alfa olan row olduğundan lossless decomposition

A black text with black letters

Description automatically generated with medium confidence

A diagram of a process

Description automatically generated

A black text on a white background

Description automatically generated

WHAT IS WEAK ENTITY?

* Weak entity is not able to be identified by its attributes alone.
* A weak entity set is one whose existence is dependent on another entity
* otel room is dependent on the existence of the otel. Rooms are part of the otel and depends on primary key of entity set that is otel.

WHAT IS REFERENTIAL INTEGRITY CONSTRAINT? EXPLAIN AND GIVE AN EXAMPLE

* concept that ensures the consistency and accuracy of relationships between tables
* A foreign key in one table refers to the primary key in another table
* ON DELETE CASCADE …

EXPLAIN STEPS IN QUERY PROCESSING

* parsing and translation
  + checks query code is valid (syntax), checks if it makes sense (semantics). After that query is translated into relational algebra.
* optimization
  + plans the execution to be most optimized way.
* evaluation
  + follow optimizers plan and executes the query

WHAT IS DDL, DML?

* DDL: data definition language. specification notation for defining the db schema. For example, creating the tables, giving attributes and types. This generates set of table templates stored in data dictionary.
* DML: data manipulation language. This is the query language that helps making queries. 2 types of dmls: procedural (require a user to specify what data are needed and how to get those data) and declarative (require a user to specify what data are needed without specifying how to get those data).
* DCL: control, grant, revoke

DEFINE 3NF AND BCNF GIVE EXAMPLE BEFORE THEM AND AFTER THEM

* Normalizations prevents inconsistencies.
* 3NF at least one of the following holds (a and B are attributes of R):
  + α → *β* is trivial (i.e., *β* ∈ α)
  + α is a superkey for *R*
  + Each attribute *A* in *β* – α is contained in a candidate key for *R.*
  + *before:*
    - Employees (EmployeeID, EmployeeName, ProjectID, ProjectName, ProjectDescription)
  + after:
    - Employees (EmployeeID, EmployeeName)
    - Projects (ProjectID, ProjectName, ProjectDescription)
    - EmployeeProjects (EmployeeID, ProjectID)
* BCNF
  + If a relation is in bcnf, it is in 3nf
  + at least one holds
    - α → *β* is trivial (i.e., *β* ⊆ α)
    - α is a superkey for *R*
  + *in\_dep* (*ID, name, salary, dept\_name, building, budget* )
    - not in BCNF
    - dept\_name -> building, budget
      * holds on in\_dep but dept\_name is not a superkey
    - decompose it into instructor and department then they are both BCNF

WHAT ARE FUNCTIONAL DEPENDENCIES, WHY WE NEED THEM?

* An instance of a relation that satisfies all such real-world constraints is called a **legal instance** of the relation;
* A functional dependency is a generalization of the notion of a *key*
* functional dependency, indicates that how attributes in a db depend on another attributes. These are necessary to ensure data consistency and protect the data integrity.

WHAT IS DATABASE CLOSURE?

* The set of all functional dependencies logically implied by F is the closure of F.

WHAT IS CHECK CONSTRAINT?

* check constraint is applied to a column of a database. It specifies that the values a column can take must meet a certain condition.

**create table** *section*

(*course\_id* **varchar** (8),

*sec\_id* **varchar** (8),

*semester* **varchar** (6),

*year* **numeric** (4,0),

*building* **varchar** (15),

*room\_number* **varchar** (7),

*time slot id* **varchar** (4),

**primary key** (*course\_id*, *sec\_id*, *semester*, *year*),

**check** (*semester* **in** ('Fall', 'Winter', 'Spring', 'Summer')))

DATA CUBE VS DATABASE

* A **data cube** is a multidimensional generalization of a cross-tab
* Cross-tabs can be used as views on a data cube
* databases are generally used for daily tasks like storing and managing the data
* data cubes are data structure that is optimized to be used for multidimensional data analysis, deciding processes.

her super key candidate key değildir

candidate keye bakarken

super key mi

is any subset of AG a superkey

if so not candidate key

soldakini ayıramazsın

sağdakini açabilirsin tek tek

A white paper with black and white text

Description automatically generated

kesişimleri ya r1’e ya r2’e gidecek

lossless decomposition

soldakilerin hepsi superkey 🡪 bcnf

3nf

soldakilerin hepsi superkey

ya da

sağdakiler candidate key’de var mı

soldaki nereye gideceğini bilmeli, gideceği nokta tek olmalı

A white background with black text

Description automatically generated

2.madde reelden betayı çıkarmak direkt

trivialları geç

sağdaki soldakinin alt kümesi

A black screen with white text

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

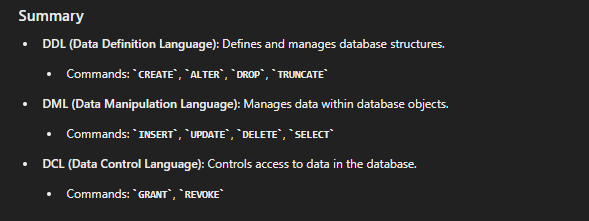
A screenshot of a computer

Description automatically generated

A black screen with white text

Description automatically generated

DDL



Dense index — Index record appears for every search-key value in the file.

Sparse Index: contains index records for only some search-key values.

Clustering index: in a sequentially ordered file, the index whose search key specifies the sequential order of the file.

Secondary index: an index whose search key specifies an order different from the sequential order of the file.

Multilevel index: treat index kept on disk as a sequential file and construct a sparse index on it.