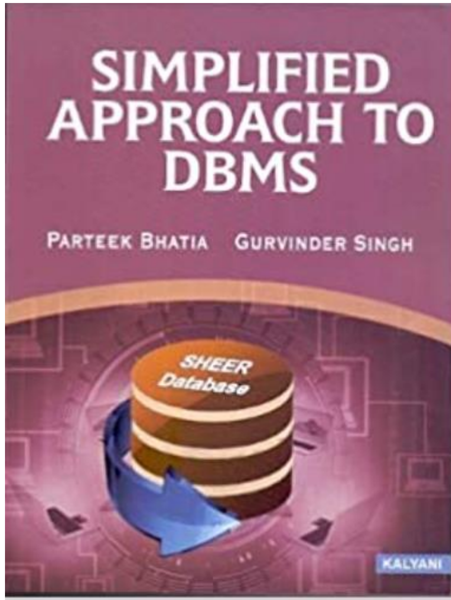
Chapter: 1
Fundamentals of DBMS

By
Dr. Parteek Bhatia
Associate Professor
Department of Computer Science & Engineering
Thapar Institute of Engineering and Technology
Patiala

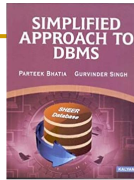


D Data

B Base

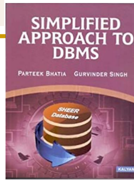
M Management

S System



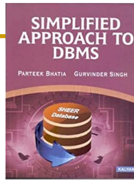
Understanding Data and Information

Data	Information
Data is raw facts and figures	Processed form of data
23 is data	Age 23 is information
Data cannot be used for decision making	Information is useful for decision making
Data is atomic.	Information is collection of data.



Database

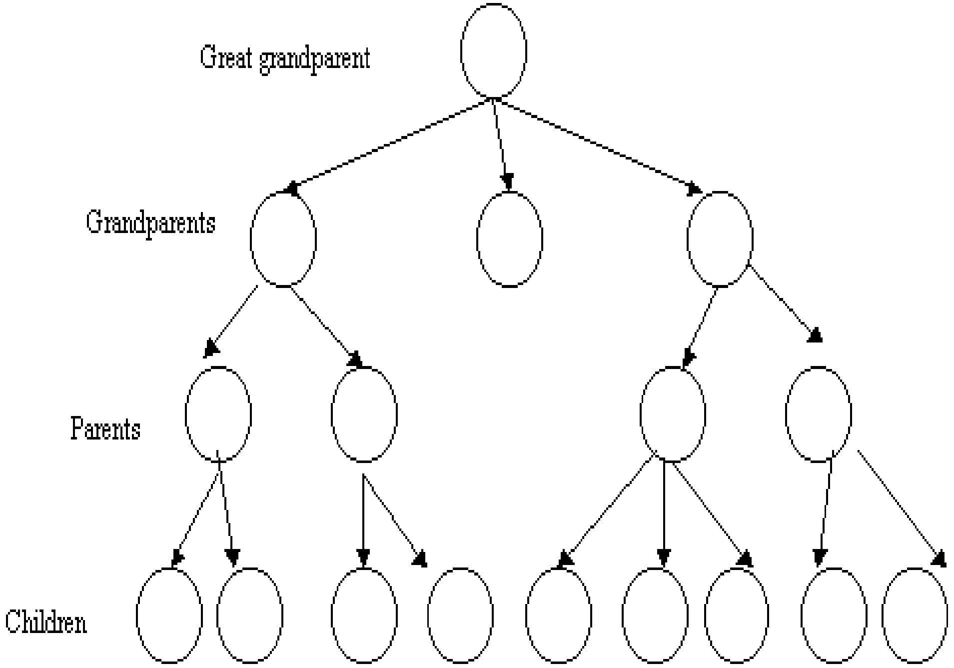
The related information when placed is an organized form makes a database.



Ways to Organize the Information

Conversion of Data to Information

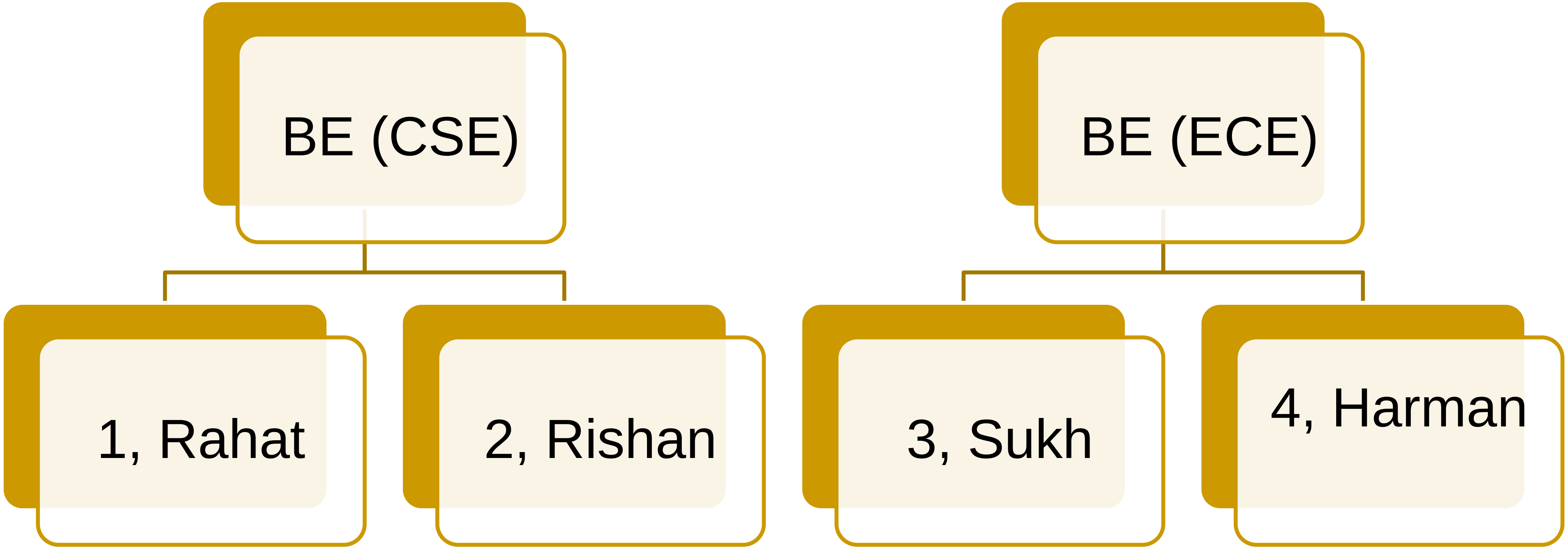
- Tree Data Structure

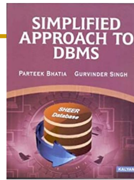


Ways to Organize the Information

Conversion of Data to Information

- Tree Data Structure

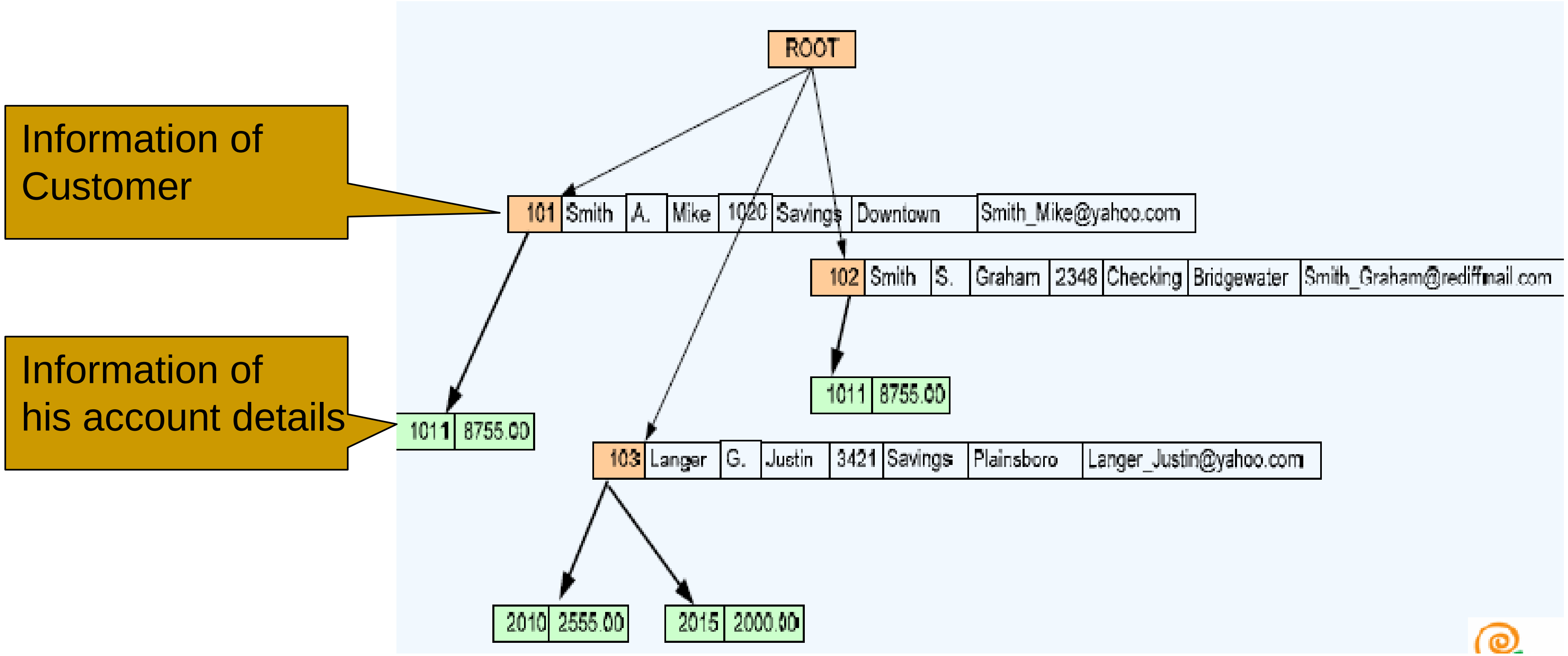


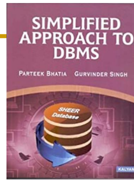


Ways to Organize the Information

Conversion of Data to Information

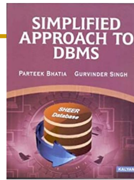
- Tree Data Structure: Information of Customers and their Account details





Ways to Convert Data to Information

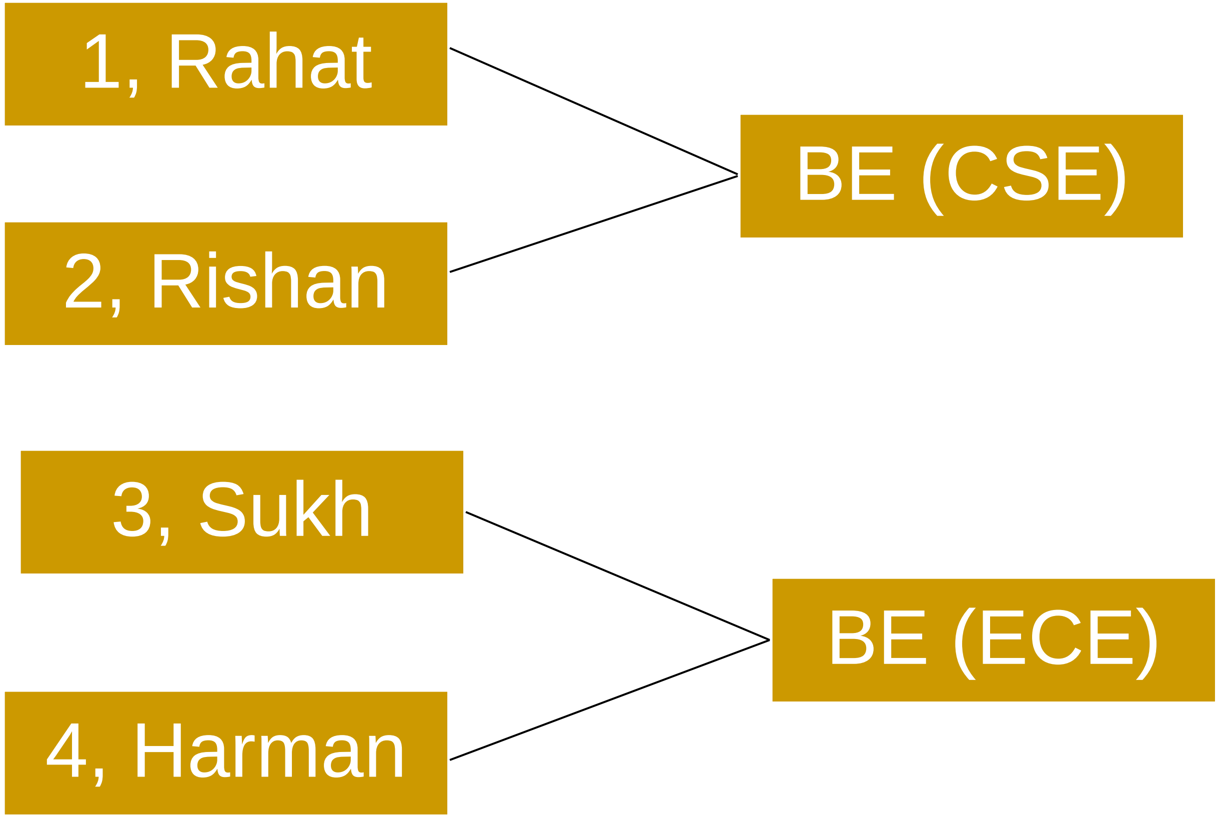
- Data Structure
  - Tree
- Data Model
  - Hierarchical Model

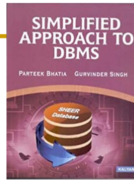


Ways to Organize the Information

Conversion of Data to Information

- Graph Data Structure

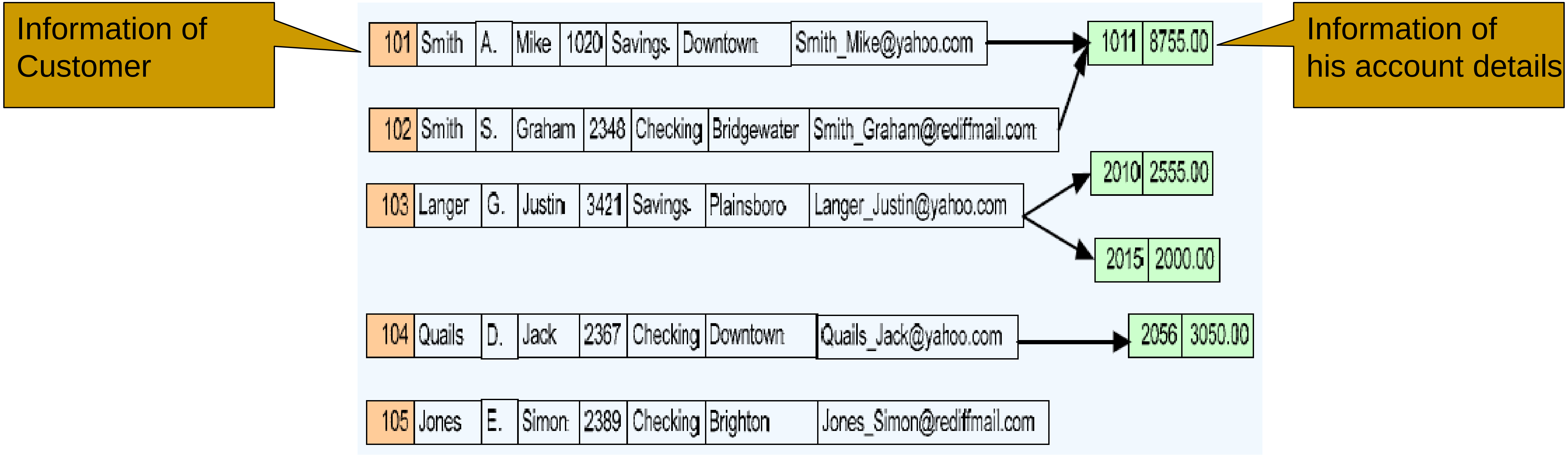


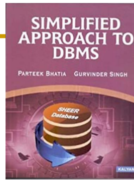


Ways to Organize the Information

Conversion of Data to Information

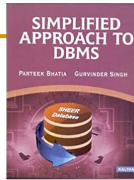
- Graph Data Structure: Information of Customers and their Account details





Ways to Convert Data to Information

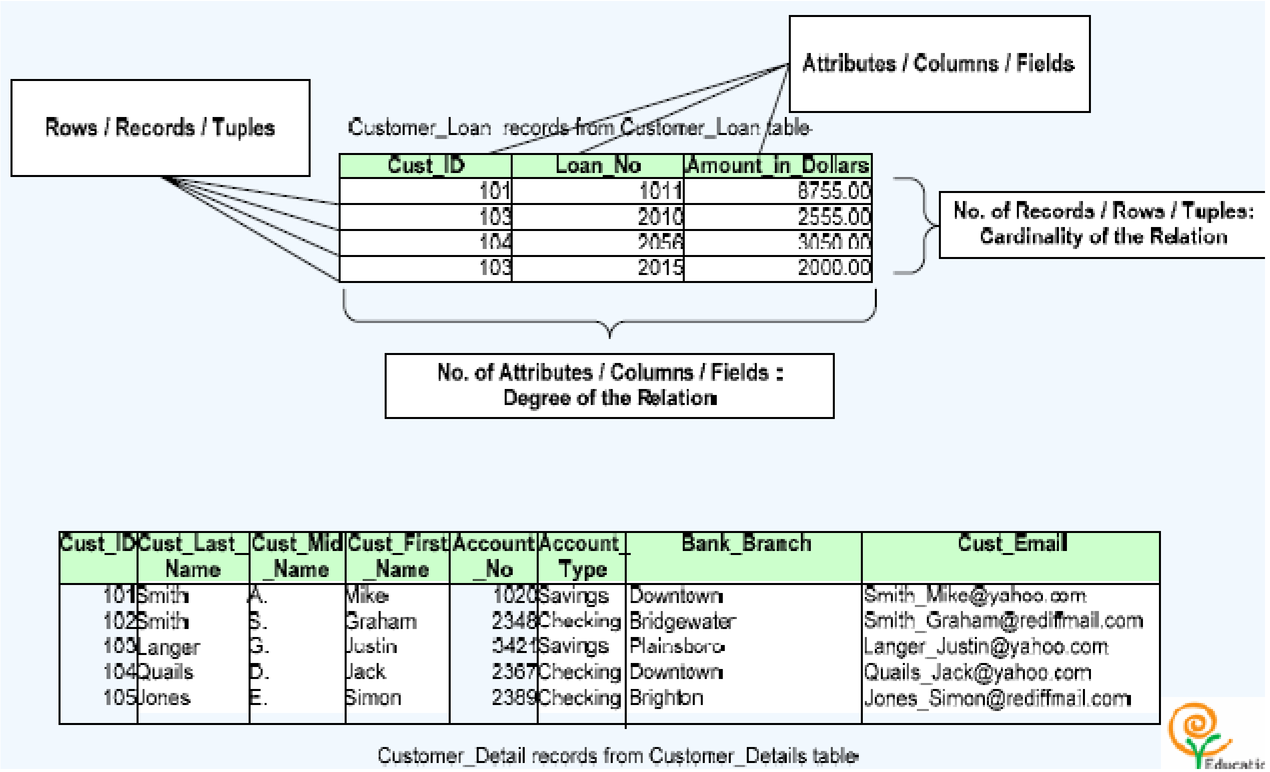
- Data Structure
  - Graph
- Model
  - Network Model



Ways to Organize the Information

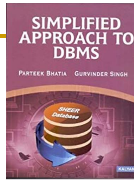
Conversion of Data to Information

- Arrays Data Structure: Information of Customers and their Account details



Cust_ID	Loan_No	Amount_in_Dollars
101	1011	8755.00
103	2010	2555.00
104	2056	3050.00
103	2015	2000.00

Cust_ID	Cust_Last_Name	Cust_Mid_Name	Cust_First_Name	Account_No	Account_Type	Bank_Branch	Cust_Email
101	Smith	A.	Mike	1020	Savings	Downtown	Smith_Mike@yahoo.com
102	Smith	S.	Graham	2348	Checking	Bridgewater	Smith_Graham@rediffmail.com
103	Langer	G.	Justin	3421	Savings	Plainsboro	Langer_Justin@yahoo.com
104	Quails	D.	Jack	2367	Checking	Downtown	Quails_Jack@yahoo.com
105	Jones	E.	Simon	2389	Checking	Brighton	Jones_Simon@rediffmail.com

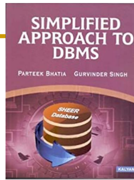


Ways to Organize the Information

Conversion of Data to Information

- Arrays Data Structure

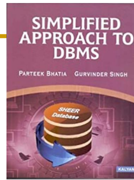
Rno	Name	Class
1	Rahat	BE(CSE)
2	Rishan	BE(CSE)
3	Sukh	BE(ECE)
4	Harman	BE(ECE)



Ways to Organize the Information

Conversion of Data to Information

- Data Structure
  - Array/Table
- Model
  - Relational Model



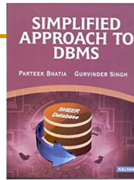
Ways to Organize the Information

Conversion of Data to Information

- Three Data Models
  - Hierarchical Data Model: Based on tree data structure
  - Network Data Model: Based on graph data structure
  - Relational Data Model: Based on Arrays/tables

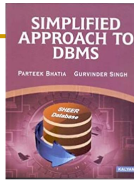
Relational Data Model is simple and easy to use.

Thus, out of all data models Relational Data model is commonly used. And popular software like Oracle, MySQL, MS Access are based on this model.



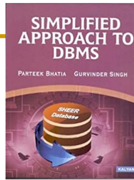
DBMS: Management

- It is management of information.
- We can perform Insert, Update, Delete and Retrieve operations over database.



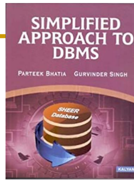
Operations on Databases

- To add new information (e.g. to add the address of a new friend in your address book)
- To view or retrieve the stored information (e.g. you have to find the address of one of your old friends)
- To modify or edit the existing information (e.g. your friend has shifted to a new place so his address would get changed)
- To remove or delete the unwanted information (e.g. your friend has changed his/her mobile number, so his/her mobile number would have to be removed from list)
- Arranging the information in a desired order etc.



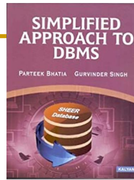
DBMS: System

- A system or software which manage the database on a computer.



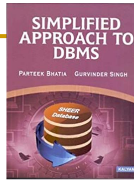
DBMS

- A software responsible to manage database, i.e., a software responsible to manage insert, update, delete and retrieve operations over database in a computer is known as Database Management System or DBMS.
- Examples: Oracle, MySQL, Sybase, SQL Server, FoxPro etc.



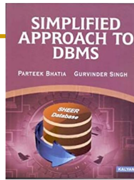
Maintaining Database on Computers

- There are two approaches
  - Traditional File Based Approach
  - Centralized DBMS approach

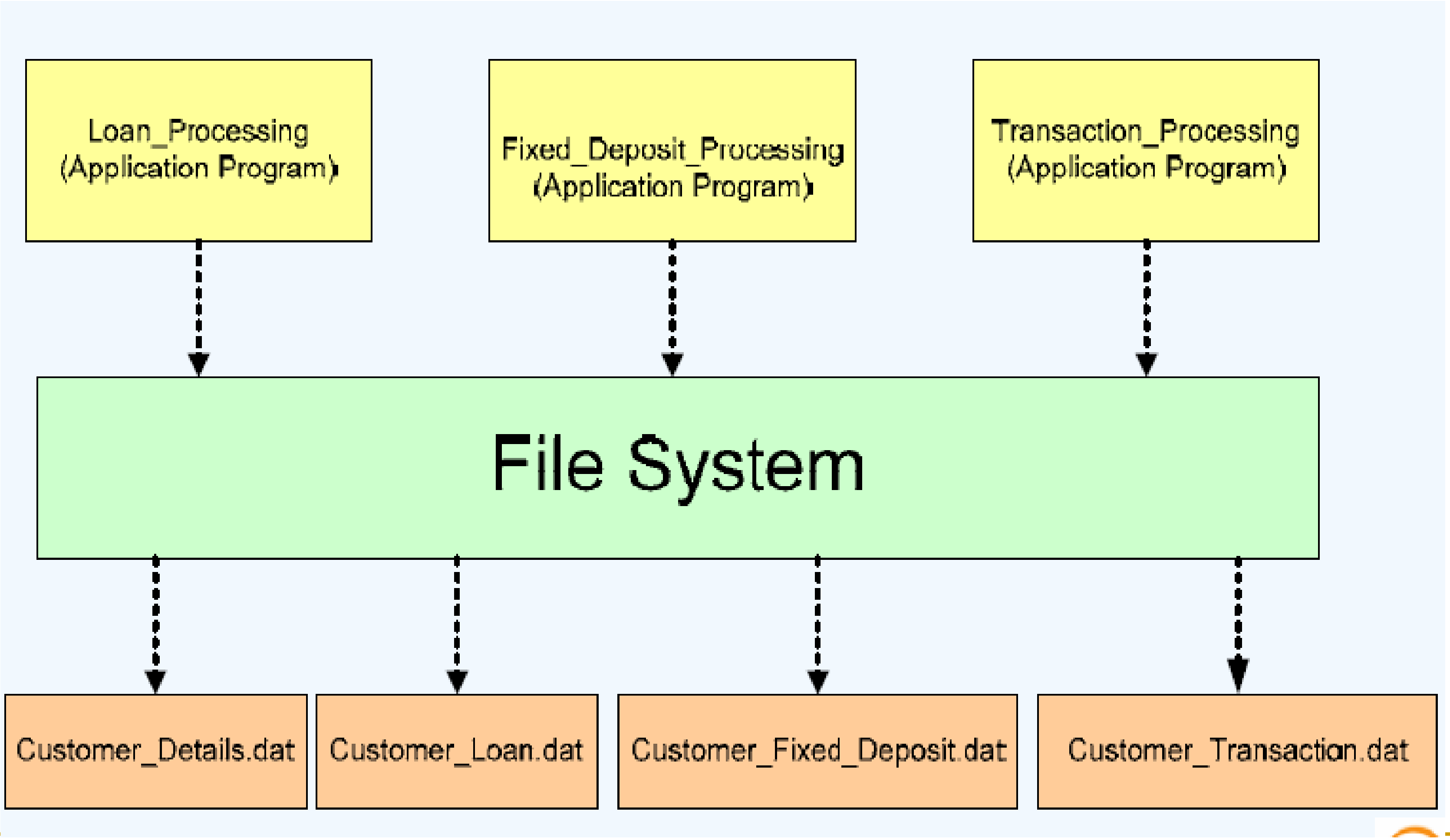


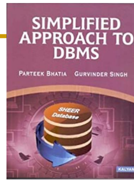
File Based Approach

- It is an early attempt to computerize the manual filing system. It is first implemented by using file handling utilities of COBOL (Common Business Oriented Language).
- Data is stored in flat files. For example: Text files, csv files
- Each file called a flat file, contained and processed information for one specific function.
- Now a days file Handling utilities of Programming languages like C, C++, Java etc. is used to implement it.



File Based Approach





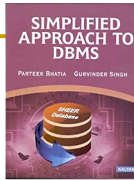
File Based Approach

Ways of storing data in files – customer data

4176	Aniruddha Sarkar	SBU1
4181	Manoj Saha	SBU1
4183	Moushumi Dharchoudhury	SBU1
4203	Suryanarayana D.V.S.S.	SBU1
4204	Vivek Rai	SBU1

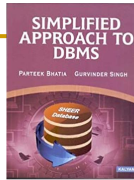
Predefined

4176 AniruddhaSarkar SBU1
4181 ManojSaha SBU1
4183 MoushumiDharchoudhury SBU1
4203 SuryanarayanaD.V.S. SBU1
4204 Vivek Rai SBU1



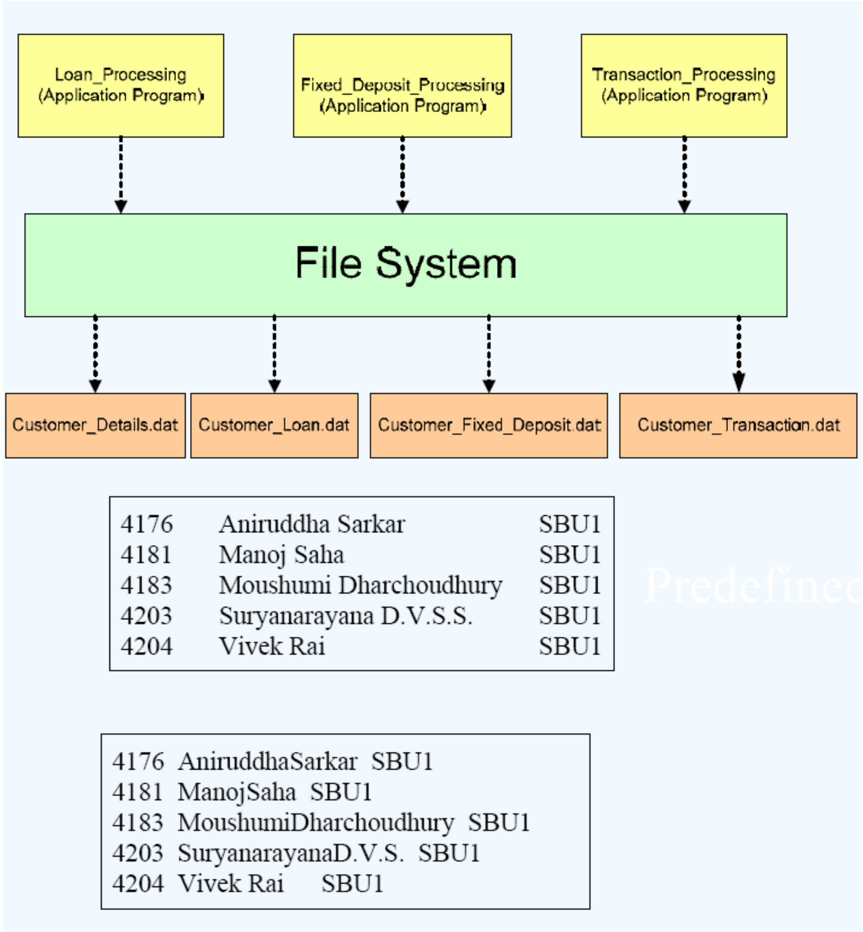
Limitations of the File-Based Approach

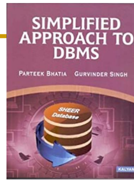
- **Separated and Isolated Data**
- **Duplication of data**
- **Data Dependence**
- **Difficulty in representing data from the user’s view**
- **Data Inflexibility**
- **Incompatible file formats**



Limitations of the File-Based Approach

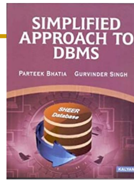
- **Separated and Isolated Data**
- **Duplication of data**
- **Data Dependence**
- **Difficulty in representing data from the user's view**
- **Data Inflexibility**
- **Incompatible file formats**



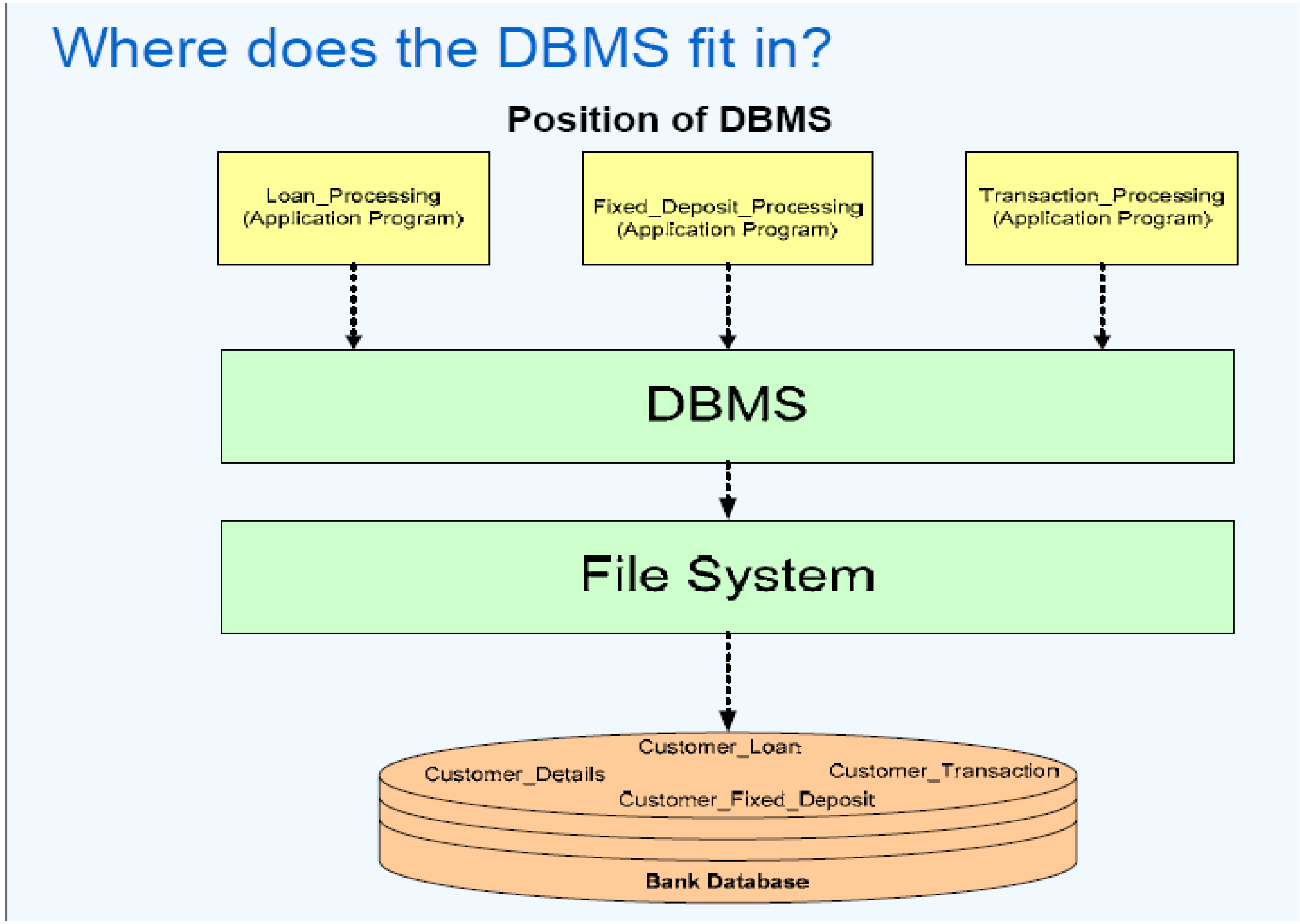


Database Approach

- The database is a single, large repository of data, which can be used simultaneously by many departments and users.

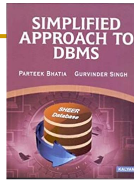


Database Approach



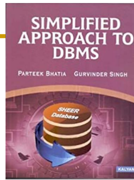
Example: File Based System

General Office	Library	Hostel	Account Office
Rollno	Rollno	Rollno	Rollno
Name	Name	Name	Name
Class	Class	Class	Class
Father_Name	Address	Father_Name	Address
Date_of_birth	Date_of_birth	Date_of_birth	Phone_No
Address	Phone_No	Address	Fee
Phone_No	No_of_books_issued	Phone_No	Installments
Previous_Record	Fine	Mess_Bill	Discount
Attendance	etc.	RoomNo	Balance
Marks		etc.	Total
etc.			etc.



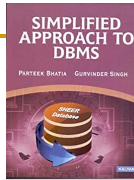
Example: Database Approach

General Office	Library	Hostel	Account Office
Rollno	Rollno	Rollno	Rollno
Name	No_of_books_issued	RoomNo	Fee
Class	Fine	Mess_Bill	Installments
Father_Name	etc.	etc.	Discount
Address			Balance
Phone_No			Total
Date_of_birth			etc.
Previous_Record			
Attendance			
Marks			
etc.			

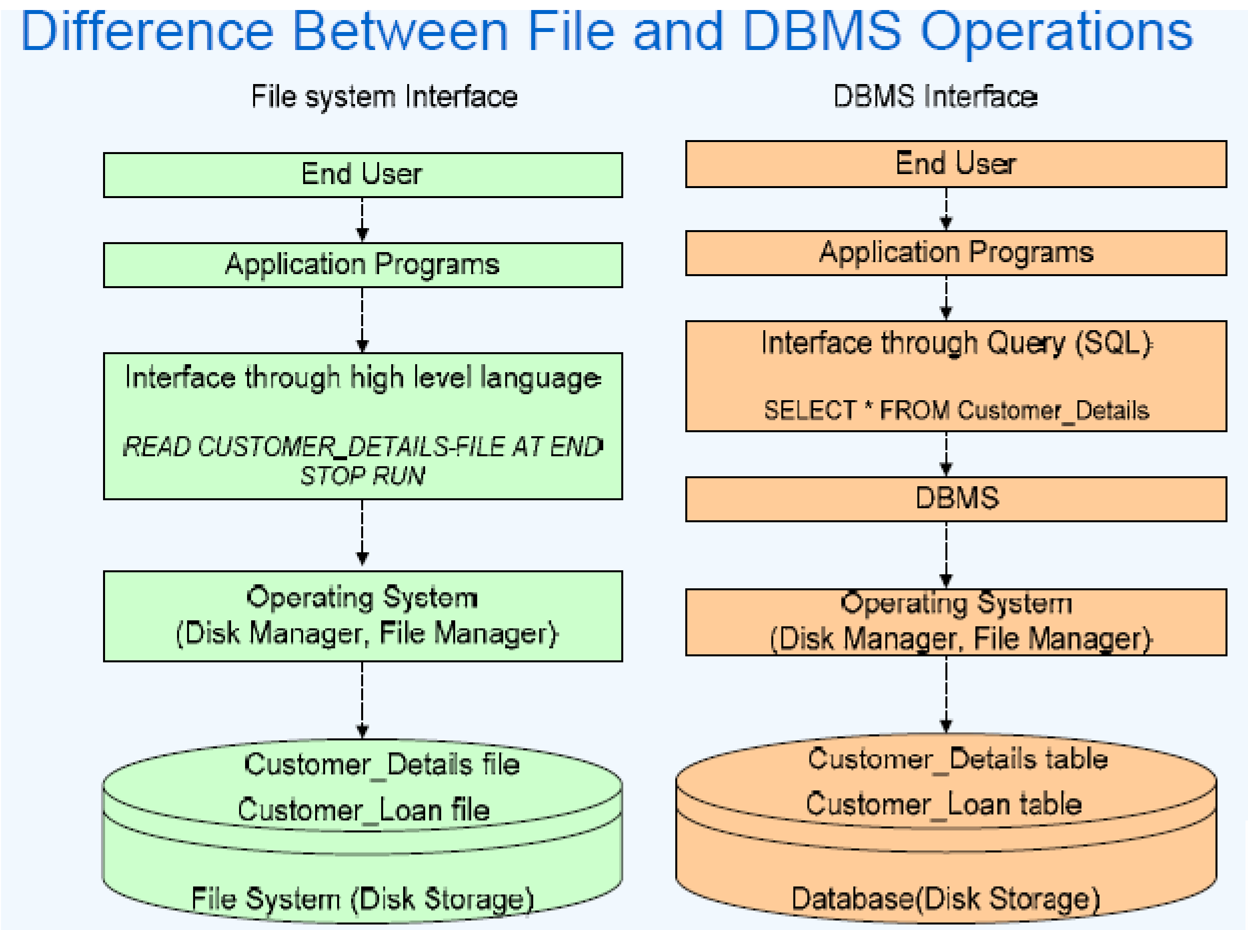


Comparison of File Management System with Database Management System

File Management e.g. C++ or COBOL program	Database Management e.g. Oracle or Sybase
Small systems	Large systems
Relatively cheap	Relatively expensive
Few 'files	Many 'files'
Files are files	Files are tables
Simple structure	Complex structure
Redundant data	Reduced redundancy
Chances of inconsistency	Consistent
Isolated data	Data can be shared
Little preliminary design	Vast preliminary design
Integrity left to application programmer	Rigorous inbuilt integrity checking
No security	Rigorous security
Simple, primitive backup/recovery	Complex & sophisticated backup/recovery
Often single user	Multiple users

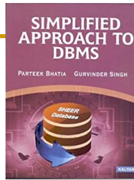


Comparison of File Management System with Database Management System



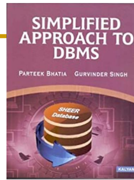
Advantages of DBMS

- **Controlling Redundancy**



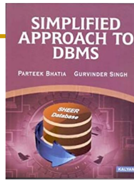
Integrity can be enforced

- Integrity of data means that data in database is always accurate, such that incorrect information cannot be stored in database.



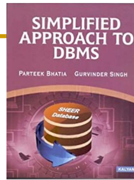
Inconsistency can be avoided

- Data of same entity should be same at all the places.
- Duplication of data may results in to inconsistency as the two entries regarding the same data may not agree. At such times the data is said to be inconsistent.



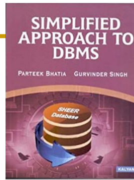
Other Advantages

- Data can be shared
- Providing Backup and Recovery
- Standards can be enforced
- Restricting unauthorized access
- Solving enterprise requirement than individual requirement



Disadvantages of DBMS

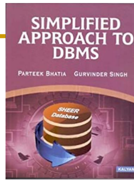
- Complexity
- Size
- Performance
- Higher impact of a failure
- Cost of DBMS
- Additional Hardware costs
- Cost of Conversion



When not to Use a DBMS

The overhead costs of using a DBMS are due to the following:

- High initial investment in hardware, software, and training
- Overhead for providing security, concurrency control, recovery, and integrity functions.
- Additional problems may arise if the database designers and DBA do not properly design the database or if the database systems applications are not implemented properly.



Components of DBMS

- Hardware
- Software
- Data
- Users
- Procedures

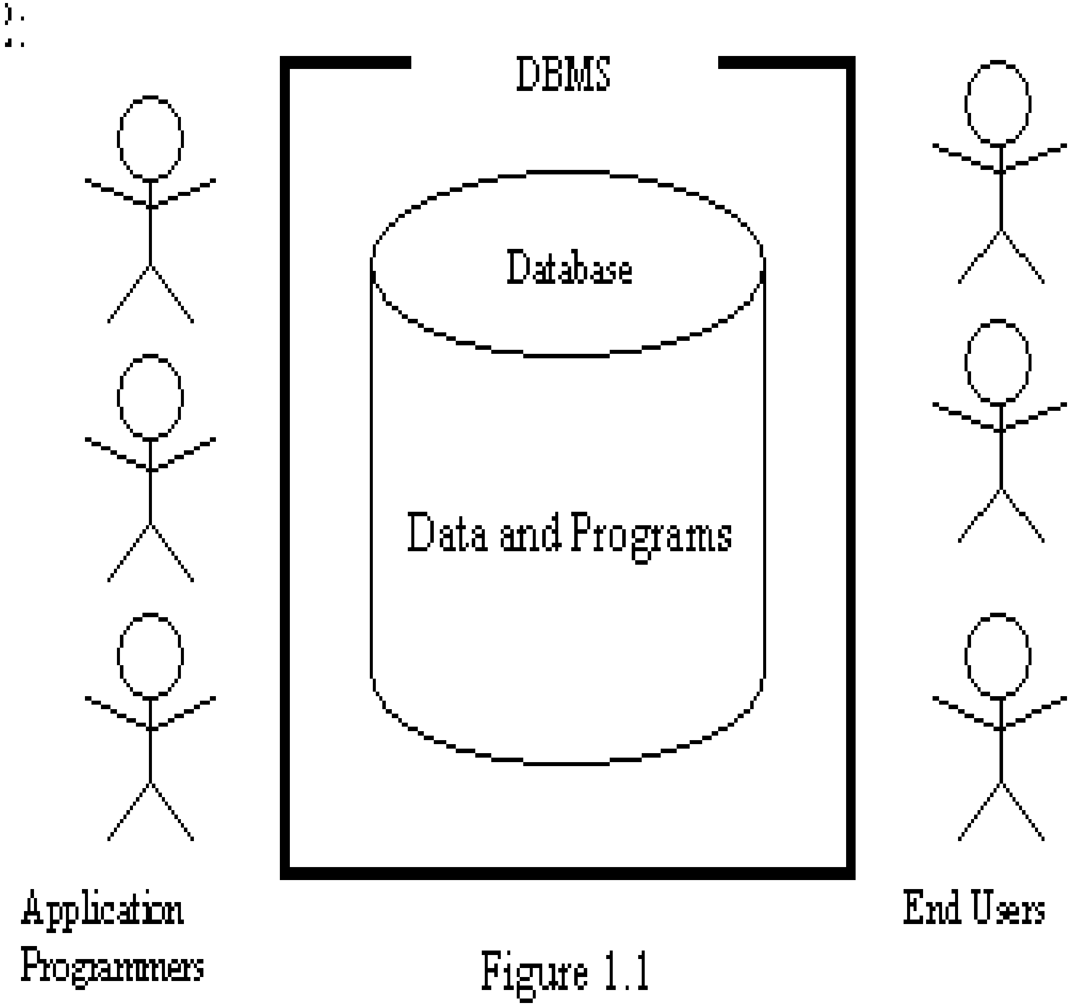
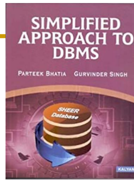
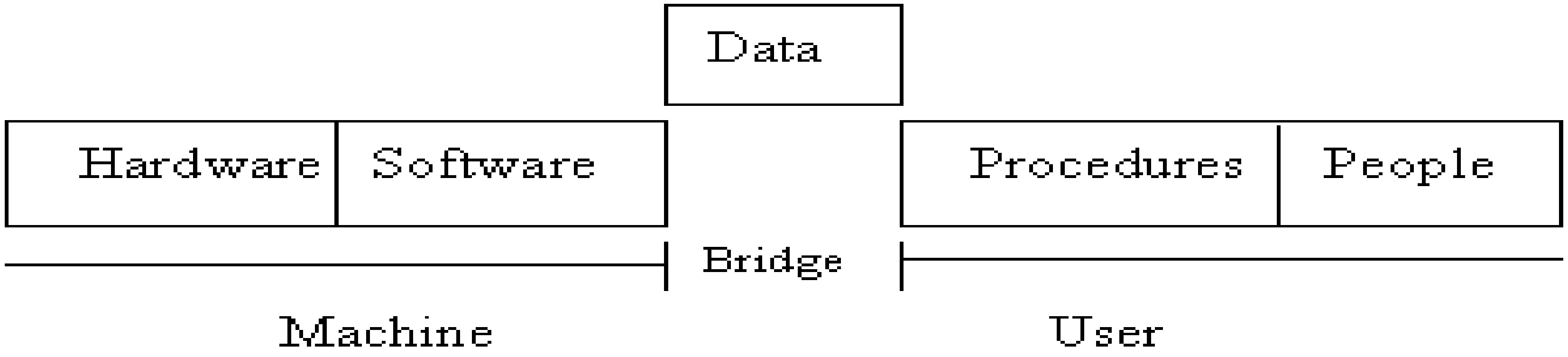


Figure 1.1

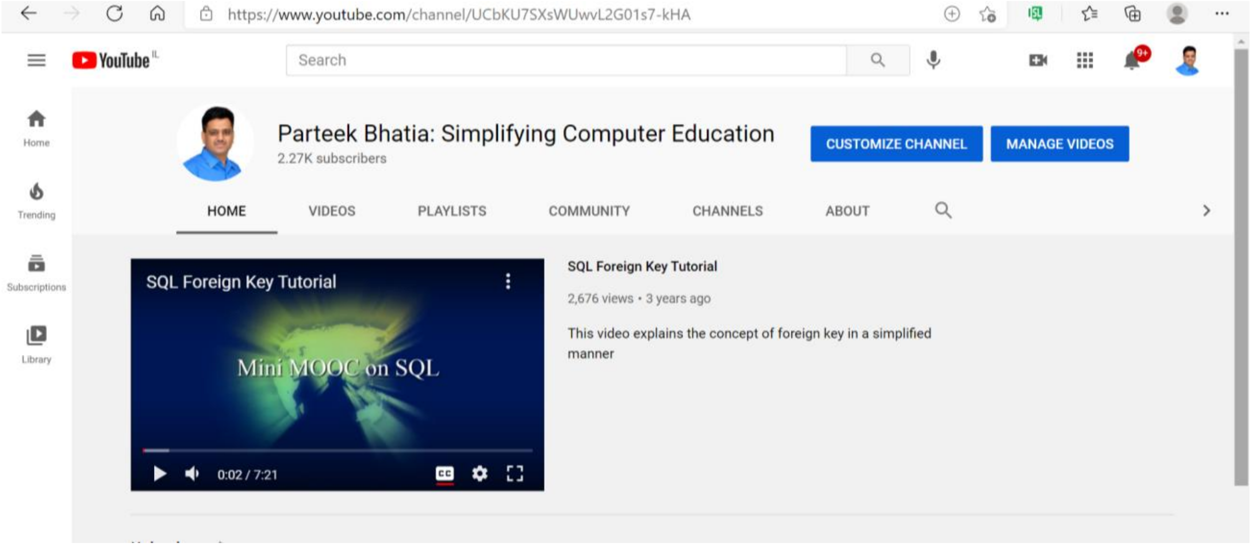


Components of DBMS



For more information

- Subscribe to YouTube Channel from the Author
  - To receive latest video tutorials on Data Mining, Machine Learning, DBMS, Big Data, NoSQL and many more.
- https://www.youtube.com/user/parteekbhatia



amazon simplified approach to d ×
Amazon.in: Buy Simplified Appro ×

https://www.amazon.in/Simplified-Approach-Gurvinder-Bhatia-Parteek/dp/B07WCJJ18B

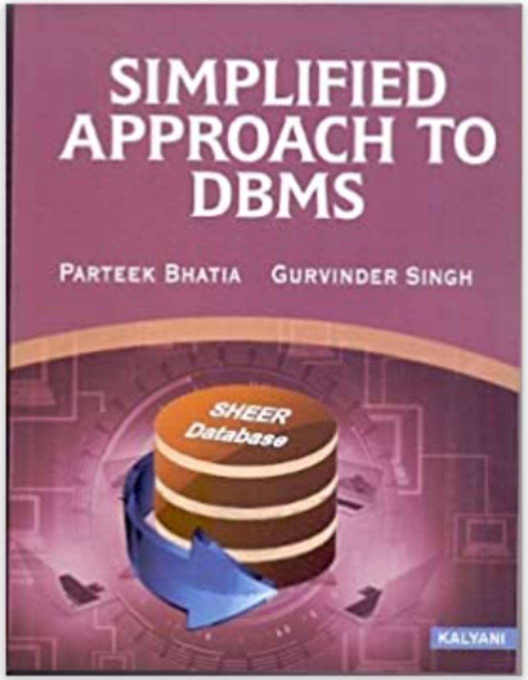
amazon.in prime
Deliver to Parteek
Patiala 147004
Books
Hello, Parteek
Account & Lists
Returns
& Orders
5
Cart

All | Prime | Parteek's Amazon.in | Amazon Pay | Mobiles | Best Sellers | Today's Deals

MASTER WATCH NOW prime video
*Redirects to PrimeVideo.com

Books | Advanced Search | New Releases & Pre-orders | Best Sellers | Browse Genres | Children's & Young Adult | Textbooks | Exam Central | All Indian Languages

Subtotal
₹ 3,695.00



See this image

Simplified Approach to DBMS Unbound – 1 January 2019

by Gurvinder Singh Bhatia Parteek (Author)

2 ratings

› See all formats and editions

Unbound
₹ 400.00

3 New from ₹ 400.00

Delivery By: **Feb 8 - 9** Details

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

₹ 400.00
M.R.P.: ~~₹ 450.00~~
You Save: ₹ 50.00 (11%)
Inclusive of all taxes

FREE Delivery. Details

In stock.
Sold by Lyall Book Depot..
Quantity: 1

Add to Cart
Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

Share

Type here to search
ENG
00:14
04-02-2021

Some of online Video Courses created by Author over Udemy Platform

Simplified Approach to DBMS | U

udemy.com/course/simplified-approach-to-dbms/

Paused

Apps | Life+Style (Chd), LS... | InformIT: Safari Boo... | ICEET 2014 | Fortune Hotels - M... | Book Lowest Price I... | Captive Portal | BELIEVE in, INVEST i... | Machine Learning f...

Anytime, anywhere | Learn on your schedule from any device

Udemy | Categories | Search for anything | Udemy for Business | Teach on Udemy | Log in | Sign up

Development > Database Design & Development > Database Management

Simplified Approach to DBMS

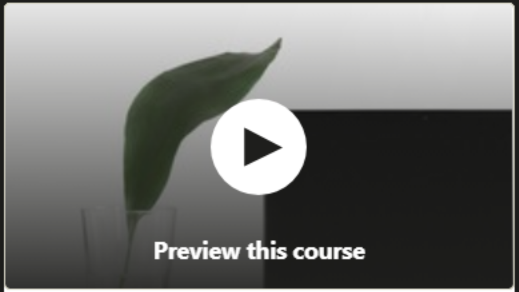
Simplified way to learn DBMS

4.3 ★★★★☆ (2 ratings) 14 students

Created by Parteek Bhatia

Last updated 11/2020 English

Wishlist | Share | Gift this course



Personal | Teams

₪79.90

Add to cart

Buy now

30-Day Money-Back Guarantee

This course includes:

- 9.5 hours on-demand video
- Full lifetime access

What you'll learn

- Understanding the meaning of term DBMS
- File Based System Vs. Data Base Management System
- Hierarchical, Network and Relational Data Models
- Concept of different types of keys like Super Key, Candidate Key, Primary Key and Foreign Key
- Designing of Entity Relation Model
- Conversion of ER Model to Tables
- SQL: Creation of Tables
- SQL: Insertion, Updation, Deletion and Retrieval of Data

19min

Type here to search

ENG 00:25 04-02-2021

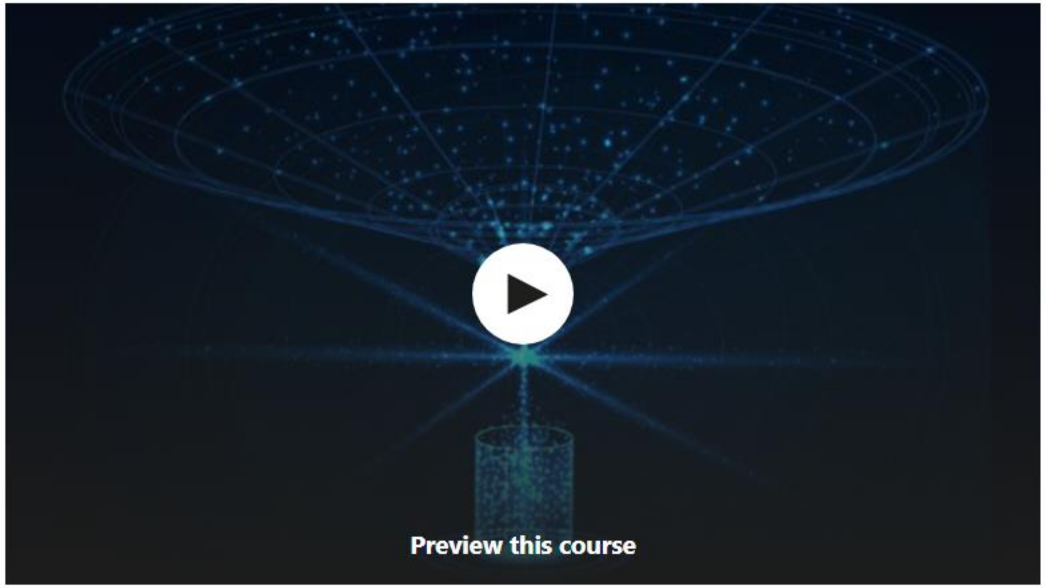
Online Course on SQL at Udemy

https://www.udemy.com/course/learn-sql-in-a-simplified-manner/

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Instructor My learning

Development > Programming Languages > SQL



Learn SQL in a simplified manner

Learn SQL in just four weeks, right from scratch !

4.1 ★★★★☆ (467 ratings) 12,875 students 1hr 56min of on-demand video

Created by Parteek Bhatia

English English [Auto]

Free

Enroll now

What you'll learn Course content Reviews Instructors

✓ SQL is one of the most in demand tech skills today, and learning SQL would surely be one of the fastest and most efficient way to improve ones career prospectus. After the completion of course, one will be able to read, write and comprehend simple to most complex queries of SQL, using which they could work upon any SQL database such as Oracle, Amazon Redshift and others.

Type here to search ENG 00:19 04-02-2021

Mastering PL/SQL | Udemy

udemy.com/course/mastering-plsql/

Apps | Life+Style (Chd), LS... | InformIT: Safari Boo... | ICEET 2014 | Fortune Hotels - M... | Book Lowest Price I... | Captive Portal | BELIEVE in, INVEST i... | Machine Learning f...

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Teach on Udemy Log in Sign up

Teaching & Academics > Online Education > Pl/SQL

Mastering PL/SQL

Simplified way to learn PL/SQL

Highest Rated 4.7 ★★★★☆ (17 ratings) 110 students

Created by Parteek Bhatia

Last updated 5/2020 English English [Auto]

Wishlist Share Gift this course

Personal Teams

₪79.90

Add to cart

Buy now

30-Day Money-Back Guarantee

This course includes:

9.5 hours on-demand video

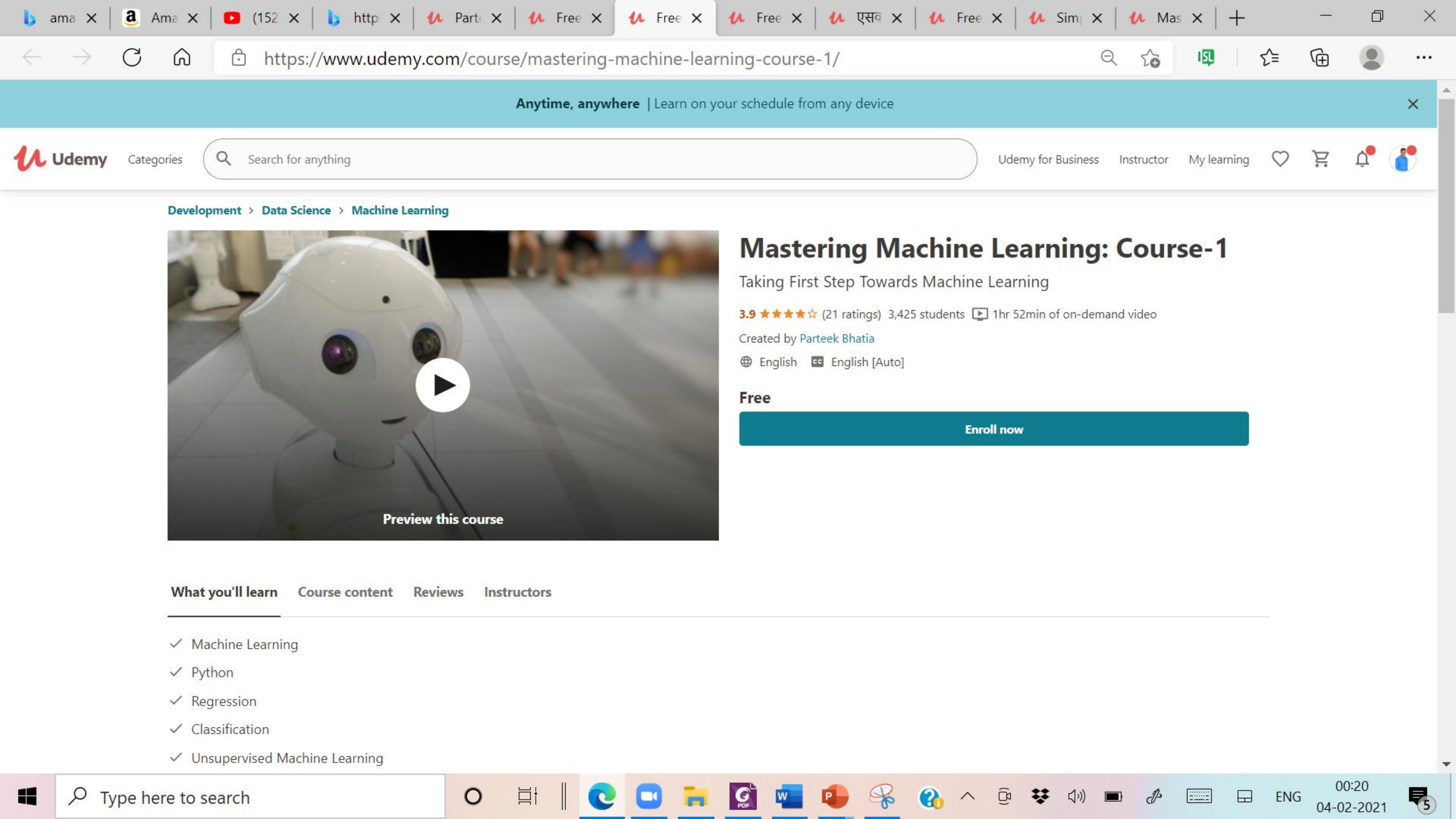
What you'll learn

- ✓ Basics of PL/SQL
- ✓ Variables and Constants
- ✓ Conditional, Looping and Sequential Statements
- ✓ Exception Handling
- ✓ Cursor Handling
- ✓ Sub-Programs, Functions and Procedures
- ✓ Triggers
- ✓ DBMS:: Get chance to enroll in Simplified Approach to DBMS Course completely free

Waiting for www.udemy.com...

Type here to search

ENG 00:23 04-02-2021


https://www.udemy.com/course/mastering-machine-learning-course-1/
Anytime, anywhere | Learn on your schedule from any device
Udemy
Categories
Search for anything
Udemy for Business
Instructor
My learning
Development > Data Science > Machine Learning
Preview this course
Mastering Machine Learning: Course-1
Taking First Step Towards Machine Learning
3.9 (21 ratings) 3,425 students 1hr 52min of on-demand video
Created by Parteek Bhatia
English
English [Auto]
Free
Enroll now
What you'll learn
Course content
Reviews
Instructors
Machine Learning
Python
Regression
Classification
Unsupervised Machine Learning
Type here to search
ENG
00:20
04-02-2021

https://www.udemy.com/course/genetic-algorithm-for-machine-learning/

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Instructor My learning

IT & Software > Other IT & Software > Genetic Algorithm

Genetic Algorithm for Machine Learning

Simplified Way to Learn

New 0.0 (0 ratings) 447 students 1hr 32min of on-demand video

Created by Parteek Bhatia

English

Free

Enroll now

What you'll learn Course content Reviews Instructors

- ✓ Working Principle of Genetics Algorithms
- ✓ Natural Selection
- ✓ Implementation of Natural Selection through Roulette Wheel
- ✓ Crossover or Recombination
- ✓ Concept of Probability of Crossover and Its usage in generation of Population

Touchpad

Type here to search ENG 00:20 04-02-2021

IT & Software > Other IT & Software > Machine Learning

Association Mining for Machine Learning

Simplified Way to Learn

New 4.6 ★★★★★ (13 ratings) 629 students 1hr 48min of on-demand video

Created by Parteek Bhatia

English English [Auto]

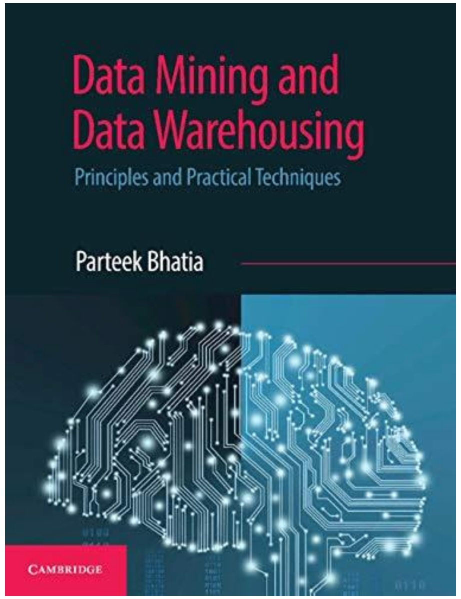
Free

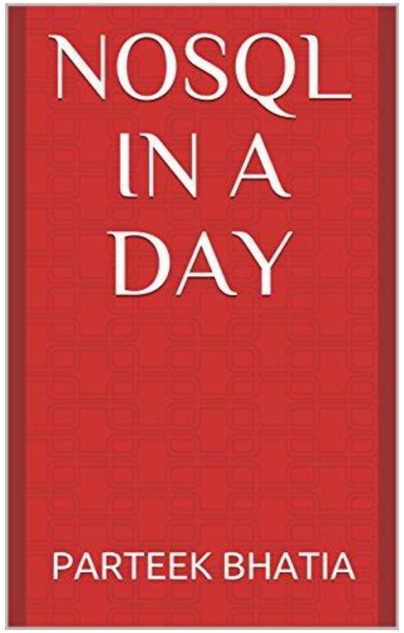
Enroll now

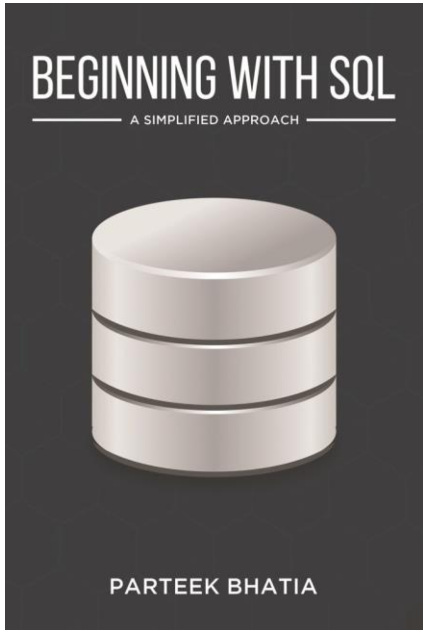
What you'll learn Course content Reviews Instructors

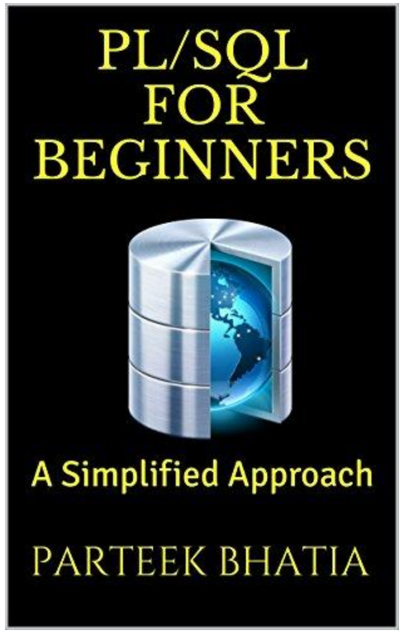
- ✓ Association Mining
- ✓ Naive Algorithm
- ✓ Apriori Algoorithm

Books from the Same Author









For more information visit: www.parteekbhatia.com

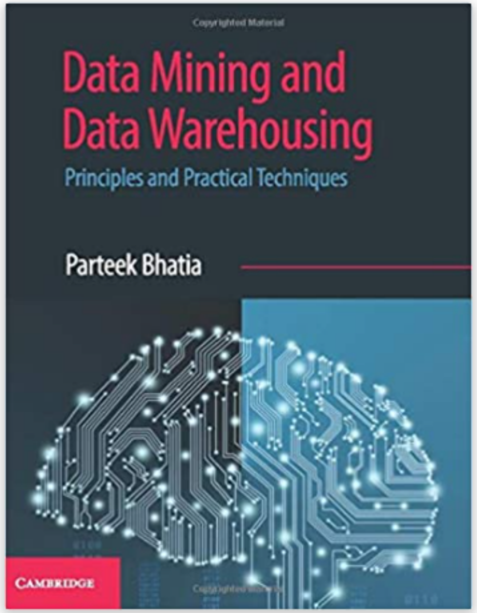
Coming Soon…

- **Text Book**
- **Machine Learning: Principles and Practical Techniques**

https://www.amazon.in/Data-Mining-Warehousing-Principles-Techniques/dp/1108727743/ref=sr_1_...

Data Mining and Data Warehousing and thousands of other textbooks are available for instant download on your Kindle Fire tablet or on the free Kindle apps for iPad, Android tablets, PC or Mac. Learn more

‹ Back to results



See all 2 images

Data Mining and Data Warehousing: Principles and Practical Techniques

Paperback – 27 June 2019

by Parteek Bhatia (Author)

★★★★★ 15 ratings

› See all formats and editions

Kindle Edition	Paperback
₹ 1,374.45	₹ 519.00 prime
Read with Our Free App	14 New from ₹ 499.00

FREE delivery: **Monday, Feb 8**
Order within 13 hrs and 59 mins Details

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

₹ 519.00
M.R.P.: ~~₹ 695.00~~
You Save: ₹ 176.00 (25%)
Inclusive of all taxes
prime

In stock.
Sold by Cloudtail India and Fulfilled by Amazon.
Quantity: 1

Add to Cart
Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

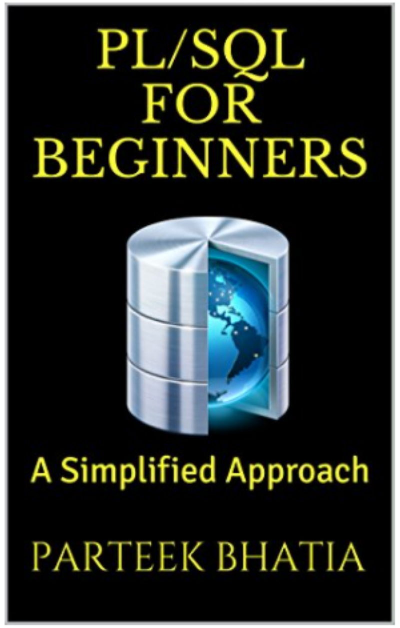
Share

Subtotal
₹ 3,695.00

‹ Back to results

You purchased this item on 4 August, 2016.
View this order | You can find this title on your Amazon Kindle apps and devices, or the Manage Your Content and Devices page. Need additional help? Click here

Look inside



PL/SQL for Beginners: A Simplified Approach Kindle Edition

by Parteek Bhatia (Author) | Format: Kindle Edition

5 ratings

› See all formats and editions

Kindle Edition
₹ 0.00 kindleunlimited

This title and over 1 million more available with Kindle Unlimited
₹ 99.00 to buy

PL/SQL: A Simplified Approach
Why read this book?
This book covers the basics of PL/SQL with lots of real world examples.
This book will help to gain better skills, better placement and better future.
...plified manner.

Kindle Price: ₹ 99.00
inclusive of all taxes

includes free wireless delivery via ***Amazon Whispernet***

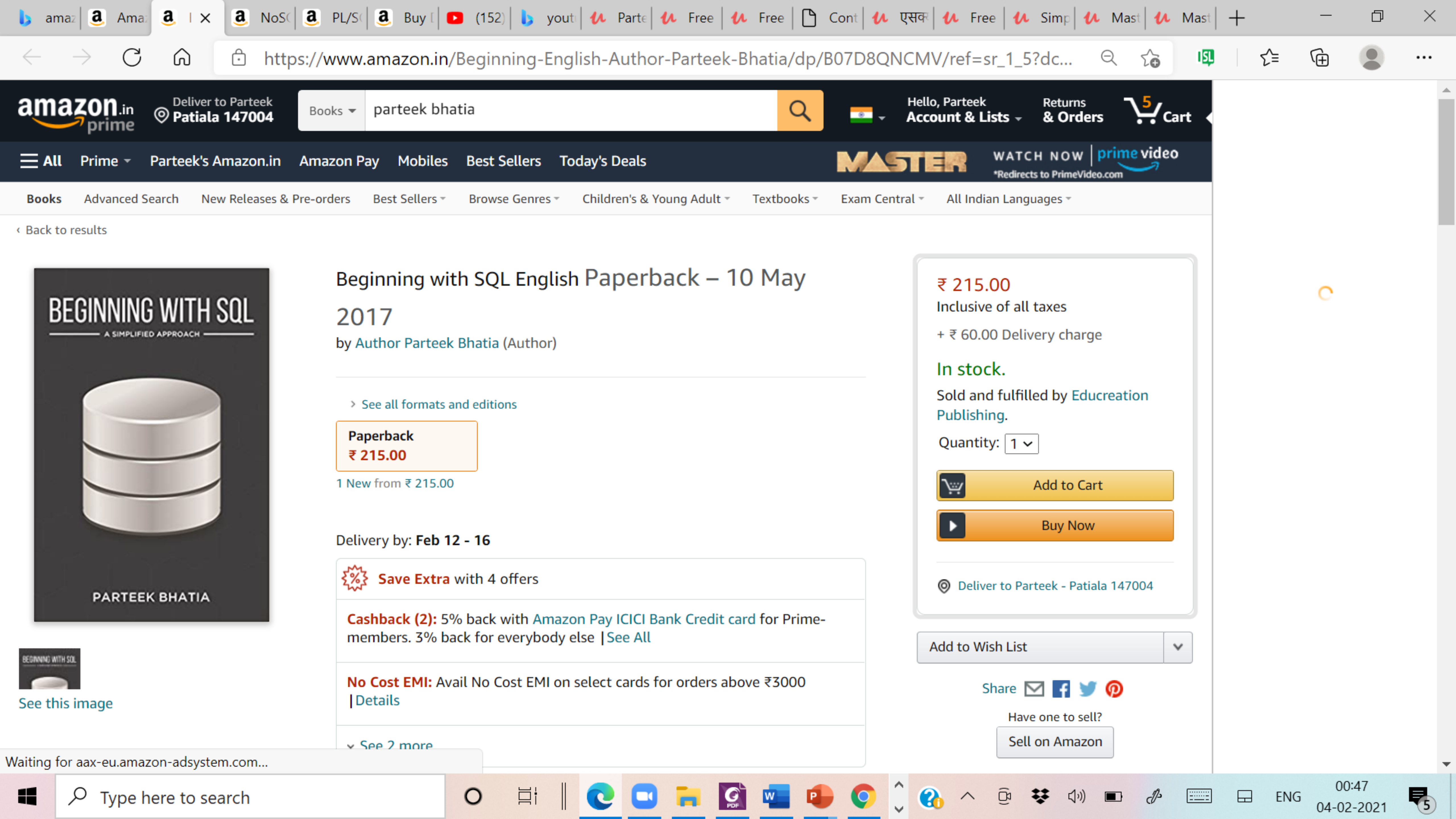
Sold by: Amazon Asia-Pacific Holdings Private Limited

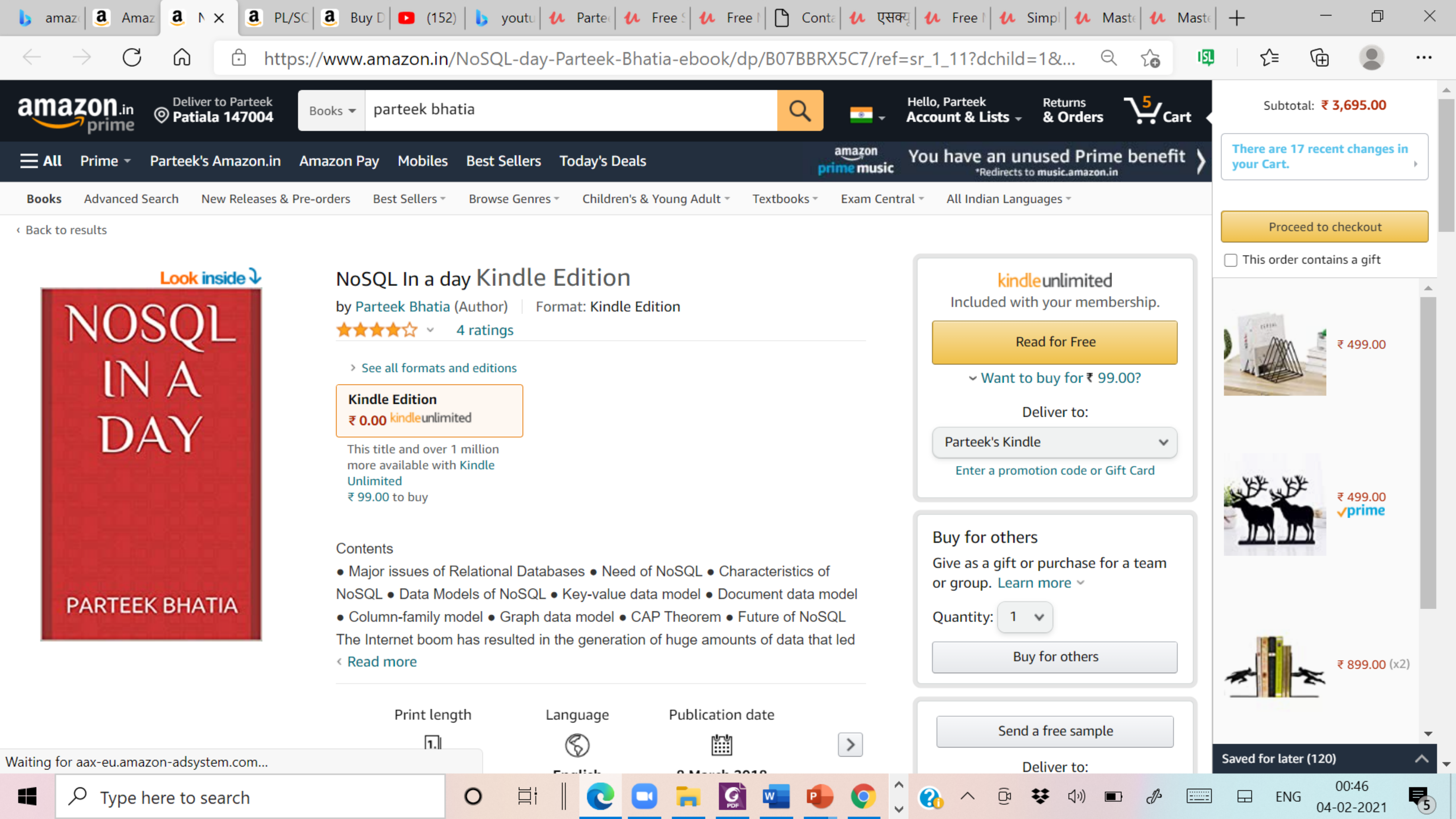
kindleunlimited
Included with your membership.

Read Now

You own this item. It's available on your Kindle apps and devices.
Enter a promotion code or Gift Card

Buy for others
Give as a gift or purchase for a team or group. Learn more


https://www.amazon.in/Beginning-English-Author-Parteek-Bhatia/dp/B07D8QNCMV/ref=sr_1_5?dc...
Deliver to Parteek
Patiala 147004
Books
parteek bhatia
Hello, Parteek
Account & Lists
Returns
& Orders
Cart
All
Prime
Parteek's Amazon.in
Amazon Pay
Mobiles
Best Sellers
Today's Deals
WATCH NOW
prime video
*Redirects to PrimeVideo.com
Books
Advanced Search
New Releases & Pre-orders
Best Sellers
Browse Genres
Children's & Young Adult
Textbooks
Exam Central
All Indian Languages
‹ Back to results
BEGINNING WITH SQL
A SIMPLIFIED APPROACH
PARTEEK BHATIA
See this image
Beginning with SQL English Paperback – 10 May 2017
by Author Parteek Bhatia (Author)
See all formats and editions
Paperback
₹ 215.00
1 New from ₹ 215.00
Delivery by: Feb 12 - 16
Save Extra with 4 offers
Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All
No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details
See 2 more
₹ 215.00
Inclusive of all taxes
+ ₹ 60.00 Delivery charge
In stock.
Sold and fulfilled by Educreation Publishing.
Quantity: 1
Add to Cart
Buy Now
Deliver to Parteek - Patiala 147004
Add to Wish List
Share
Have one to sell?
Sell on Amazon
Waiting for aax-eu.amazon-adsystem.com...
Type here to search
ENG
00:47
04-02-2021


https://www.amazon.in/NoSQL-day-Parteek-Bhatia-ebook/dp/B07BBRX5C7/ref=sr_1_11?dchild=1&...
Deliver to Parteek
Patiala 147004
Books
parteek bhatia
Hello, Parteek
Account & Lists
Returns
& Orders
5
Cart
All
Prime
Parteek's Amazon.in
Amazon Pay
Mobiles
Best Sellers
Today's Deals
You have an unused Prime benefit
*Redirects to music.amazon.in
Books
Advanced Search
New Releases & Pre-orders
Best Sellers
Browse Genres
Children's & Young Adult
Textbooks
Exam Central
All Indian Languages
‹ Back to results
Look inside
NOSQL IN A DAY
PARTEEK BHATIA
NoSQL In a day Kindle Edition
by Parteek Bhatia (Author) Format: Kindle Edition
4 ratings
See all formats and editions
Kindle Edition
₹ 0.00 kindleunlimited
This title and over 1 million more available with Kindle Unlimited
₹ 99.00 to buy
Contents
● Major issues of Relational Databases ● Need of NoSQL ● Characteristics of NoSQL ● Data Models of NoSQL ● Key-value data model ● Document data model ● Column-family model ● Graph data model ● CAP Theorem ● Future of NoSQL
The Internet boom has resulted in the generation of huge amounts of data that led
Read more
Print length
Language
Publication date
kindleunlimited
Included with your membership.
Read for Free
Want to buy for ₹ 99.00?
Deliver to:
Parteek's Kindle
Enter a promotion code or Gift Card
Buy for others
Give as a gift or purchase for a team or group. Learn more
Quantity: 1
Buy for others
Send a free sample
Deliver to:
Subtotal: ₹ 3,695.00
There are 17 recent changes in your Cart.
Proceed to checkout
This order contains a gift
₹ 499.00
₹ 499.00
prime
₹ 899.00 (x2)
Saved for later (120)
Waiting for aax-eu.amazon-adsystem.com...
Type here to search
ENG
00:46
04-02-2021

For more information

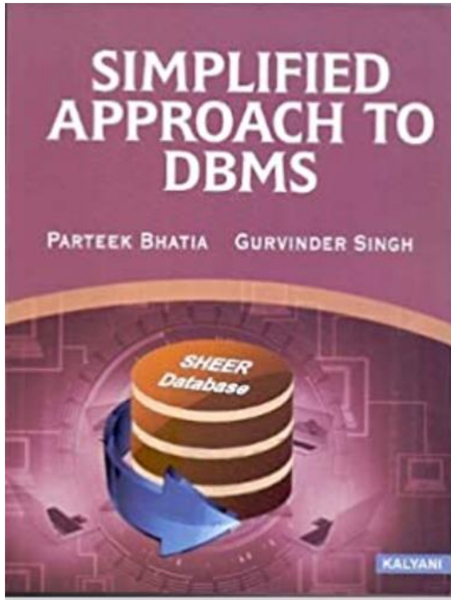
- Please visit: www.parteekbhatia.com
- For Feedback: parteek.bhatia@gmail.com
- For online Video Courses: https://www.udemy.com/user/parteek-kumar-4/
- For YouTube Channel:
- https://www.youtube.com/user/parteekbhatia
- For Books: Amazon.in : Parteek Bhatia
- For Workshops and invited talks, please write to:

parteek.bhatia@gmail.com

Thanks

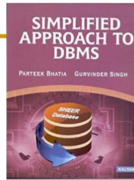
Chapter: 2
Architecture of Database Management System

By
Parteek Bhatia
Associate Professor
Department of Computer Science & Engineering
Thapar Institute of Engineering and Technology
Patiala



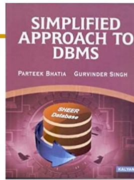
Early Two Level Architecture

- Given by Data Base Task Group (DBTG) in 1971.
- Early attempt to standardize database development.
- Recommend two level approach.
  - Schema
  - Subschema

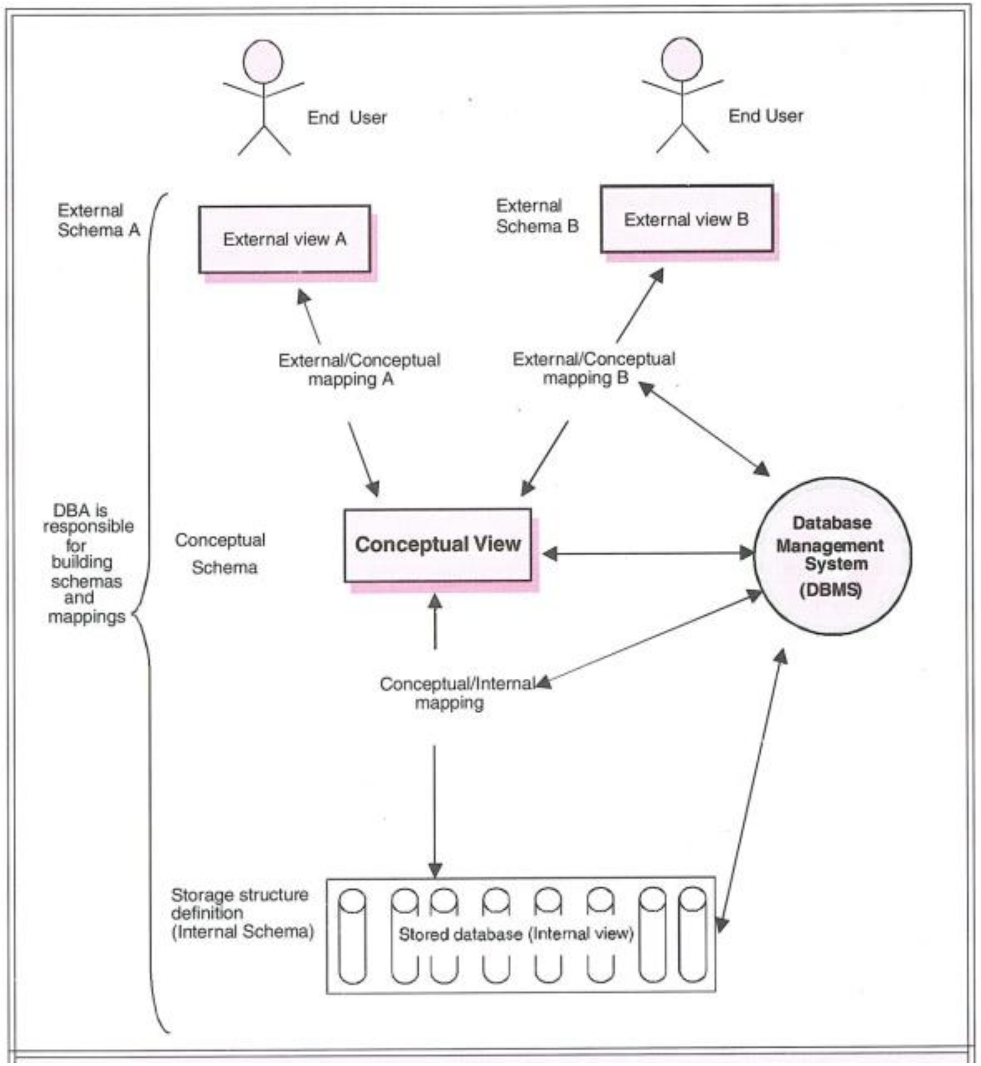


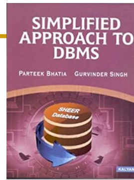
Objective of the Three Level Architecture

- Each user should be able to access the same data, but have a different customized view of the data.
- User's interaction with the database should be independent of storage considerations.
- The Database Administrator (DBA) should be able to change the database storage structures without affecting the user's views.
- The internal structure of the database should be unaffected by changes to the physical aspects of storage, such as the changeover to a new storage device.
- The DBA should be able to change the conceptual structure of the database without affecting all users.



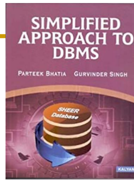
Three Level Architecture of DBMS





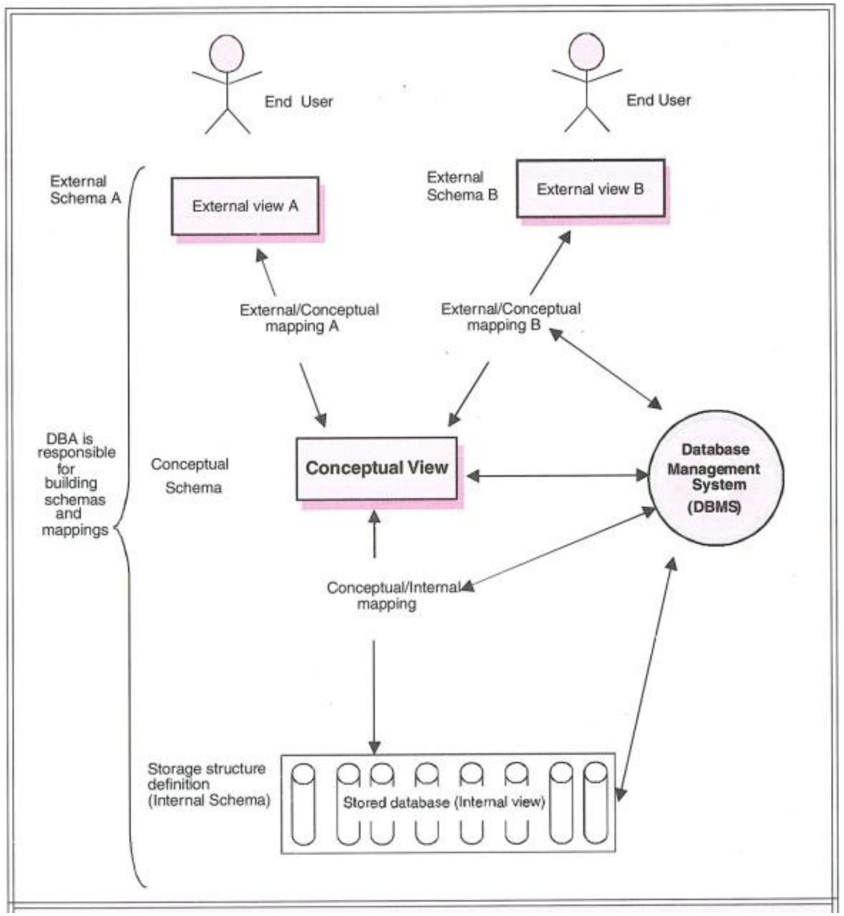
Three Level Architecture of DBMS

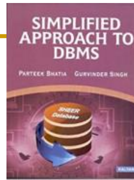
- External Level
- Conceptual Level
- Internal Level



External Level or View level

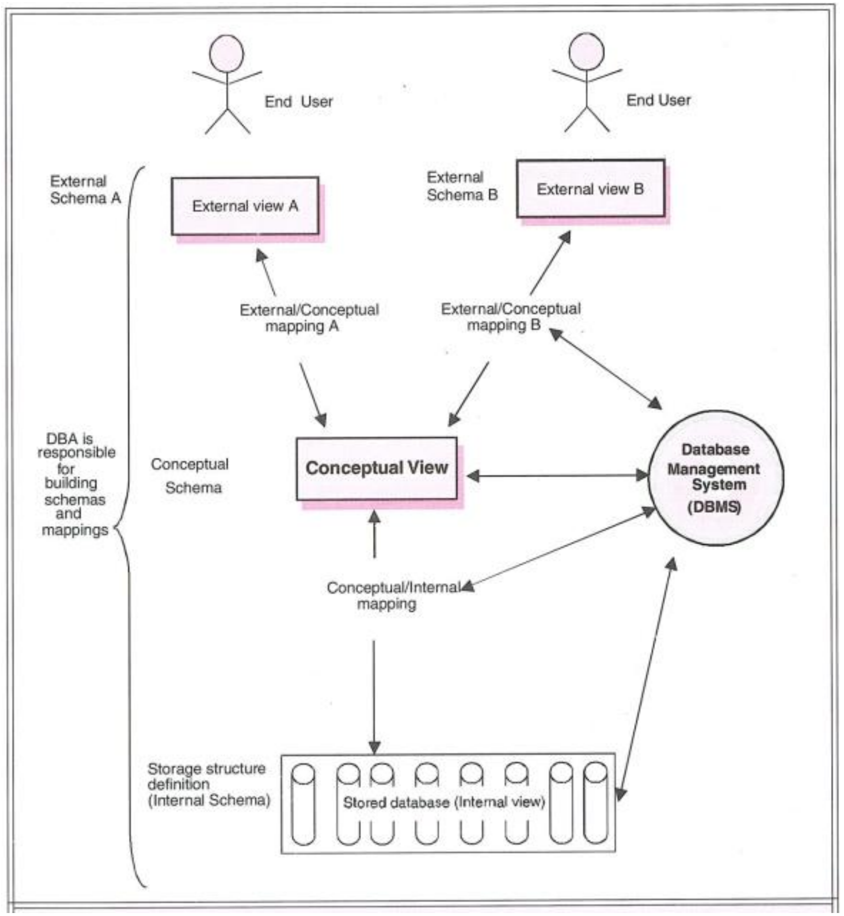
- It is the users' view of the database.
- It describes that part of the database that is relevant to each user.
- It is closest to the end users.
- External level is also known as the view level.

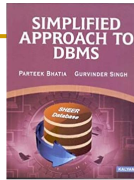




External Level or View level

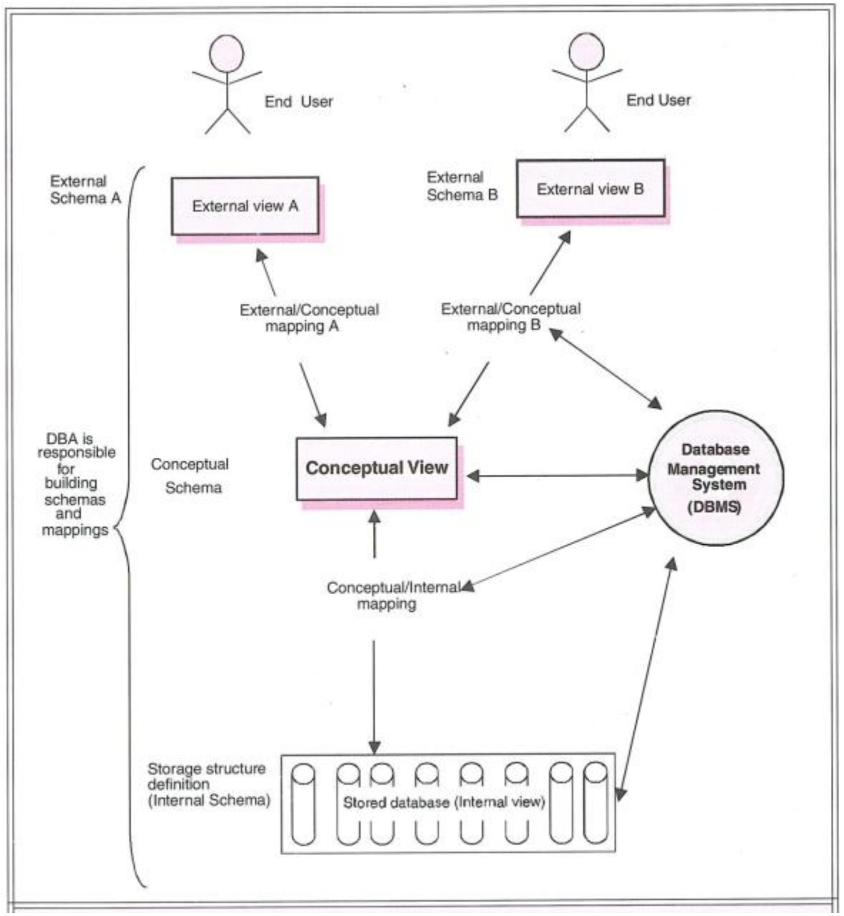
- It deals with the way in which individual users view data.
- Individual users are given different views according to their requirements.
- Example: one user may view dates in the form (day, month, year), while another may view dates as (year, month, day).
- One may be interested in First Name other may be in First Name and Last Name.

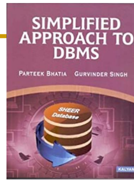




Conceptual Level or Logical Level

- This level must not contain any storage dependent details.
- For example it contains only information about data type and its size but not any storage considerations, such as the number of bytes occupied.
- Conceptual level is also known as the logical level.

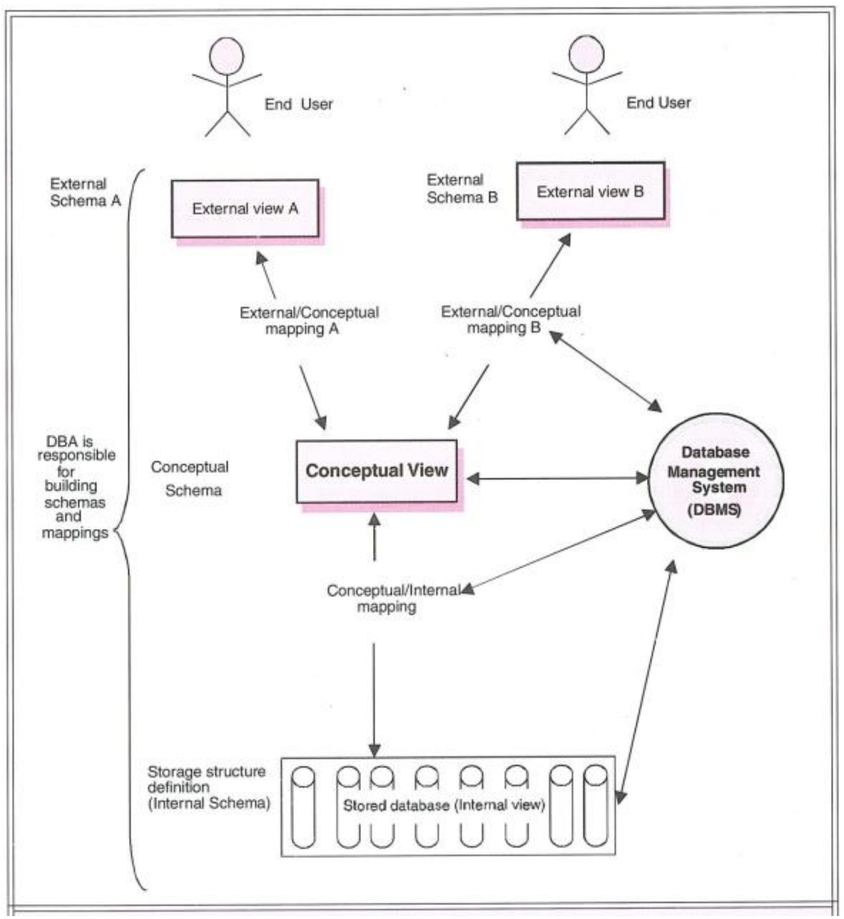




Conceptual Level or Logical Level

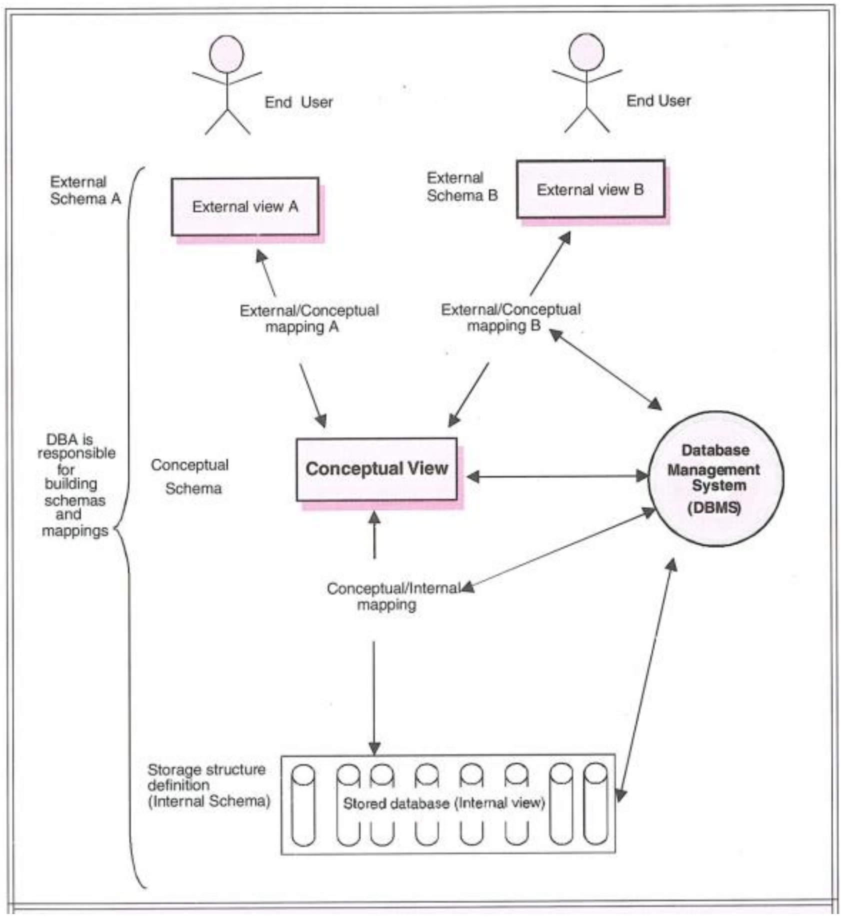
This level deals with

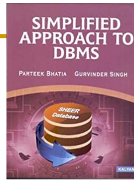
- All entities, their attributes, and their relationships
- The constraints on the data
- Security and integrity information



Internal Level or Storage level

- It is the physical representation of the database on the computer.
- This level describes how the data is stored in the database.
- It concerns the way the data are physically stored on the hardware.

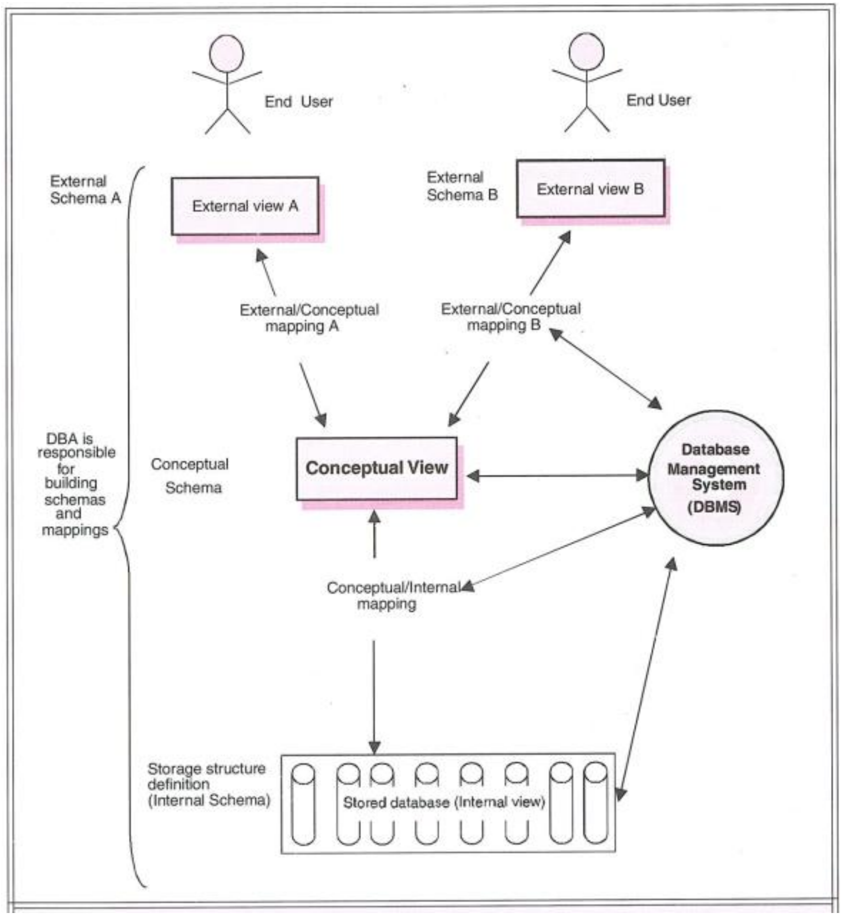


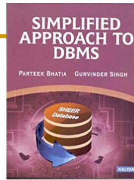


Internal Level or Storage level

The internal level is concerned with

- Storage space allocation for data and indexes;
- Record descriptions for storage (with stored sizes for data items);
- Record placement;
- Data compression and data encryption techniques.



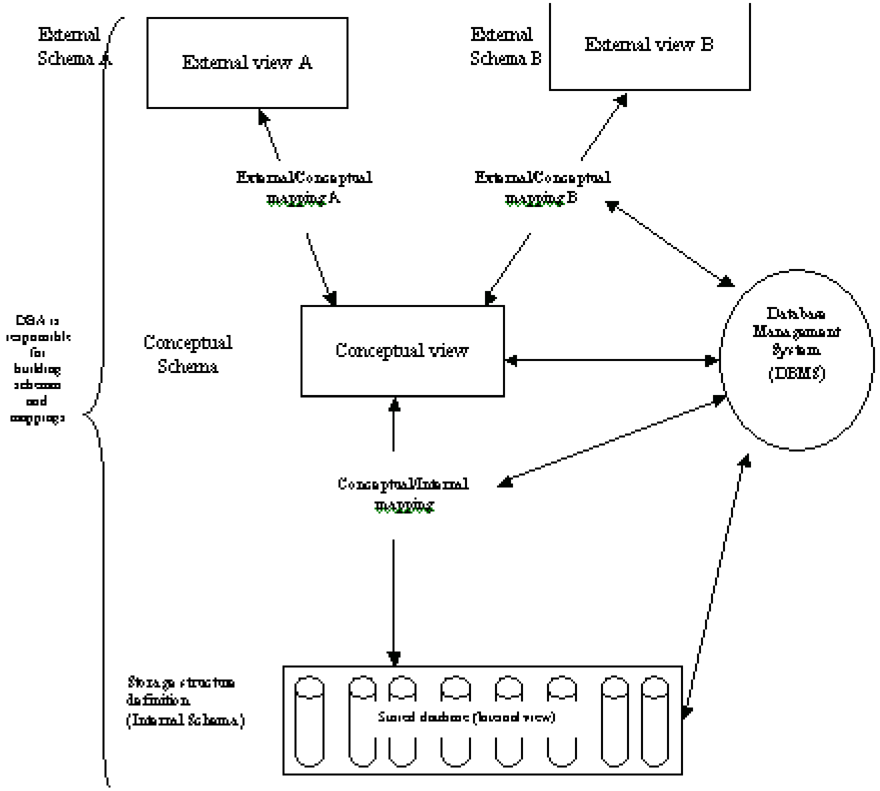


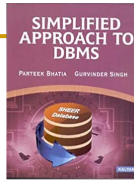
Three Level Architecture
Mapping between the Views

Mapping between Views

External/Conceptual Mapping:

- Each external schema is related to the conceptual schema by the external/conceptual mapping.
- It maps logical record in the external view to one (or more) conceptual record(s) in the conceptual view.

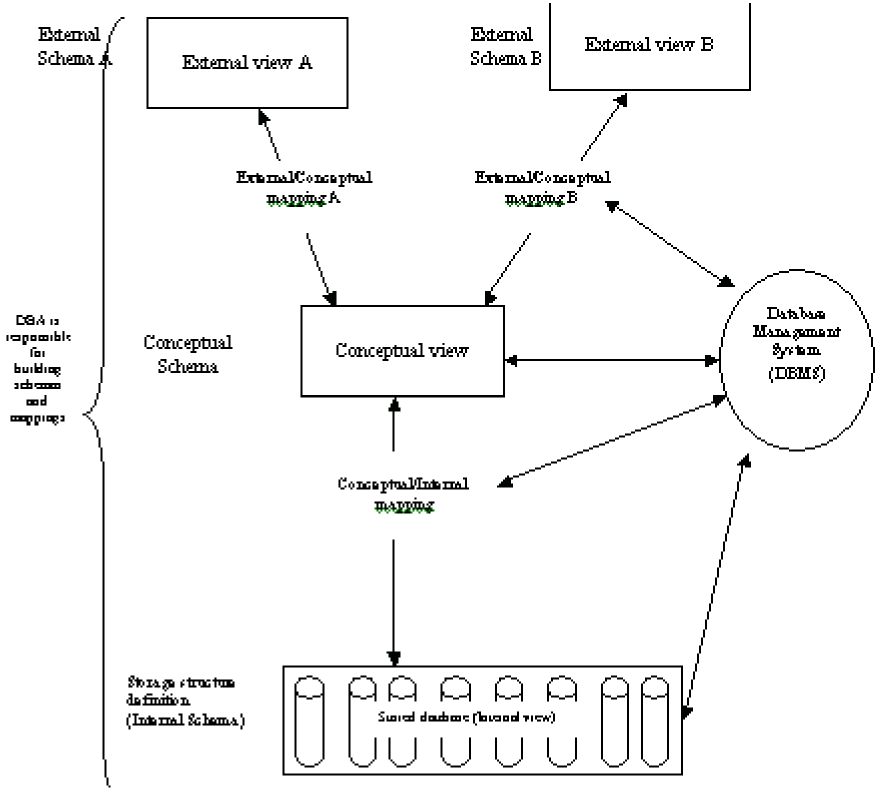


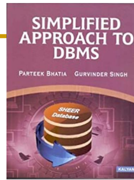


Mapping between Views

Conceptual/Internal Mapping:

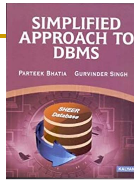
- Conceptual schema is related to the internal schema by the conceptual/internal mapping.
- This enables the DBMS to find the actual record or combination of records in physical storage that constitute a logical record in conceptual schema.





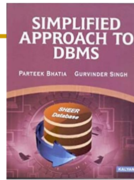
Schema

- **The overall description of the database is called the Database Schema**.
- There are three different types of schema in the database corresponding to each data view of database. In other words, the data views at each of three levels are described by schema.

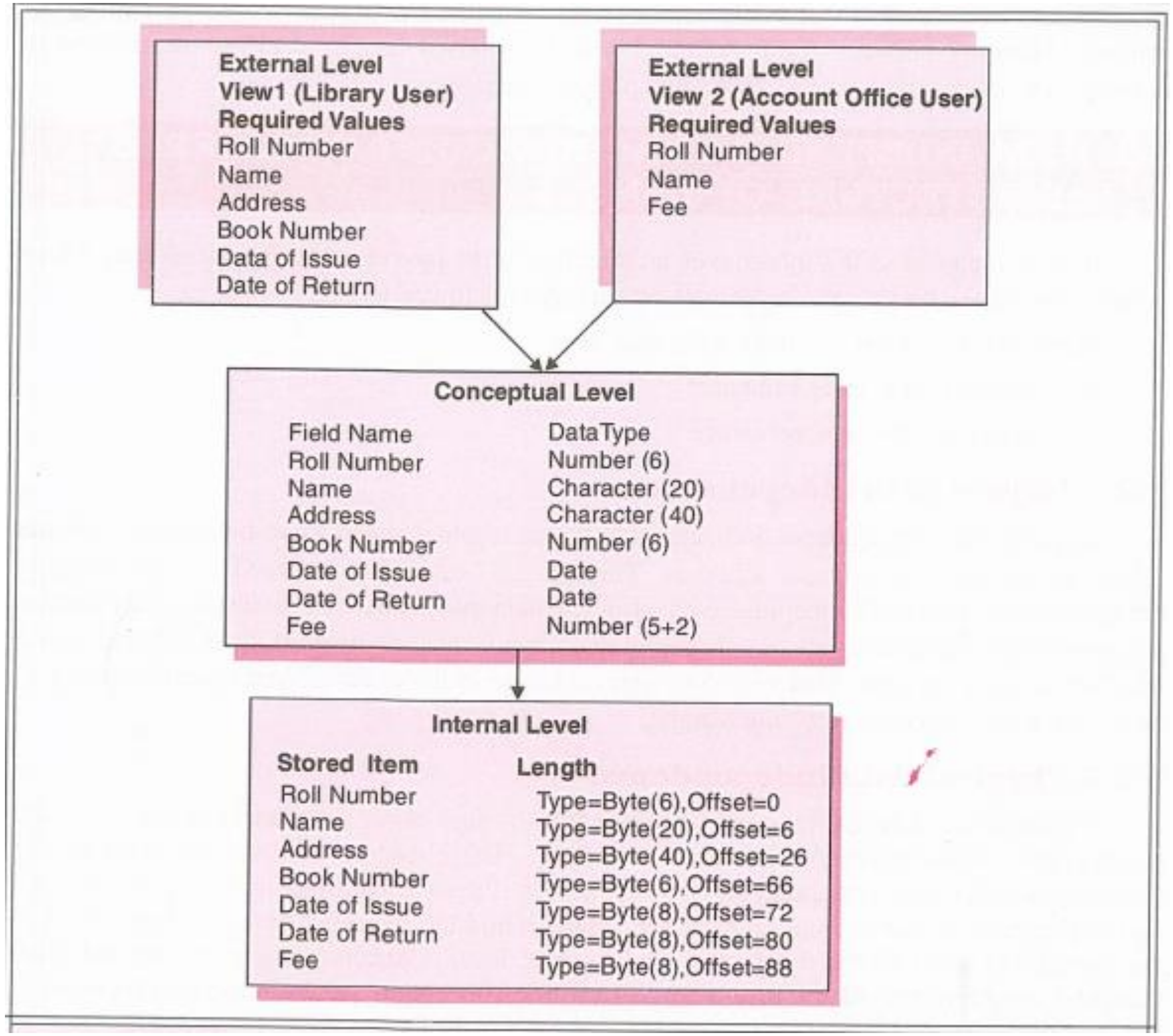


Schema

- The External view is described by means of a schema called **external schema** that correspond to different views of the data.
- Similarly the Conceptual view is defined by **conceptual schema**, which describes all the entities, attributes, and relationship together with integrity constraints.
- Internal View is defined by **internal schema**, which is a complete description of the internal model, containing definition of stored records, the methods of representation, the data fields, and the indexes used.

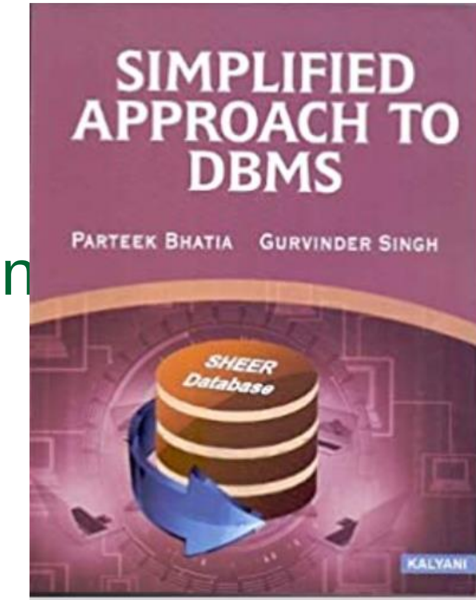


Example: University Management System



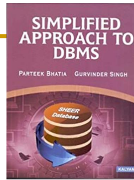
Data Independence
Achievement of Three Level Architecture

By
Parteek Bhatia
Associate Professor
Department of Computer Science & Engineerin
Thapar University
Patiala



Data Independence

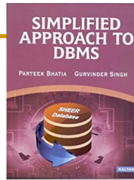
- Application Programs should be independent from the way the data is stored and accessed.



Data Independence-Achievement of Layered Architecture of DBMS

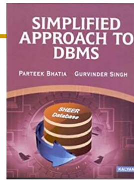
There are two kinds of data independence:

- Logical data independence
- Physical data independence



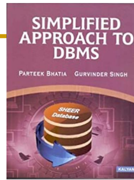
Logical data independence

Logical data independence indicates that the conceptual schema can be changed without affecting the existing external schemas. The change would be absorbed by the mapping between the external and conceptual levels.



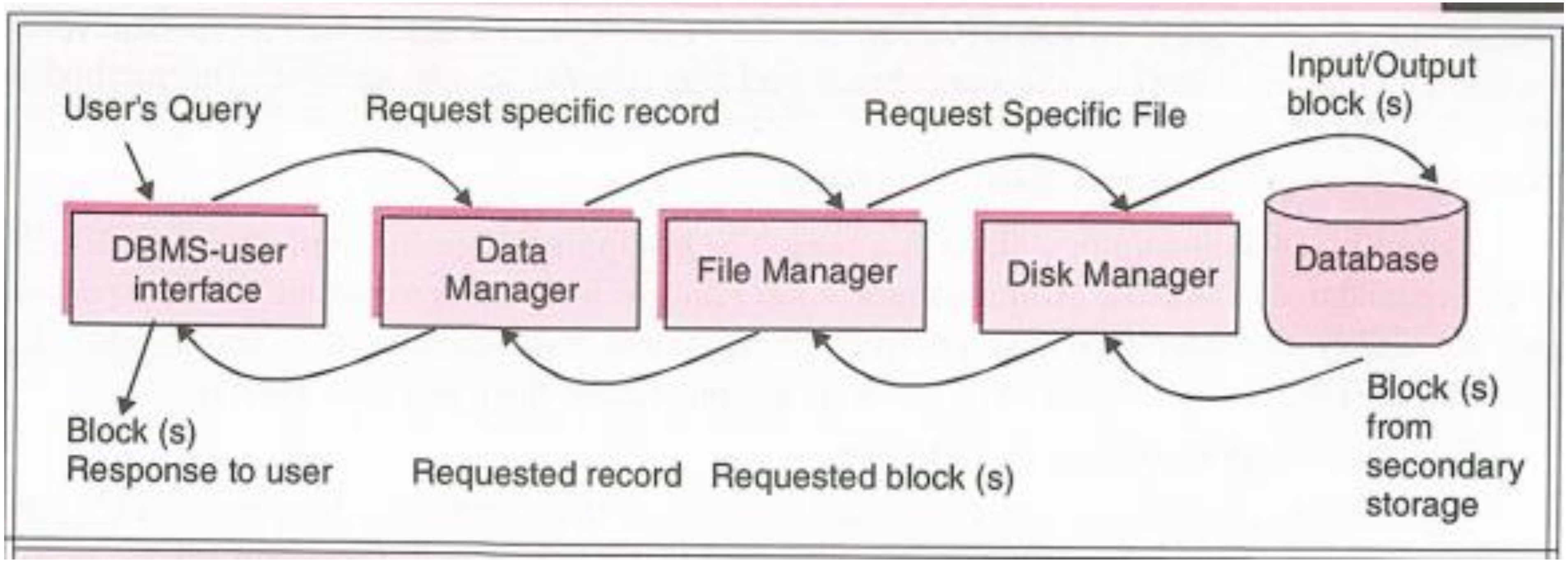
Physical data independence

Physical data independence indicates that the physical storage structures or devices could be changed without affecting conceptual schema. The change would be absorbed by the mapping between the conceptual and internal levels.



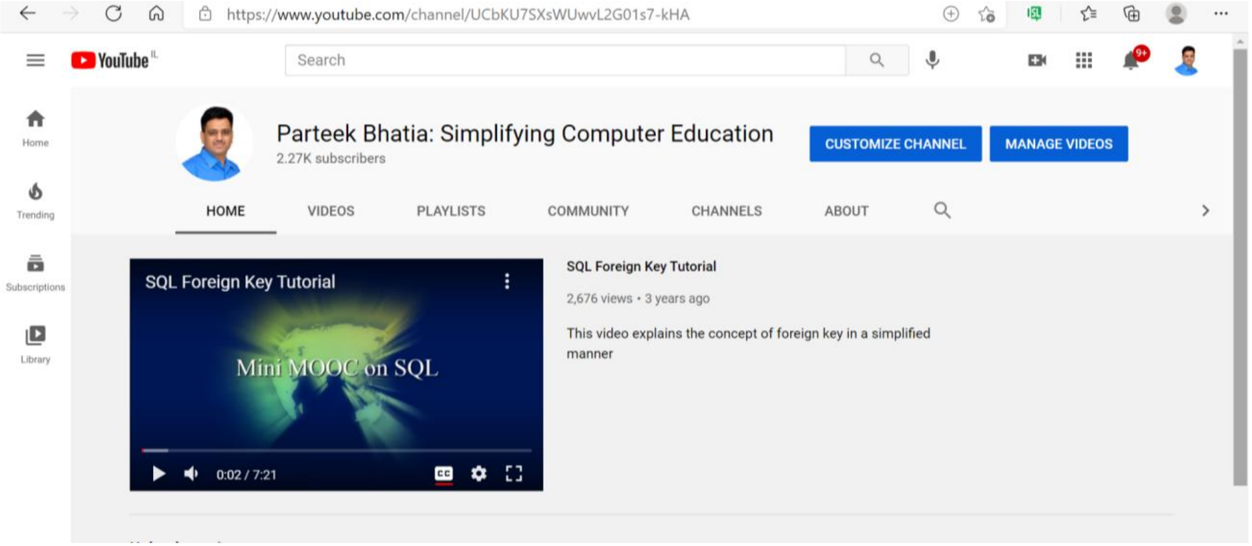
Procedure for Data Access by DBMS

Procedure for Data Access by DBMS



For more information

- Subscribe to YouTube Channel from the Author
  - To receive latest video tutorials on Data Mining, Machine Learning, DBMS, Big Data, NoSQL and many more.
- https://www.youtube.com/user/parteekbhatia



amazon simplified approach to d

Amazon.in: Buy Simplified Appro

https://www.amazon.in/Simplified-Approach-Gurvinder-Bhatia-Parteek/dp/B07WCJJ18B

amazon.in prime

Deliver to Parteek
Patiala 147004

Books

Hello, Parteek
Account & Lists

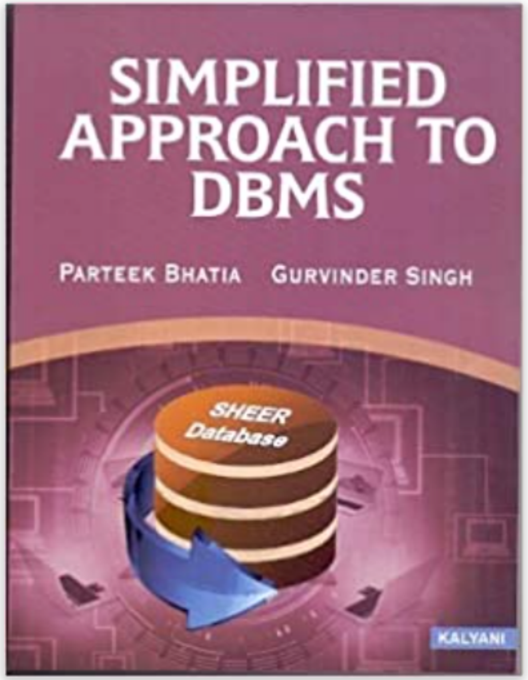
Returns
& Orders

5 Cart

Subtotal
₹ 3,695.00

All | Prime | Parteek's Amazon.in | Amazon Pay | Mobiles | Best Sellers | Today's Deals

Books | Advanced Search | New Releases & Pre-orders | Best Sellers | Browse Genres | Children's & Young Adult | Textbooks | Exam Central | All Indian Languages



See this image

Simplified Approach to DBMS Unbound – 1 January 2019

by Gurvinder Singh Bhatia Parteek (Author)

2 ratings

See all formats and editions

Unbound
₹ 400.00

3 New from ₹ 400.00

Delivery By: **Feb 8 - 9** Details

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

₹ 400.00

M.R.P.: ~~₹ 450.00~~

You Save: ₹ 50.00 (11%)

Inclusive of all taxes

FREE Delivery. Details

In stock.

Sold by Lyall Book Depot..

Quantity: 1

Add to Cart

Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

Share

Type here to search

ENG 00:14 04-02-2021

Some of online Video Courses created by Author over Udemy Platform

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Teach on Udemy Log in Sign up

Development > Database Design & Development > Database Management

Simplified Approach to DBMS

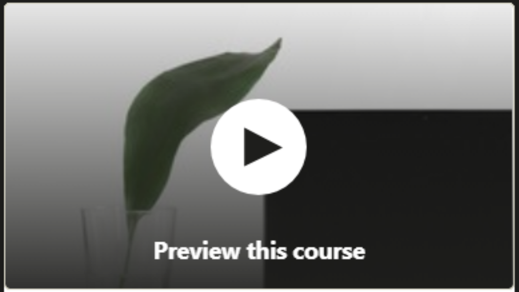
Simplified way to learn DBMS

4.3 ★★★★☆ (2 ratings) 14 students

Created by Parteek Bhatia

Last updated 11/2020 English

Wishlist | Share | Gift this course



Personal | Teams

₪79.90

Add to cart

Buy now

30-Day Money-Back Guarantee

This course includes:

- 9.5 hours on-demand video
- Full lifetime access

What you'll learn

- Understanding the meaning of term DBMS
- File Based System Vs. Data Base Management System
- Hierarchical, Network and Relational Data Models
- Concept of different types of keys like Super Key, Candidate Key, Primary Key and Foreign Key
- Designing of Entity Relation Model
- Conversion of ER Model to Tables
- SQL: Creation of Tables
- SQL: Insertion, Updation, Deletion and Retrieval of Data

Online Course on SQL at Udemy

Udemy

ROYAL ACADEMY OF ENGINEERING | Funded

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

Simplified Approach to SQL

QUICK START
Ace Interviews and College Exams in 1 month

BI - LINGUAL
Explanation of each of the topic in English and Hindi

INTERACTIVE
Discussion forums and direct messaging to instructor

RELAXING
Study the course at your own pace

STUDY ANYWHERE
Study on your smartphone with udemy app

COURSE CONTENT

Week 1 : *Introduction to SQL and performing basic operations with SQL.*
Week 2 : *Creation of Tables with Integrity constraints.*
Week 3 : *Table Alterations and Joins.*
Week 4 : *Grouping of Data*

ABOUT THE INSTRUCTOR

Dr. Parteek Bhatia is Associate Professor in the Department of Computer Science and Engineering at Thapar Institute of Engineering and Technology, Patiala. He has more than 18 years of academic experience.He has authored several books in various areas of computer science. His book - Simplified approach to DBMS is one of the bestseller. Currently, he is working on plethora of Projects which are funded by Department of Science and Technology, CSIR and other funding agencies of India.

QR CODE

COURSE LINK: https://www.udemy.com/learn-sql-in-a-simplified-manner/

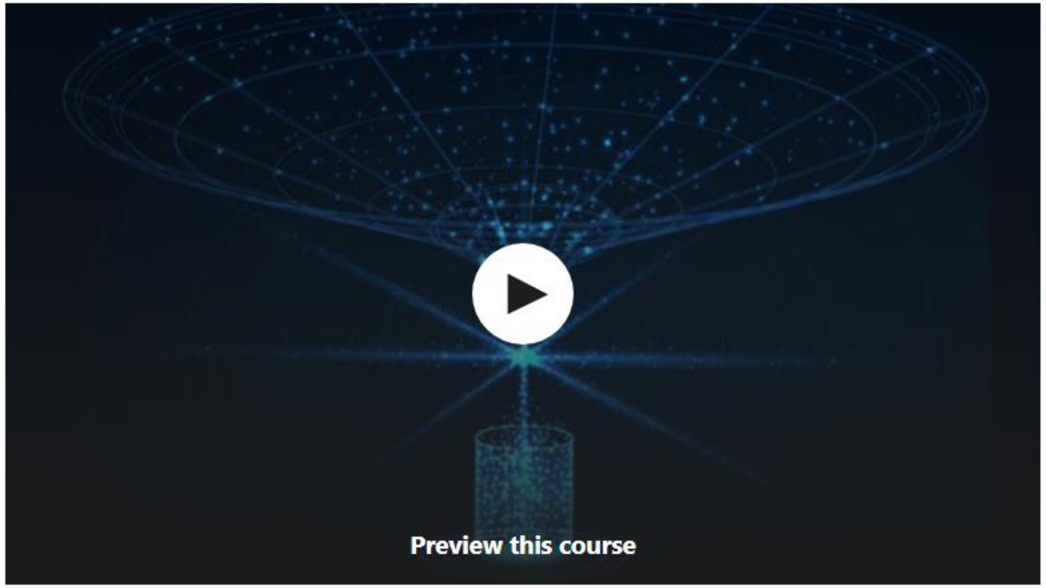
ama | Ama | (152 | http | Part | Free | Free | Free | एसक | Free | Sim | Mas

https://www.udemy.com/course/learn-sql-in-a-simplified-manner/

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Instructor My learning

Development > Programming Languages > SQL



Learn SQL in a simplified manner

Learn SQL in just four weeks, right from scratch !

4.1 ★★★★☆ (467 ratings) 12,875 students 1hr 56min of on-demand video

Created by Parteek Bhatia

English English [Auto]

Free

Enroll now

What you'll learn Course content Reviews Instructors

✓ SQL is one of the most in demand tech skills today, and learning SQL would surely be one of the fastest and most efficient way to improve ones career prospectus. After the completion of course, one will be able to read, write and comprehend simple to most complex queries of SQL, using which they could work upon any SQL database such as Oracle, Amazon Redshift and others.

Type here to search ENG 00:19 04-02-2021

Teaching & Academics > Online Education > Pl/SQL

Mastering PL/SQL

Simplified way to learn PL/SQL

Highest Rated 4.7 ★★★★☆ (17 ratings) 110 students

Created by Parteek Bhatia

Last updated 5/2020 English English [Auto]

Wishlist | Share | Gift this course

Personal | Teams

₪79.90

Add to cart

Buy now

30-Day Money-Back Guarantee

This course includes:

9.5 hours on-demand video

What you'll learn

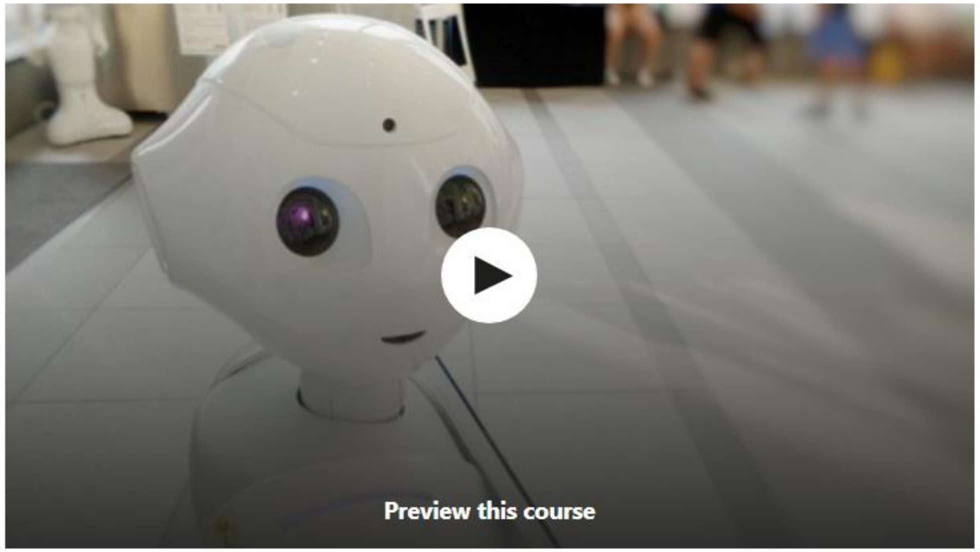
- ✓ Basics of PL/SQL
- ✓ Variables and Constants
- ✓ Conditional, Looping and Sequential Statements
- ✓ Exception Handling
- ✓ Cursor Handling
- ✓ Sub-Programs, Functions and Procedures
- ✓ Triggers
- ✓ DBMS:: Get chance to enroll in Simplified Approach to DBMS Course completely free

https://www.udemy.com/course/mastering-machine-learning-course-1/

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Instructor My learning

Development > Data Science > Machine Learning



Mastering Machine Learning: Course-1

Taking First Step Towards Machine Learning

3.9 ★★★★☆ (21 ratings) 3,425 students 1hr 52min of on-demand video

Created by Parteek Bhatia

English English [Auto]

Free

Enroll now

What you'll learn Course content Reviews Instructors

- ✓ Machine Learning
- ✓ Python
- ✓ Regression
- ✓ Classification
- ✓ Unsupervised Machine Learning

Type here to search ENG 00:20 04-02-2021

https://www.udemy.com/course/genetic-algorithm-for-machine-learning/

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Instructor My learning

IT & Software > Other IT & Software > Genetic Algorithm

Genetic Algorithm for Machine Learning

Simplified Way to Learn

New 0.0 (0 ratings) 447 students 1hr 32min of on-demand video

Created by Parteek Bhatia

English

Free

Enroll now

What you'll learn | Course content | Reviews | Instructors

- ✓ Working Principle of Genetics Algorithms
- ✓ Natural Selection
- ✓ Implementation of Natural Selection through Roulette Wheel
- ✓ Crossover or Recombination
- ✓ Concept of Probability of Crossover and Its usage in generation of Population

Touchpad

Type here to search ENG 00:20 04-02-2021

Association Mining for Machine Learning

Simplified Way to Learn

New 4.6 ★★★★★ (13 ratings) 629 students 1hr 48min of on-demand video

Created by Parteek Bhatia

English English [Auto]

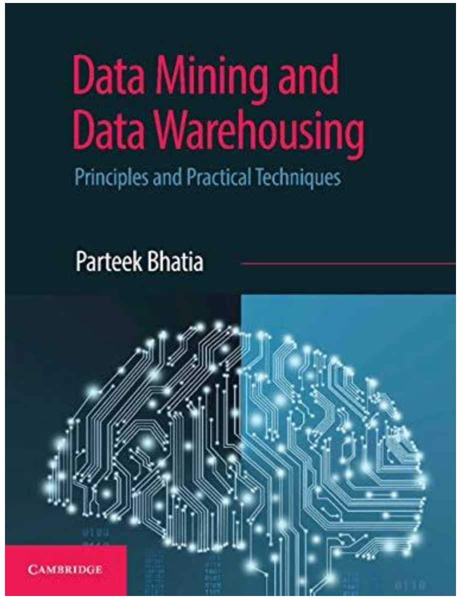
Free

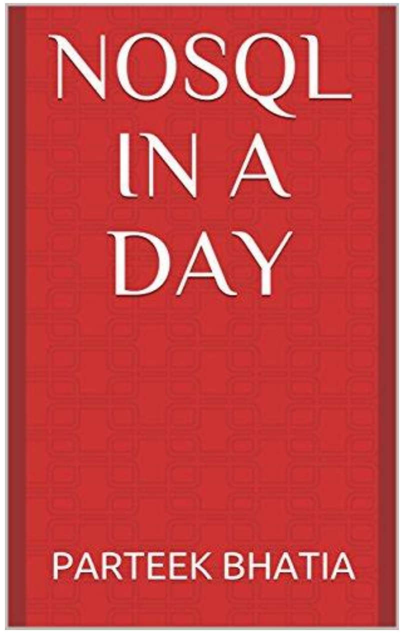
Enroll now

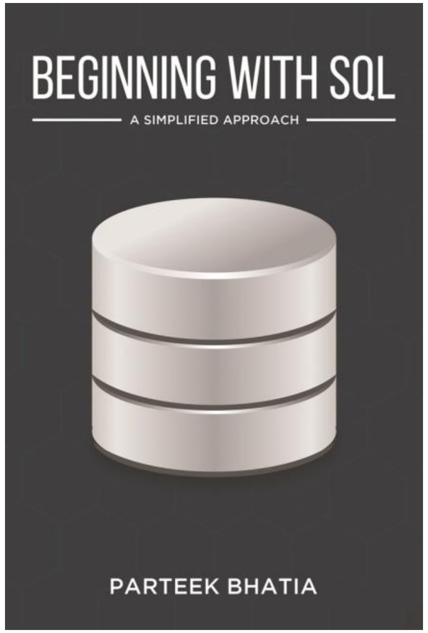
What you'll learn Course content Reviews Instructors

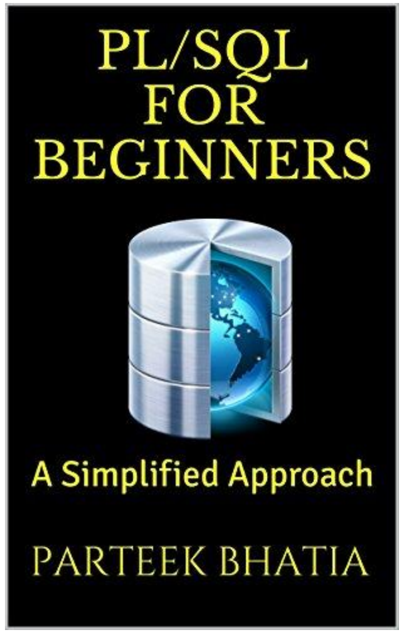
- ✓ Association Mining
- ✓ Naive Algorithm
- ✓ Apriori Algoorithm

Books from the Same Author





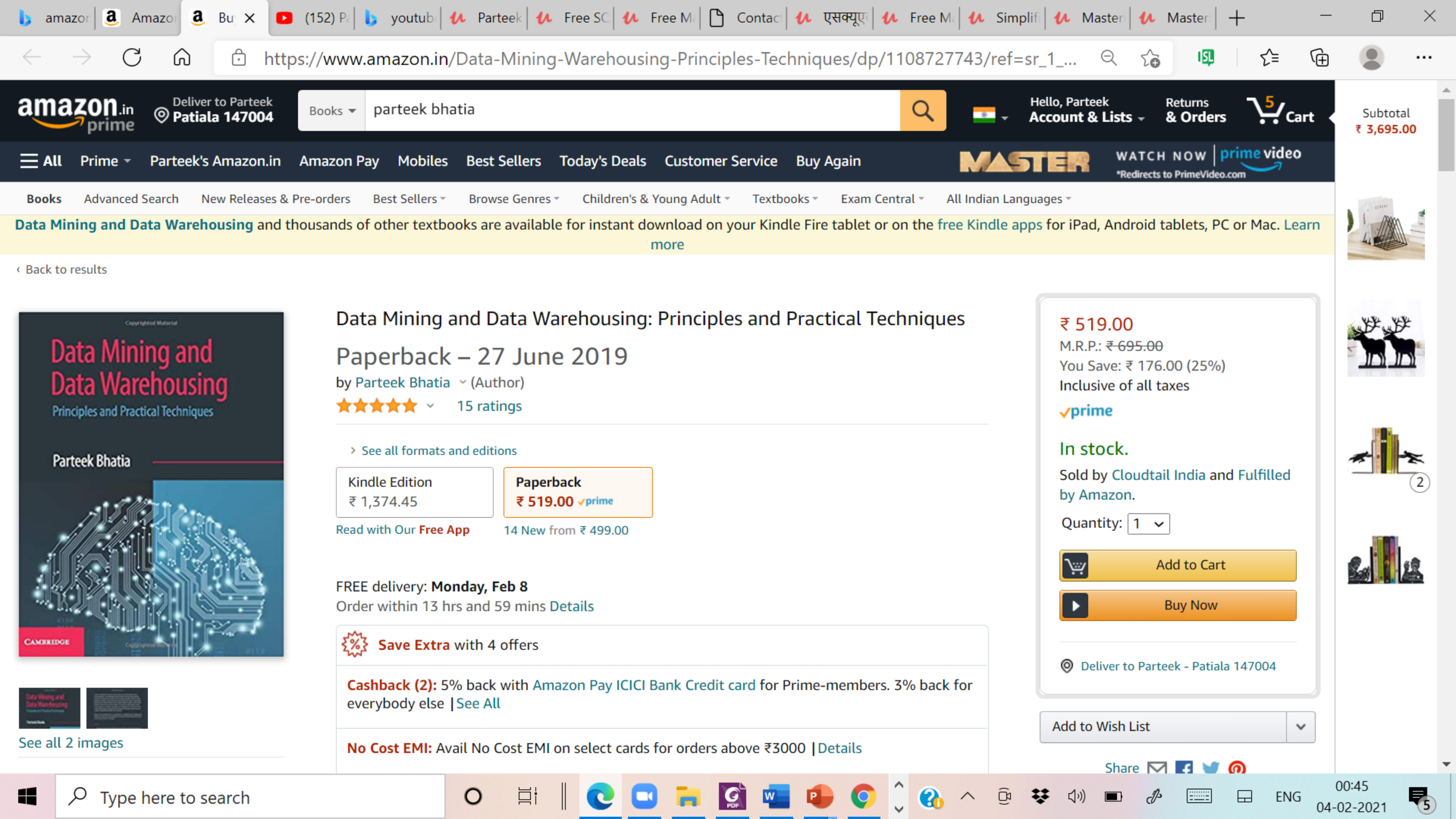




For more information visit: www.parteekbhatia.com

Coming Soon…

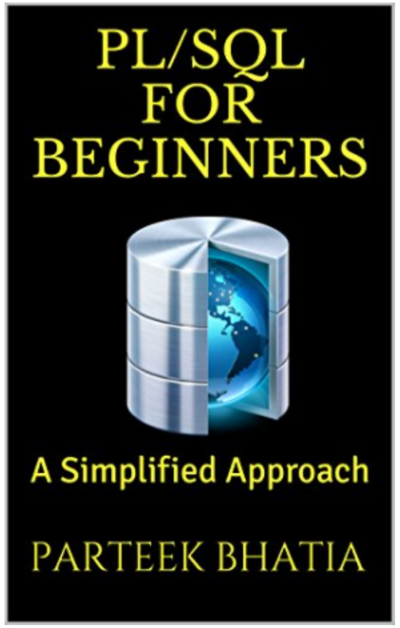
- **Text Book**
- **Machine Learning: Principles and Practical Techniques**



https://www.amazon.in/Data-Mining-Warehousing-Principles-Techniques/dp/1108727743/ref=sr_1_...
amazon.in prime
Deliver to Parteek
Patiala 147004
Books
parteek bhatia
Hello, Parteek
Account & Lists
Returns
& Orders
5
Cart
Subtotal
₹ 3,695.00
All
Prime
Parteek's Amazon.in
Amazon Pay
Mobiles
Best Sellers
Today's Deals
Customer Service
Buy Again
MASTER
WATCH NOW
prime video
*Redirects to PrimeVideo.com
Books
Advanced Search
New Releases & Pre-orders
Best Sellers
Browse Genres
Children's & Young Adult
Textbooks
Exam Central
All Indian Languages
Data Mining and Data Warehousing and thousands of other textbooks are available for instant download on your Kindle Fire tablet or on the free Kindle apps for iPad, Android tablets, PC or Mac. Learn more
‹ Back to results
Data Mining and Data Warehousing
Principles and Practical Techniques
Parteek Bhatia
CAMBRIDGE
See all 2 images
Data Mining and Data Warehousing: Principles and Practical Techniques
Paperback – 27 June 2019
by Parteek Bhatia (Author)
15 ratings
See all formats and editions
Kindle Edition
₹ 1,374.45
Paperback
₹ 519.00 prime
Read with Our Free App
14 New from ₹ 499.00
FREE delivery: Monday, Feb 8
Order within 13 hrs and 59 mins Details
Save Extra with 4 offers
Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All
No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details
₹ 519.00
M.R.P.: ₹ 695.00
You Save: ₹ 176.00 (25%)
Inclusive of all taxes
prime
In stock.
Sold by Cloudtail India and Fulfilled by Amazon.
Quantity: 1
Add to Cart
Buy Now
Deliver to Parteek - Patiala 147004
Add to Wish List
Share
Type here to search
ENG
00:45
04-02-2021

You purchased this item on 4 August, 2016.
View this order | You can find this title on your Amazon Kindle apps and devices, or the Manage Your Content and Devices page. Need additional help? Click here

Look inside



PL/SQL for Beginners: A Simplified Approach Kindle Edition

by Parteek Bhatia (Author) | Format: Kindle Edition

5 ratings

› See all formats and editions

Kindle Edition
₹ 0.00 kindleunlimited

This title and over 1 million more available with Kindle Unlimited
₹ 99.00 to buy

PL/SQL: A Simplified Approach
Why read this book?
This book covers the basics of PL/SQL with lots of real world examples.
This book will help to gain better skills, better placement and better future.
...plified manner.

Kindle Price: ₹ 99.00
inclusive of all taxes

includes free wireless delivery via ***Amazon Whispernet***

Sold by: Amazon Asia-Pacific Holdings Private Limited

kindleunlimited
Included with your membership.

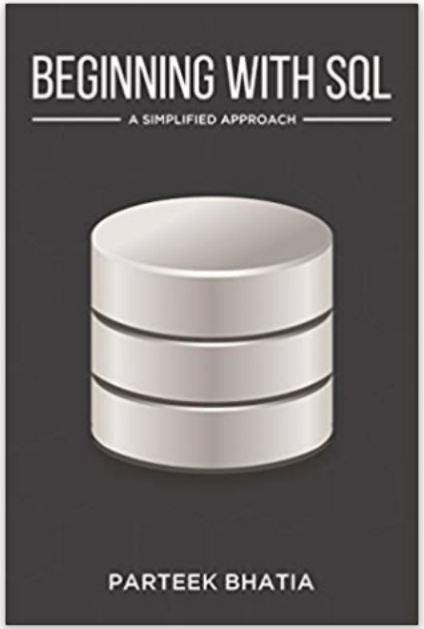
Read Now

You own this item. It's available on your Kindle apps and devices.
Enter a promotion code or Gift Card

Buy for others
Give as a gift or purchase for a team or group. Learn more

https://www.amazon.in/Beginning-English-Author-Parteek-Bhatia/dp/B07D8QNCMV/ref=sr_1_5?dc...

‹ Back to results



See this image

Beginning with SQL English Paperback – 10 May 2017

by Author Parteek Bhatia (Author)

› See all formats and editions

Paperback
₹ 215.00

1 New from ₹ 215.00

Delivery by: **Feb 12 - 16**

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

See 2 more

₹ 215.00
Inclusive of all taxes
+ ₹ 60.00 Delivery charge

In stock.

Sold and fulfilled by Educreation Publishing.

Quantity: 1

Add to Cart

Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

Share

Have one to sell?

Sell on Amazon

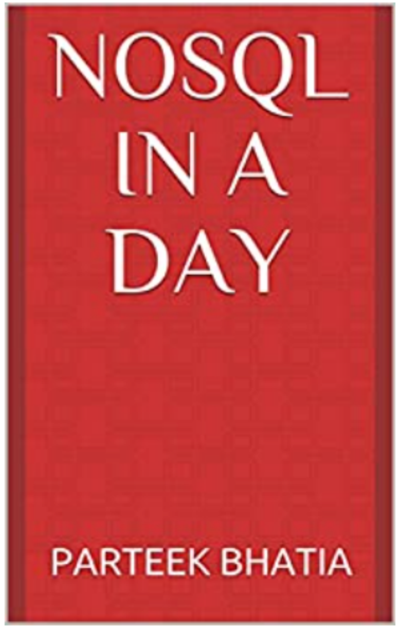
Waiting for aax-eu.amazon-adsystem.com...

Type here to search

ENG 00:47 04-02-2021

https://www.amazon.in/NoSQL-day-Parteek-Bhatia-ebook/dp/B07BBRX5C7/ref=sr_1_11?dchild=1&...

Look inside



NoSQL In a day Kindle Edition

by Parteek Bhatia (Author) | Format: Kindle Edition

★★★★☆ 4 ratings

› See all formats and editions

Kindle Edition
₹ 0.00 kindleunlimited

This title and over 1 million more available with Kindle Unlimited
₹ 99.00 to buy

Contents

● Major issues of Relational Databases ● Need of NoSQL ● Characteristics of NoSQL ● Data Models of NoSQL ● Key-value data model ● Document data model ● Column-family model ● Graph data model ● CAP Theorem ● Future of NoSQL
The Internet boom has resulted in the generation of huge amounts of data that led

‹ Read more

Print length	Language	Publication date
	English	8 March 2018

kindleunlimited
Included with your membership.

Read for Free

Want to buy for ₹ 99.00?

Deliver to:
Parteek's Kindle

Enter a promotion code or Gift Card

Buy for others

Give as a gift or purchase for a team or group. Learn more

Quantity: 1

Buy for others

Send a free sample

Deliver to:

Subtotal: ₹ 3,695.00

There are 17 recent changes in your Cart.

Proceed to checkout

This order contains a gift


₹ 499.00


₹ 499.00
prime


₹ 899.00 (x2)

Saved for later (120)

Waiting for aax-eu.amazon-adsystem.com...

Type here to search
ENG
00:46
04-02-2021

For more information

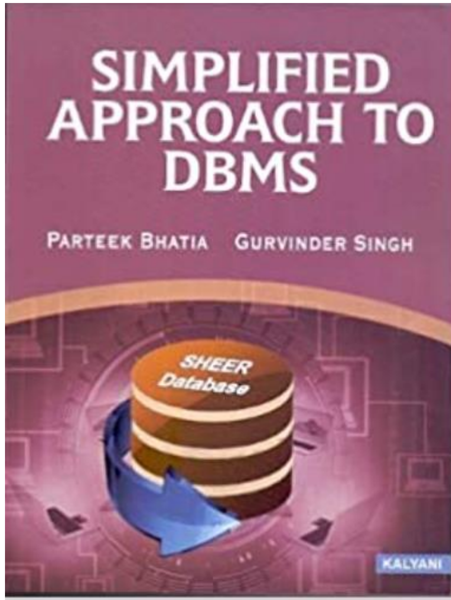
- Please visit: www.parteekbhatia.com
- For Feedback: parteek.bhatia@gmail.com
- For online Video Courses: https://www.udemy.com/user/parteek-kumar-4/
- For YouTube Channel:
- https://www.youtube.com/user/parteekbhatia
- For Books: Amazon.in : Parteek Bhatia
- For Workshops and invited talks, please write to:

parteek.bhatia@gmail.com

Thanks

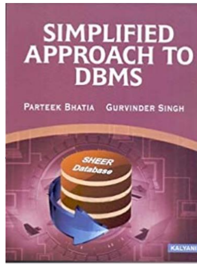
Chapter: 3
Data Models

By
Parteek Bhatia
Associate Professor
Department of Computer Science & Engineering
Thapar Institute of Engineering and Technology
Patiala



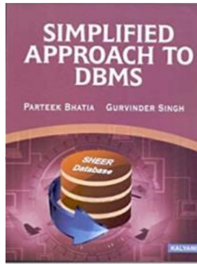
Data Models

- A model is a representation of reality, ‘real world’ objects and events, and their associations.
- It is an abstraction that concentrates on the essential.



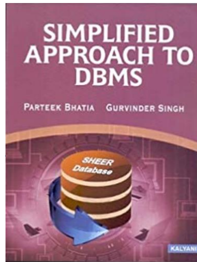
Types of Data Models

- Hierarchical Model
- Network Model
- Relational Model

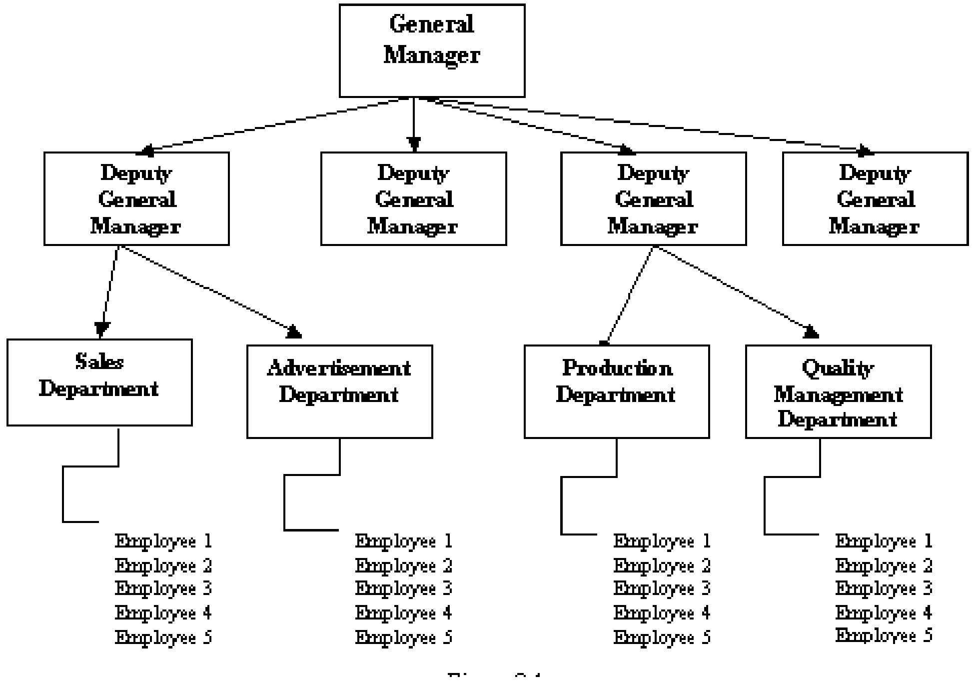


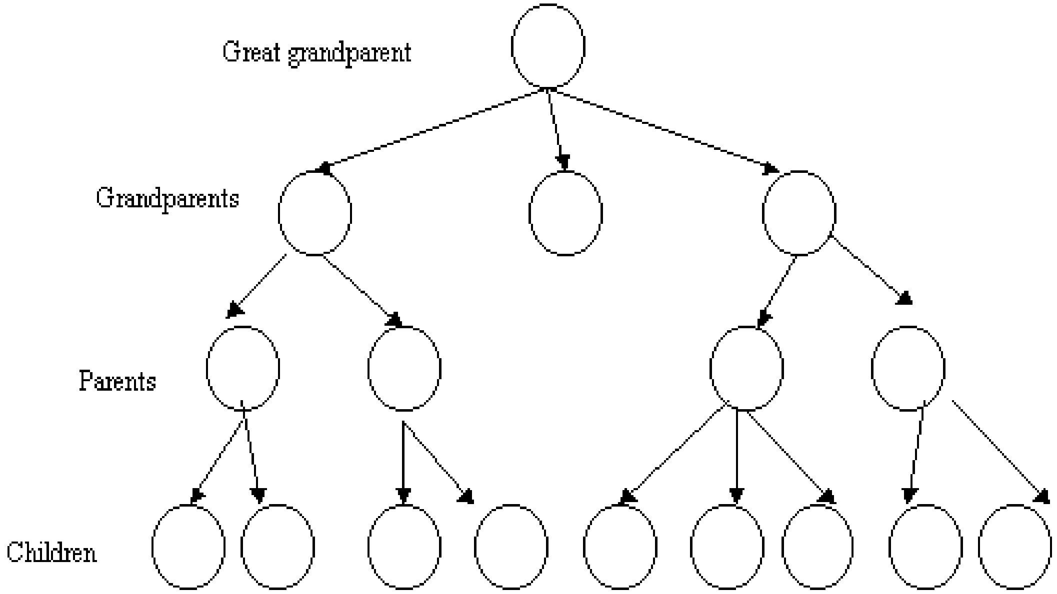
Hierarchical Model

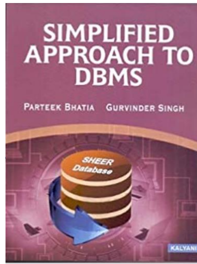
- Hierarchical Database model is one of the oldest database models, dating from late 1950s.
- One of the first hierarchical databases Information Management System (IMS).
- This model is based on tree data structure.



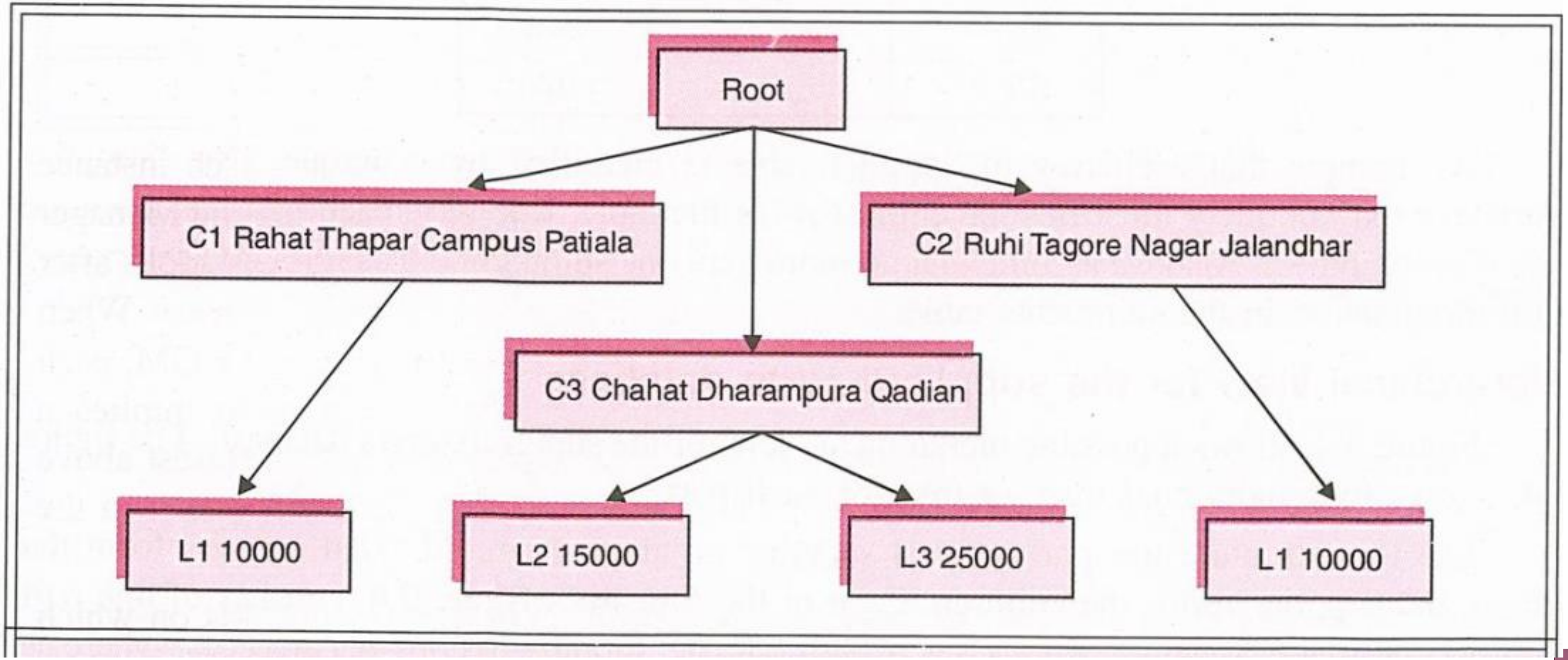
Hierarchical Model: An Example

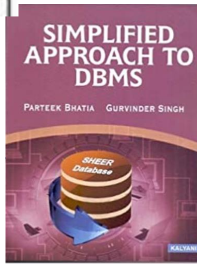






Hierarchical Model: An Example





Hierarchical Model: Supplier-Part Database

The Supplier records

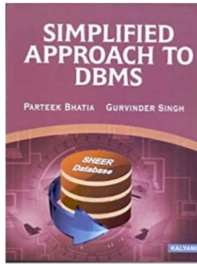
Sno	Name	Status	City
S1	Suneet	20	Qadian
S2	Ankit	10	Amritsar
S3	Amit	10	Amritsar

The Part records

Pno	Name	Color	Weight	City
P1	Nut	Red	12	Qadian
P2	Bolt	Green	17	Amritsar
P3	Screw	Blue	17	Jalandhar
P4	Screw	Red	14	Qadian

The Shipment records

Sno	Pno	Qty
S1	P1	250
S1	P2	300
S1	P3	500
S2	P1	250
S2	P2	500
S3	P2	300



Hierarchical Model: Supplier-Part Database

The Supplier records

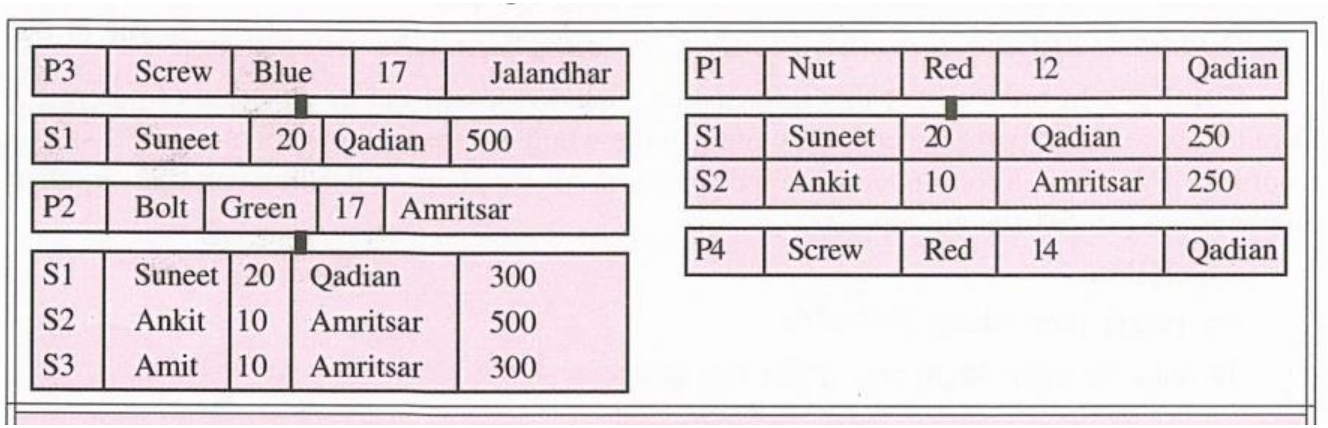
Sno	Name	Status	City
S1	Suneet	20	Qadian
S2	Ankit	10	Amritsar
S3	Amit	10	Amritsar

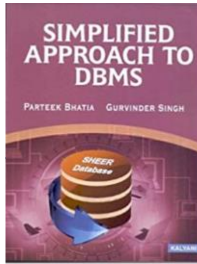
The Part records

Pno	Name	Color	Weight	City
P1	Nut	Red	12	Qadian
P2	Bolt	Green	17	Amritsar
P3	Screw	Blue	17	Jalandhar
P4	Screw	Red	14	Qadian

The Shipment records

Sno	Pno	Qty
S1	P1	250
S1	P2	300
S1	P3	500
S2	P1	250
S2	P2	500
S3	P2	300





Operations at Hierarchical Model

P3	Screw	Blue	17	Jalandhar

S1	Suneet	20	Qadian	500

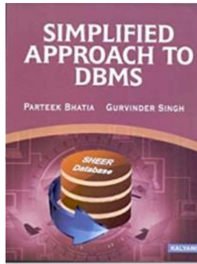
P2	Bolt	Green	17	Amritsar

S1	Suneet	20	Qadian	300
S2	Ankit	10	Amritsar	500
S3	Amit	10	Amritsar	300

P1	Nut	Red	12	Qadian

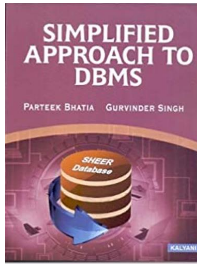
S1	Suneet	20	Qadian	250
S2	Ankit	10	Amritsar	250

P4	Screw	Red	14	Qadian



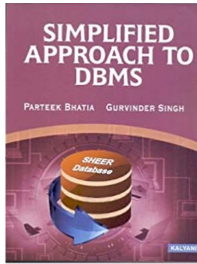
Insert Operation

- Cannot insert data of child without parent.



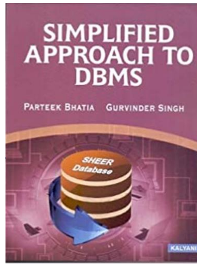
Update Operation

- The data of child record requires multiple update operations as their data is repeated.



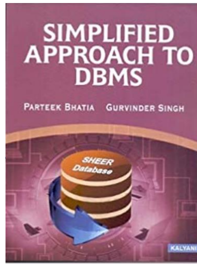
Delete Operations

- Parent removal causes deletion of child data.

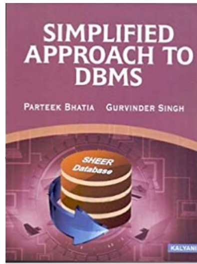


Retrieval Operations are Asymmetric

- **Query1:** Find part numbers for parts supplied by supplier S2.
- **Algorithm**
- do until no more parts;
-     get next part;
-     get [next] supplier under this part where SNO=S2;
-     if found then print PNO;
- end;

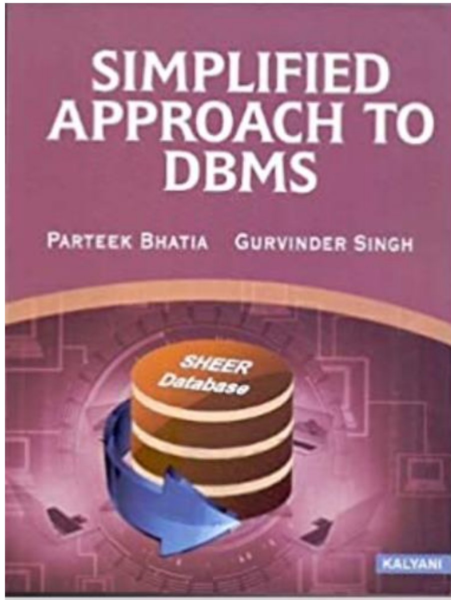


- **Query1:** Find supplier numbers who supply part P2.
- **Algorithm**
- get [next] part where PNO=P2;
- do until no more shipments under this part;
-     get next supplier under this part;
-     print SNO;
- end;
-



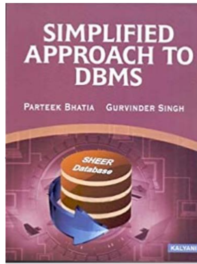
Network Data Model

By
Parteek Bhatia
Associate Professor
Department of Computer Science & Engineering
Thapar Institute of Engineering and Technology
Patiala



Network Model

- The Network model replaces the hierarchical tree with a graph thus allowing more general connections among the nodes.
- The main difference of the network model from the hierarchical model, is its ability to handle many to many (M:M) relations.
- A network structure allows 1:1 (one:one), 1:M (one:many), M:1 (many: one), M:M (many:many) relationships among entities.
- The network model was evolved to specifically handle non-hierarchical relationships.



Network Model: Supplier-Part Database

The Supplier records

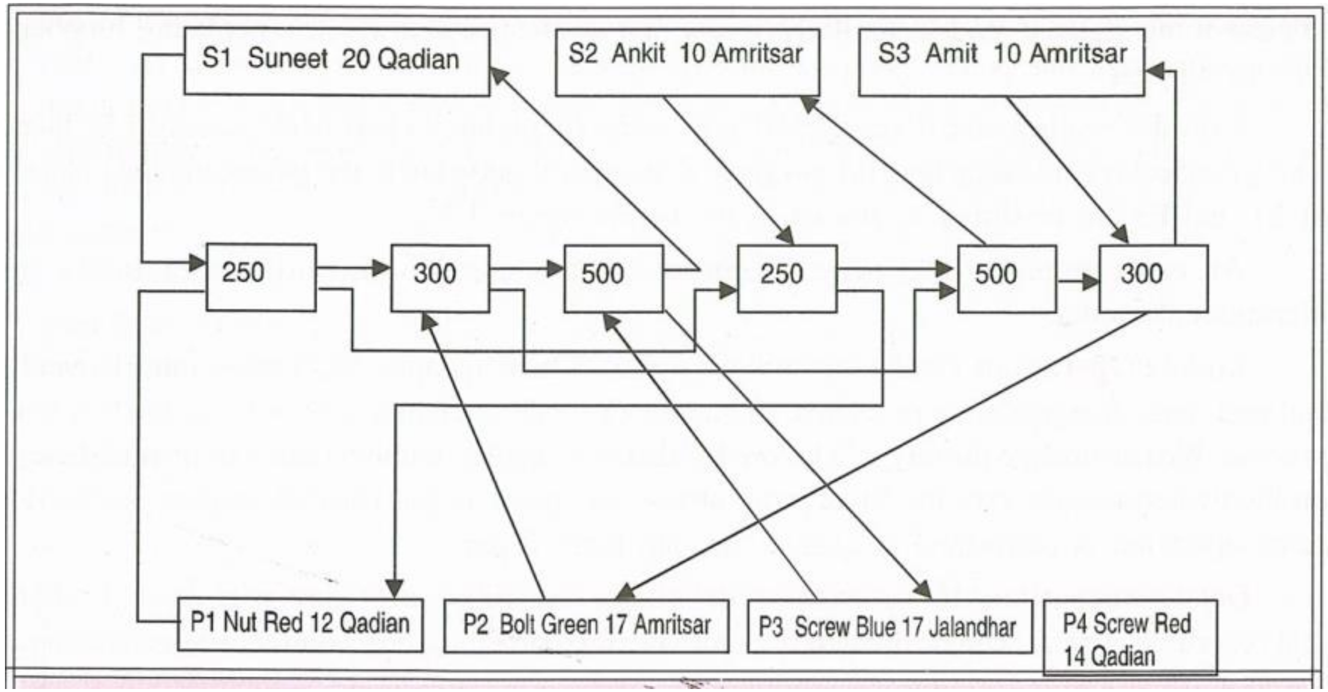
Sno	Name	Status	City
S1	Suneet	20	Qadian
S2	Ankit	10	Amritsar
S3	Amit	10	Amritsar

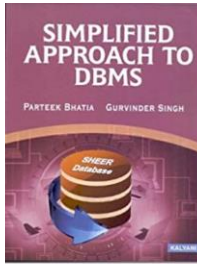
The Part records

Pno	Name	Color	Weight	City
P1	Nut	Red	12	Qadian
P2	Bolt	Green	17	Amritsar
P3	Screw	Blue	17	Jalandhar
P4	Screw	Red	14	Qadian

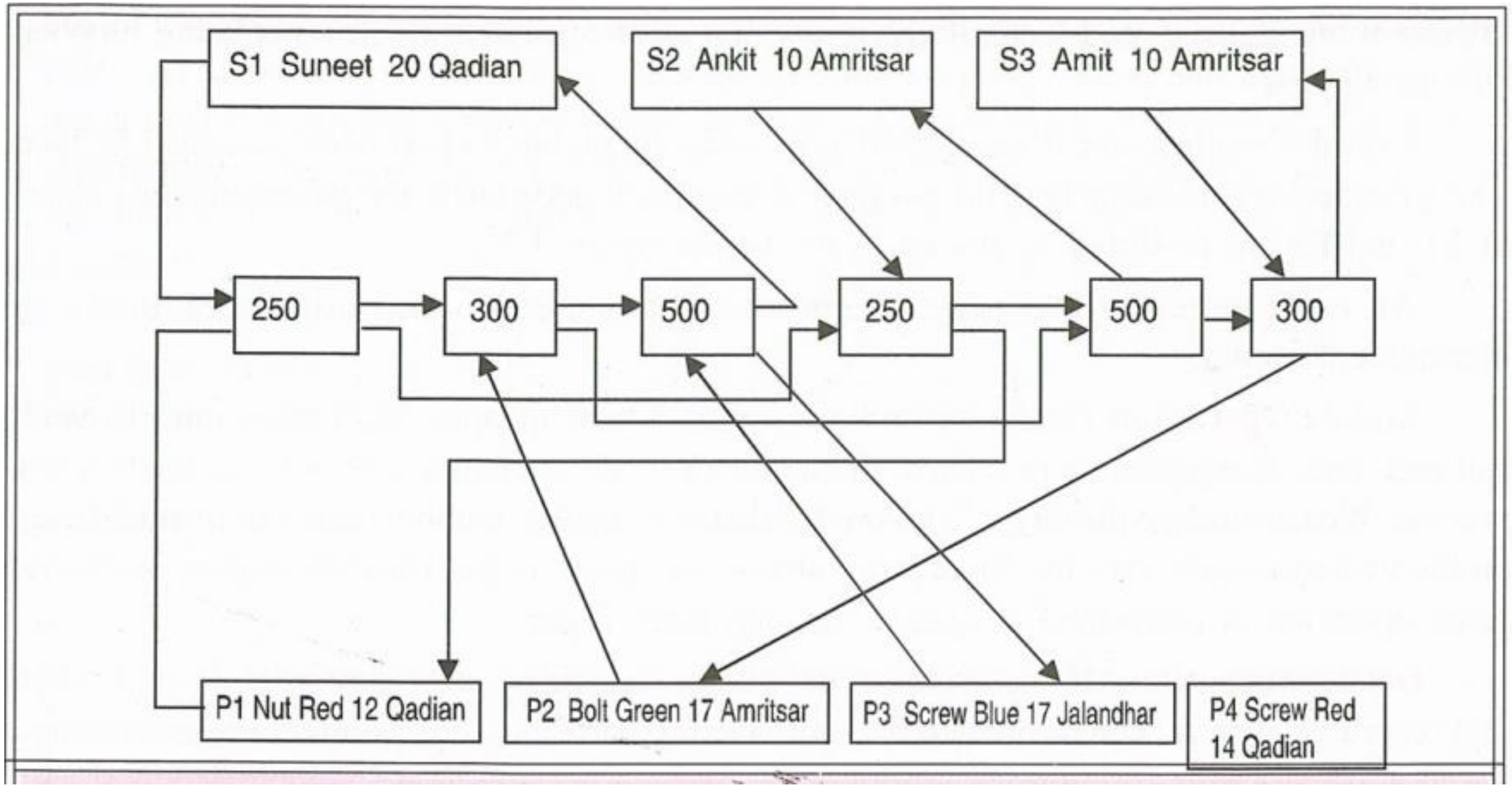
The Shipment records

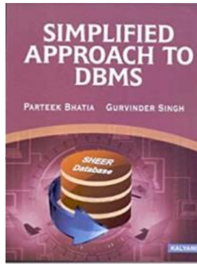
Sno	Pno	Qty
S1	P1	250
S1	P2	300
S1	P3	500
S2	P1	250
S2	P2	500
S3	P2	300



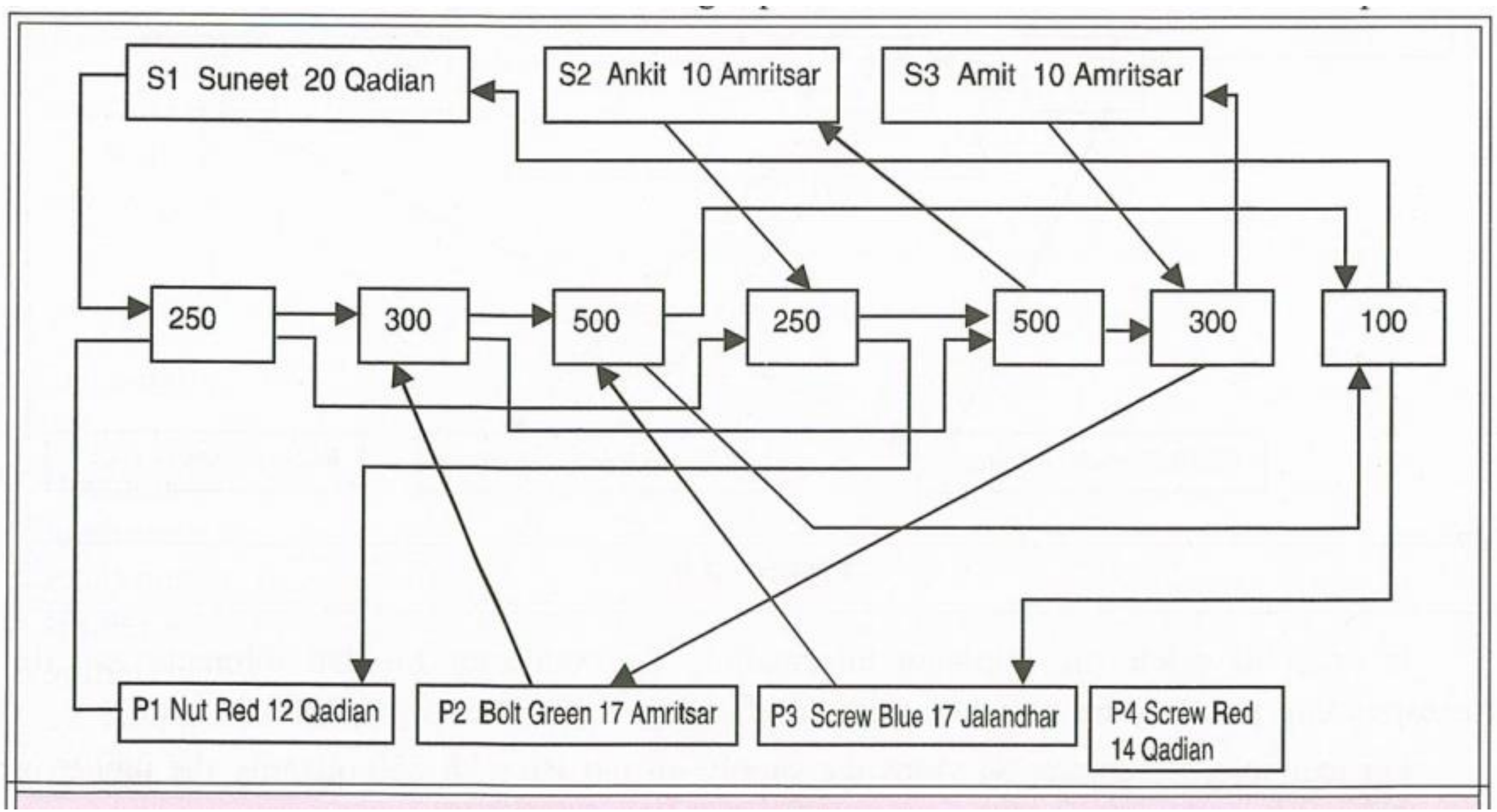


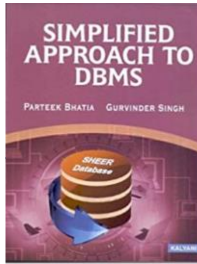
Network Model: Supplier-Part Database



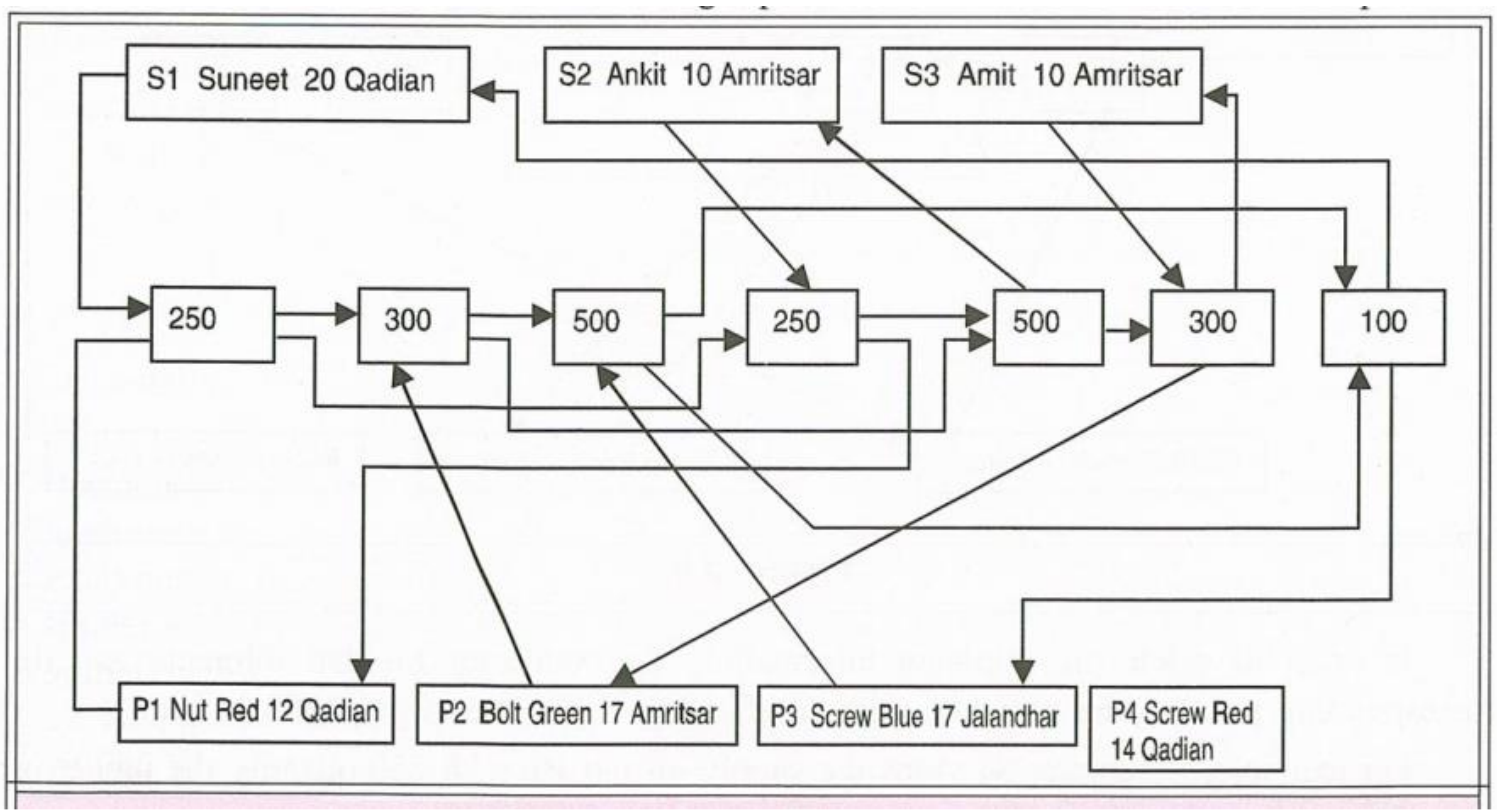


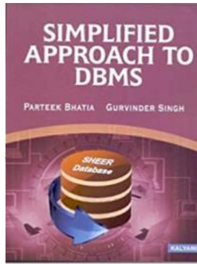
Insert Operation



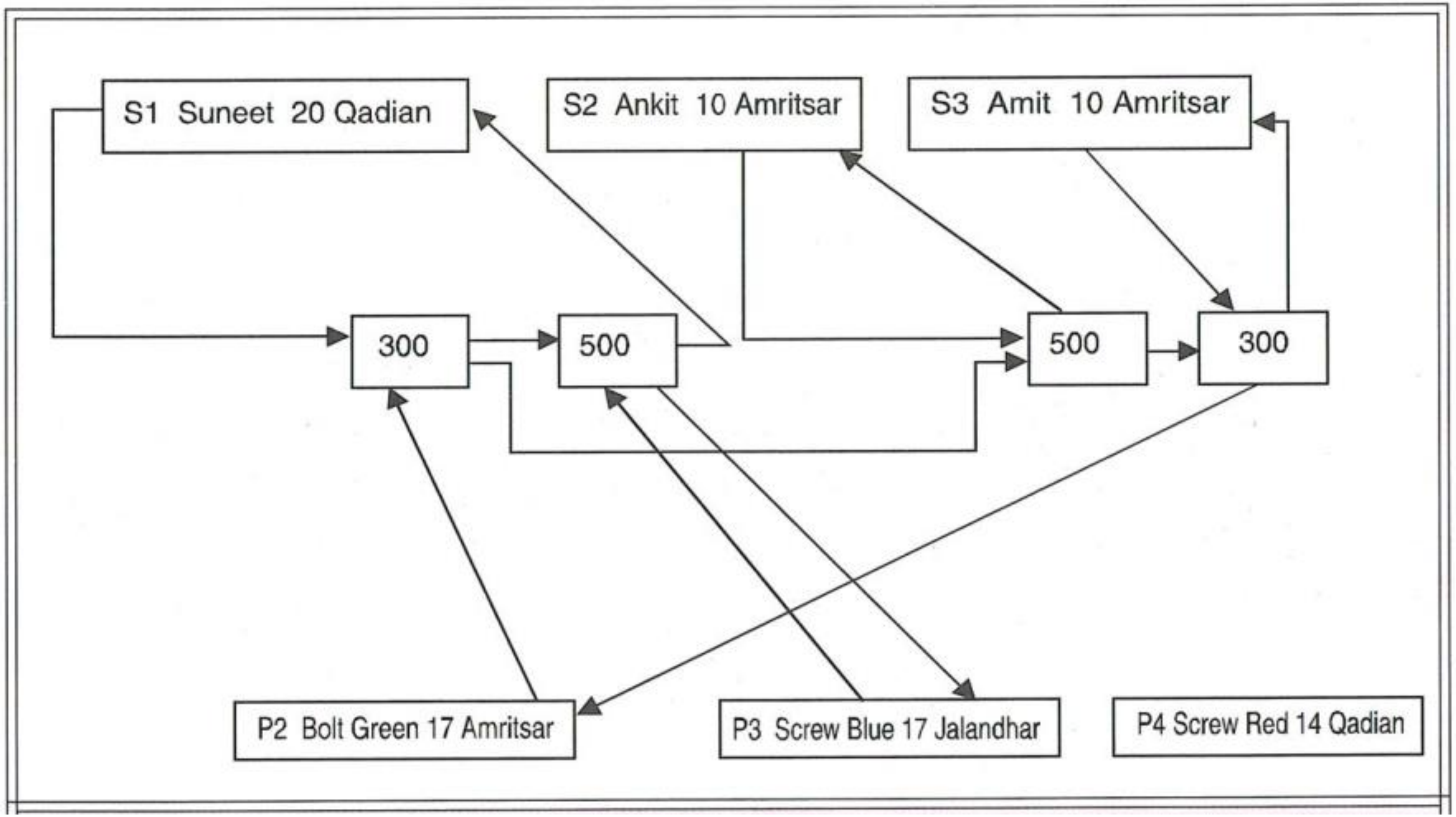


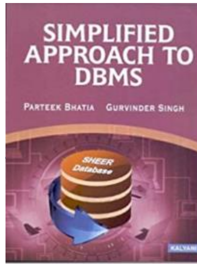
Update Operation



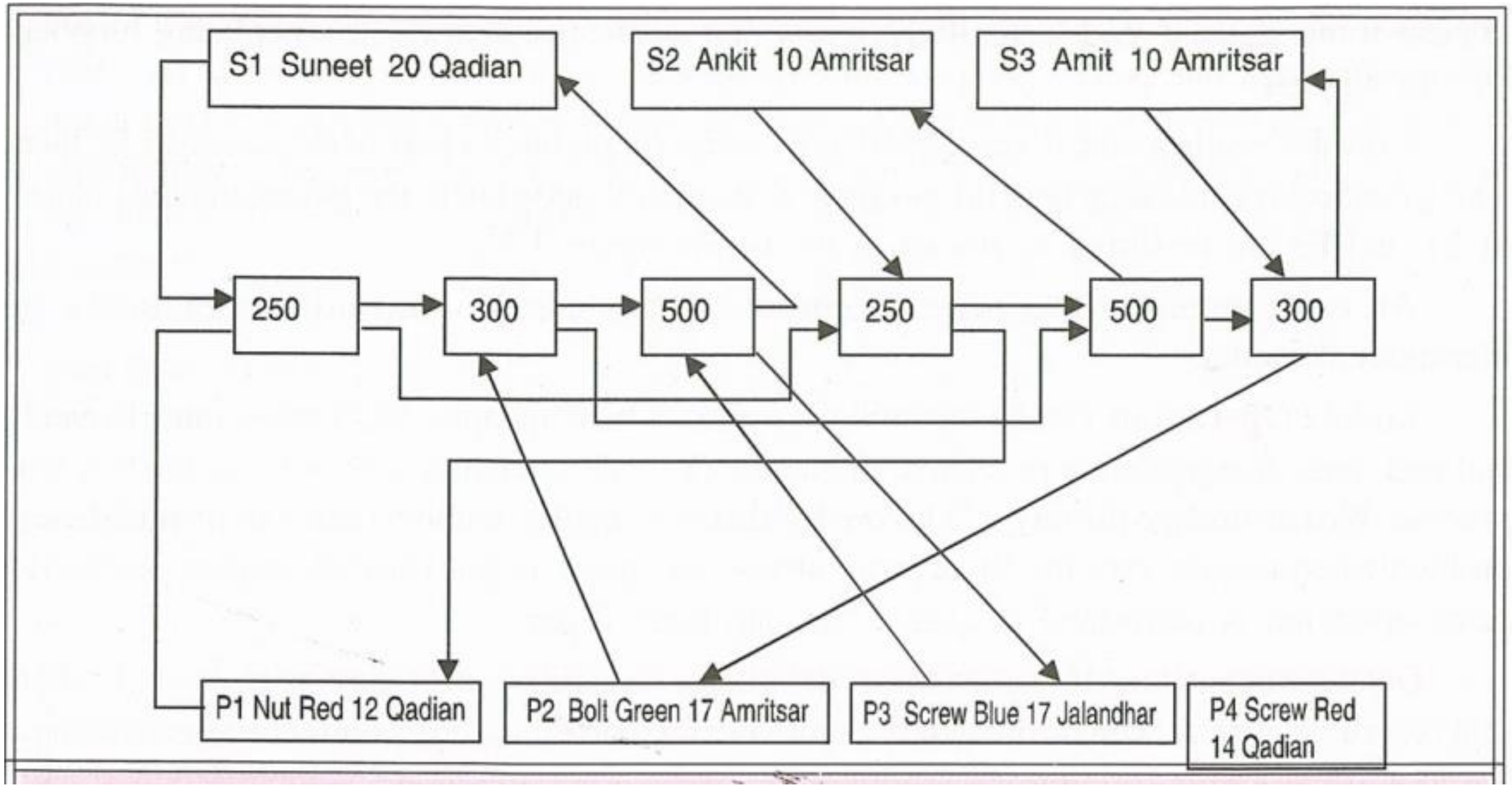


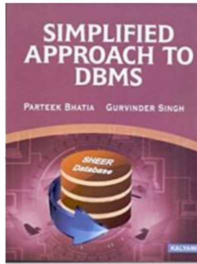
Delete Operation





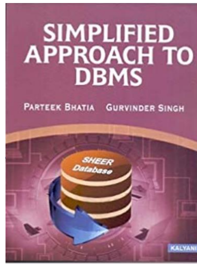
Retrieve Operation





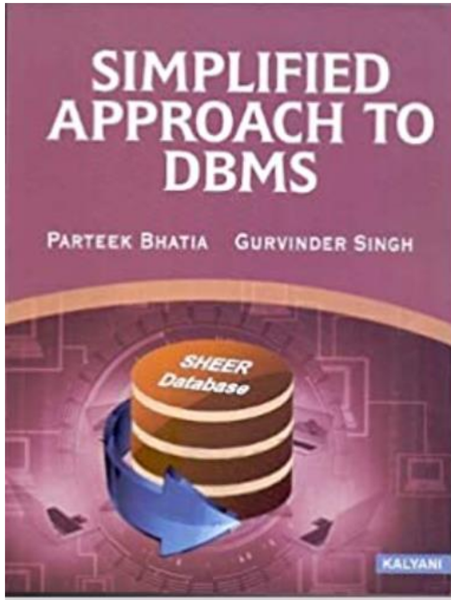
Operations over Network Model

- Insert
  - There is no anomaly.
- Update
  - There is no anomaly.
- Delete
  - There is no anomaly.
- Retrieve
  - There is no anomaly, retrieval operations are symmetric.
- Limitation of Network Model
  - The only limitation of network model is its complexity.



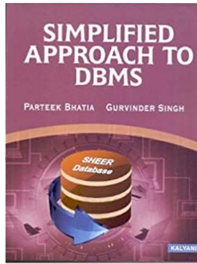
Relational Data Model

By
Parteek Bhatia
Associate Professor
Department of Computer Science & Engineering
Thapar Institute of Engineering and Technology
Patiala



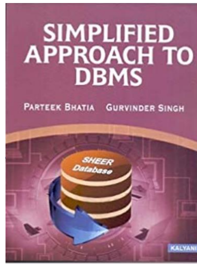
Relational Model

- Relational model stores data in the form of tables. This concept purposed by Dr. E.F. Codd, a researcher of IBM in the year 1960s. The relational model consists of three major components:
- A relational model database is defined as a database that allows you to group its data items into one or more independent tables that can be related to one another by using fields common to each related table.



Relational Model: An Example

Attributes	Emp_Code	Name	Year
Tuples	21130	Amar Jain	1
	30143	Kuldeep	3
	41894	Manoj	2
	51207	Rita Bajaj	6



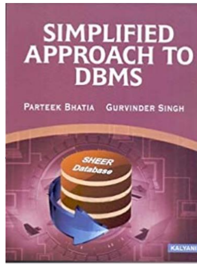
Relational Model: Customer Loan Database

Customer Table

CNO	NAME	ADDRESS	CITY
C1	Rahat	Thapar Campus	Patiala
C2	Ruhi	Tagore Nagar	Jalandhar
C3	Chahat	Dharampura	Qadian
C4	Pooja	GNDU	Amritsar

Customer_Loan Table

CNO	LNO	AMOUNT
C1	L1	10000
C2	L1	10000
C3	L2	15000
C3	L3	25000
C4	L4	35000



Relational Model: Supplier Part Database

The Supplier records

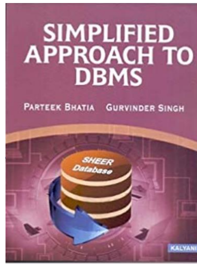
Sno	Name	Status	City
S1	Suneet	20	Qadian
S2	Ankit	10	Amritsar
S3	Amit	10	Amritsar

The Part records

Pno	Name	Color	Weight	City
P1	Nut	Red	12	Qadian
P2	Bolt	Green	17	Amritsar
P3	Screw	Blue	17	Jalandhar
P4	Screw	Red	14	Qadian

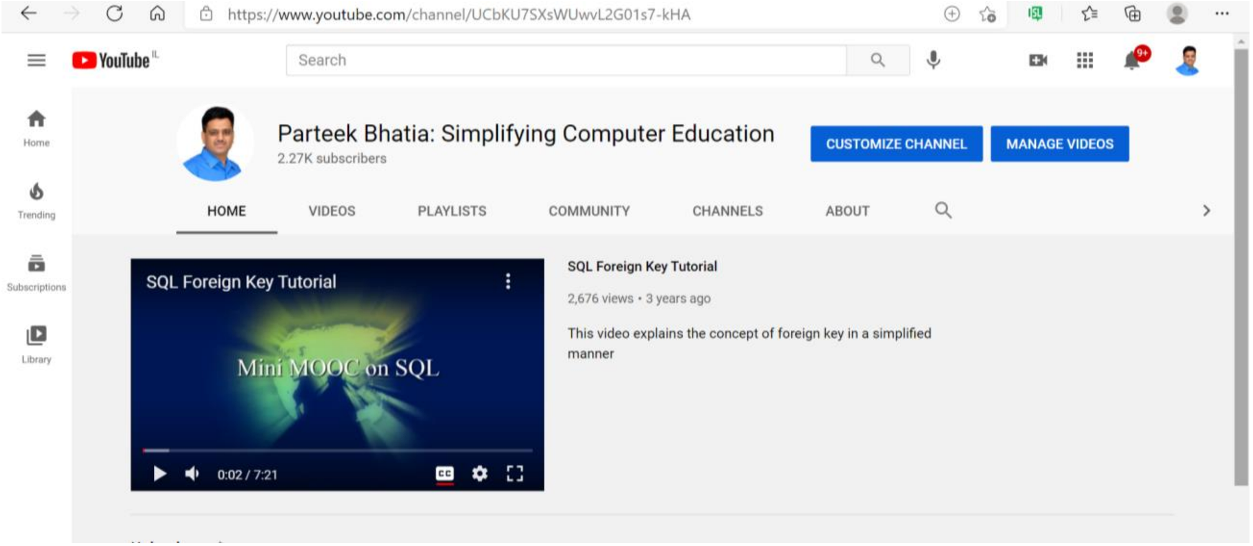
The Shipment records

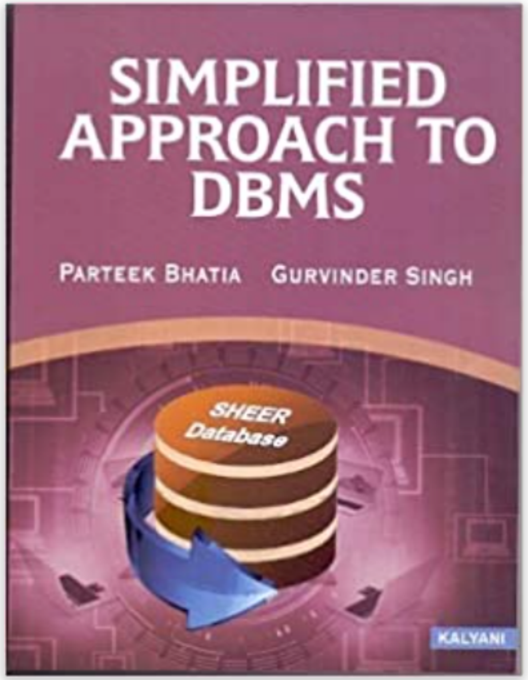
Sno	Pno	Qty
S1	P1	250
S1	P2	300
S1	P3	500
S2	P1	250
S2	P2	500
S3	P2	300



For more information

- Subscribe to YouTube Channel from the Author
  - To receive latest video tutorials on Data Mining, Machine Learning, DBMS, Big Data, NoSQL and many more.
- https://www.youtube.com/user/parteekbhatia





See this image

Simplified Approach to DBMS Unbound – 1 January 2019

by Gurvinder Singh Bhatia Parteek (Author)

2 ratings

See all formats and editions

Unbound
₹ 400.00

3 New from ₹ 400.00

Delivery By: **Feb 8 - 9** Details

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

₹ 400.00
M.R.P.: ~~₹ 450.00~~
You Save: ₹ 50.00 (11%)
Inclusive of all taxes

FREE Delivery. Details

In stock.
Sold by Lyall Book Depot..
Quantity: 1

Add to Cart
Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

Share

Some of online Video Courses created by Author over Udemy Platform

Simplified Approach to DBMS | U

udemy.com/course/simplified-approach-to-dbms/

Apps | Life+Style (Chd), LS... | InformIT: Safari Boo... | ICEET 2014 | Fortune Hotels - M... | Book Lowest Price I... | Captive Portal | BELIEVE in, INVEST i... | Machine Learning f...

Anytime, anywhere | Learn on your schedule from any device

Udemy | Categories | Search for anything | Udemy for Business | Teach on Udemy | Log in | Sign up

Development > Database Design & Development > Database Management

Simplified Approach to DBMS

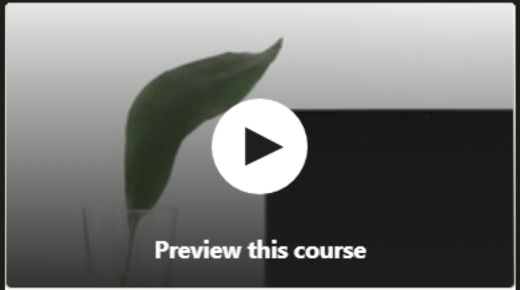
Simplified way to learn DBMS

4.3 ★★★★☆ (2 ratings) 14 students

Created by Parteek Bhatia

Last updated 11/2020 English

Wishlist | Share | Gift this course



Personal | Teams

₪79.90

Add to cart

Buy now

30-Day Money-Back Guarantee

This course includes:

- 9.5 hours on-demand video
- Full lifetime access

What you'll learn

- ✓ Understanding the meaning of term DBMS
- ✓ File Based System Vs. Data Base Management System
- ✓ Hierarchical, Network and Relational Data Models
- ✓ Concept of different types of keys like Super Key, Candidate Key, Primary Key and Foreign Key
- ✓ Designing of Entity Relation Model
- ✓ Conversion of ER Model to Tables
- ✓ SQL: Creation of Tables
- ✓ SQL: Insertion, Updation, Deletion and Retrieval of Data

Type here to search | ENG 00:25 04-02-2021

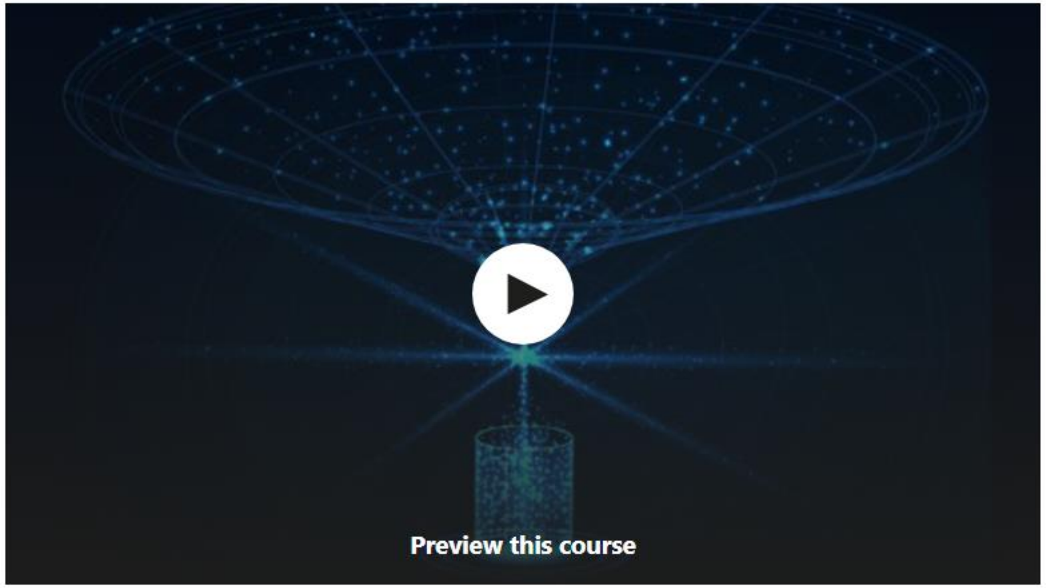
Online Course on SQL at Udemy

https://www.udemy.com/course/learn-sql-in-a-simplified-manner/

Anytime, anywhere | Learn on your schedule from any device

Udemy   Categories   Search for anything   Udemy for Business   Instructor   My learning

Development > Programming Languages > SQL



Learn SQL in a simplified manner

Learn SQL in just four weeks, right from scratch !

4.1 ★★★★☆ (467 ratings) 12,875 students   1hr 56min of on-demand video

Created by Parteek Bhatia

English   English [Auto]

Free

Enroll now

What you'll learn   Course content   Reviews   Instructors

✓ SQL is one of the most in demand tech skills today, and learning SQL would surely be one of the fastest and most efficient way to improve ones career prospectus. After the completion of course, one will be able to read, write and comprehend simple to most complex queries of SQL, using which they could work upon any SQL database such as Oracle, Amazon Redshift and others.

Type here to search   ENG   00:19 04-02-2021

Mastering PL/SQL | Udemy

udemy.com/course/mastering-plsql/

Apps | Life+Style (Chd), LS... | InformIT: Safari Boo... | ICEET 2014 | Fortune Hotels - M... | Book Lowest Price I... | Captive Portal | BELIEVE in, INVEST i... | Machine Learning f...

Udemy Categories Search for anything Udemy for Business Teach on Udemy Log in Sign up

Teaching & Academics > Online Education > Pl/SQL

Mastering PL/SQL

Simplified way to learn PL/SQL

Highest Rated 4.7 ★★★★☆ (17 ratings) 110 students

Created by Parteek Bhatia

Last updated 5/2020 English English [Auto]

Wishlist Share Gift this course

Personal Teams

₪79.90

Add to cart

Buy now

30-Day Money-Back Guarantee

This course includes:

9.5 hours on-demand video

What you'll learn

- Basics of PL/SQL
- Variables and Constants
- Conditional, Looping and Sequential Statements
- Exception Handling
- Cursor Handling
- Sub-Programs, Functions and Procedures
- Triggers
- DBMS:: Get chance to enroll in Simplified Approach to DBMS Course completely free

Waiting for www.udemy.com...

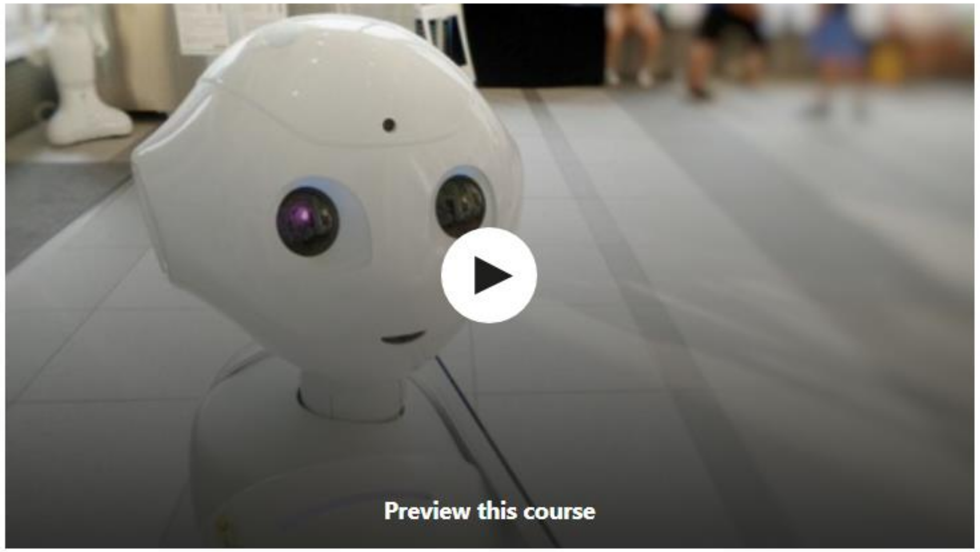
Type here to search ENG 00:23 04-02-2021

https://www.udemy.com/course/mastering-machine-learning-course-1/

Anytime, anywhere | Learn on your schedule from any device

Udemy   Categories   Search for anything   Udemy for Business   Instructor   My learning

Development > Data Science > Machine Learning



Mastering Machine Learning: Course-1

Taking First Step Towards Machine Learning

3.9 ★★★★☆ (21 ratings) 3,425 students   1hr 52min of on-demand video

Created by Parteek Bhatia

English   English [Auto]

Free

Enroll now

What you'll learn   Course content   Reviews   Instructors

- ✓ Machine Learning
- ✓ Python
- ✓ Regression
- ✓ Classification
- ✓ Unsupervised Machine Learning

Type here to search   ENG   00:20   04-02-2021

https://www.udemy.com/course/genetic-algorithm-for-machine-learning/

IT & Software > Other IT & Software > Genetic Algorithm

Genetic Algorithm for Machine Learning

Simplified Way to Learn

New 0.0 ☆☆☆☆☆ (0 ratings) 447 students 1hr 32min of on-demand video

Created by Parteek Bhatia

English

Free

Enroll now

What you'll learn Course content Reviews Instructors

- ✓ Working Principle of Genetics Algorithms
- ✓ Natural Selection
- ✓ Implementation of Natural Selection through Roulette Wheel
- ✓ Crossover or Recombination
- ✓ Concept of Probability of Crossover and Its usage in generation of Population

Touchpad

https://www.udemy.com/course/association-mining-for-machine-learning/

Anytime, anywhere | Learn on your schedule from any device

Udemy Categories Search for anything Udemy for Business Instructor My learning

IT & Software > Other IT & Software > Machine Learning

Association Mining for Machine Learning

Simplified Way to Learn

New 4.6 ★★★★☆ (13 ratings) 629 students 1hr 48min of on-demand video

Created by Parteek Bhatia

English English [Auto]

Free

Enroll now

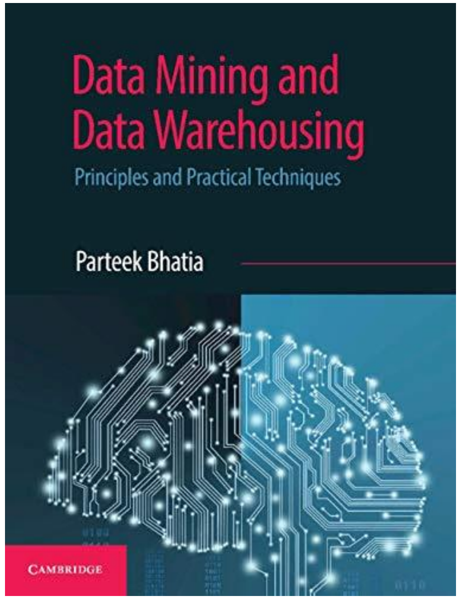
What you'll learn Course content Reviews Instructors

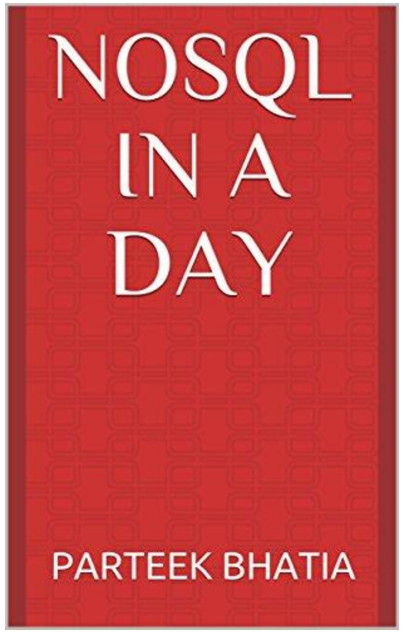
- ✓ Association Mining
- ✓ Naive Algorithm
- ✓ Apriori Algoorithm

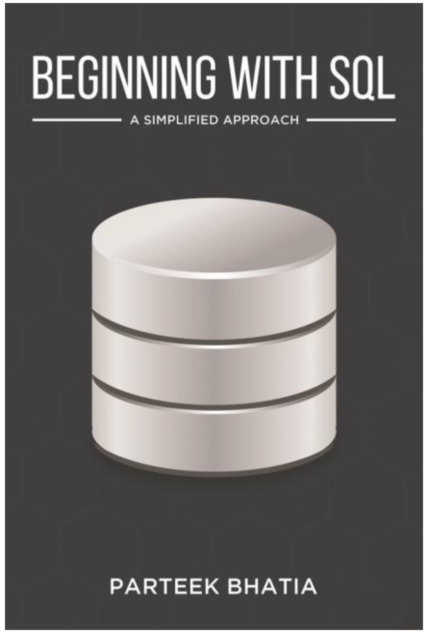
Windows Ink Workspace

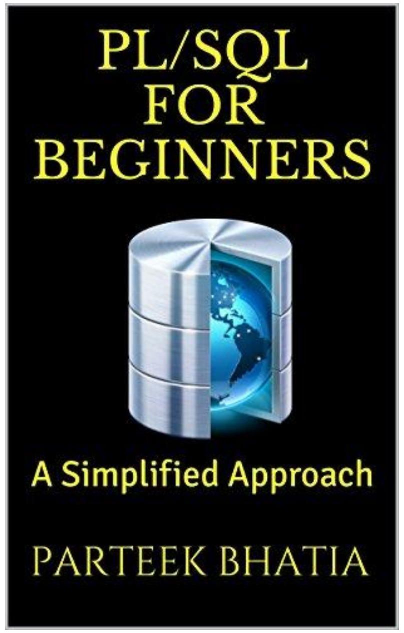
Type here to search ENG 00:20 04-02-2021

Books from the Same Author









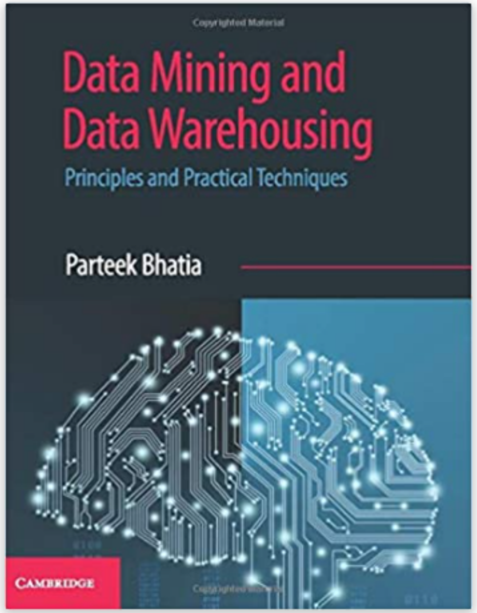
For more information visit: www.parteekbhatia.com

Coming Soon…

- **Text Book**
- **Machine Learning: Principles and Practical Techniques**

Subtotal ₹ 3,695.00

‹ Back to results



See all 2 images

Data Mining and Data Warehousing: Principles and Practical Techniques

Paperback – 27 June 2019

by Parteek Bhatia (Author)

★★★★★ 15 ratings

› See all formats and editions

Kindle Edition	Paperback
₹ 1,374.45	₹ 519.00 prime
Read with Our Free App	14 New from ₹ 499.00

FREE delivery: **Monday, Feb 8**

Order within 13 hrs and 59 mins Details

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

₹ 519.00

M.R.P.: ~~₹ 695.00~~

You Save: ₹ 176.00 (25%)

Inclusive of all taxes

prime

In stock.

Sold by Cloudtail India and Fulfilled by Amazon.

Quantity: 1

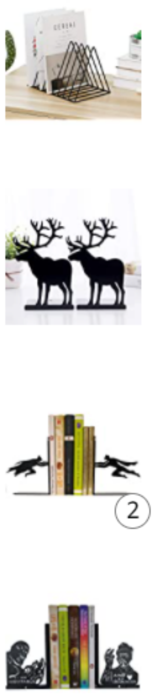
Add to Cart

Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

Share



https://www.amazon.in/PL-SQL-Beginners-Simplified-Approach-ebook/dp/B01JIUZC64/ref=sr_1_6?...

Deliver to Parteek Patiala 147004 | Books | parteek bhatia | Hello, Parteek Account & Lists | Returns & Orders | 5 Cart

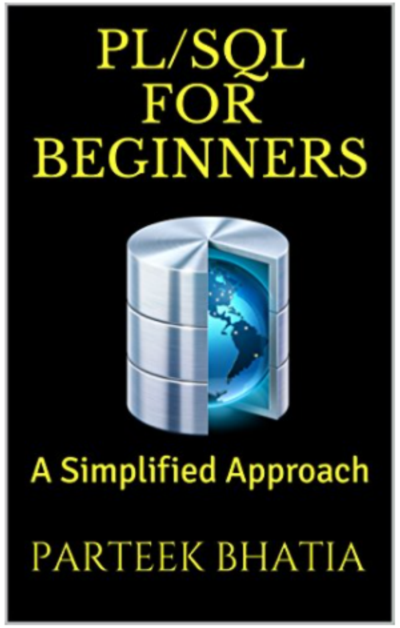
All | Prime | Parteek's Amazon.in | Amazon Pay | Mobiles | Best Sellers | Today's Deals | You have an unused Prime benefit *Redirects to music.amazon.in

Books | Advanced Search | New Releases & Pre-orders | Best Sellers | Browse Genres | Children's & Young Adult | Textbooks | Exam Central | All Indian Languages

‹ Back to results

You purchased this item on 4 August, 2016.
View this order | You can find this title on your Amazon Kindle apps and devices, or the Manage Your Content and Devices page. Need additional help? Click here

Look inside



PL/SQL for Beginners: A Simplified Approach Kindle Edition

by Parteek Bhatia (Author) | Format: Kindle Edition

5 ratings

› See all formats and editions

Kindle Edition
₹ 0.00 kindleunlimited

This title and over 1 million more available with Kindle Unlimited
₹ 99.00 to buy

PL/SQL: A Simplified Approach
Why read this book?
This book covers the basics of PL/SQL with lots of real world examples.
This book will help to gain better skills, better placement and better future.
...plified manner.

Kindle Price: ₹ 99.00
inclusive of all taxes

includes free wireless delivery via ***Amazon Whispernet***

Sold by: Amazon Asia-Pacific Holdings Private Limited

kindleunlimited
Included with your membership.

Read Now

You own this item. It's available on your Kindle apps and devices.
Enter a promotion code or Gift Card

Buy for others
Give as a gift or purchase for a team or group. Learn more

Subtotal: ₹ 3,695.00

There are 17 recent changes in your Cart.

Proceed to checkout

This order contains a gift

₹ 499.00

₹ 499.00

₹ 899.00 (x2)

Saved for later (120)

Waiting for fls-eu.amazon.in...

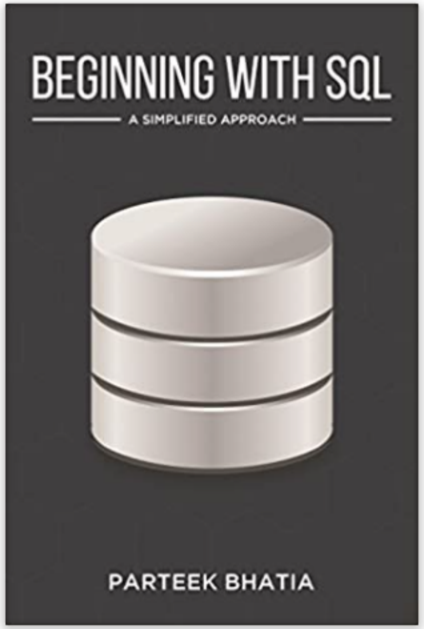
Type here to search | ENG | 00:46 04-02-2021 https://www.amazon.in/Beginning-English-Author-Parteek-Bhatia/dp/B07D8QNCMV/ref=sr_1_5?dc...

amazon.in prime | Deliver to Parteek Patiala 147004 | Books | parteek bhatia | Hello, Parteek Account & Lists | Returns & Orders | 5 Cart

All | Prime | Parteek's Amazon.in | Amazon Pay | Mobiles | Best Sellers | Today's Deals | WATCH NOW prime video *Redirects to PrimeVideo.com

Books | Advanced Search | New Releases & Pre-orders | Best Sellers | Browse Genres | Children's & Young Adult | Textbooks | Exam Central | All Indian Languages

‹ Back to results



See this image

Beginning with SQL English Paperback – 10 May 2017

by Author Parteek Bhatia (Author)

› See all formats and editions

Paperback
₹ 215.00

1 New from ₹ 215.00

Delivery by: **Feb 12 - 16**

Save Extra with 4 offers

Cashback (2): 5% back with Amazon Pay ICICI Bank Credit card for Prime-members. 3% back for everybody else | See All

No Cost EMI: Avail No Cost EMI on select cards for orders above ₹3000 | Details

See 2 more

₹ 215.00
Inclusive of all taxes
+ ₹ 60.00 Delivery charge

In stock.

Sold and fulfilled by Educreation Publishing.

Quantity: 1

Add to Cart

Buy Now

Deliver to Parteek - Patiala 147004

Add to Wish List

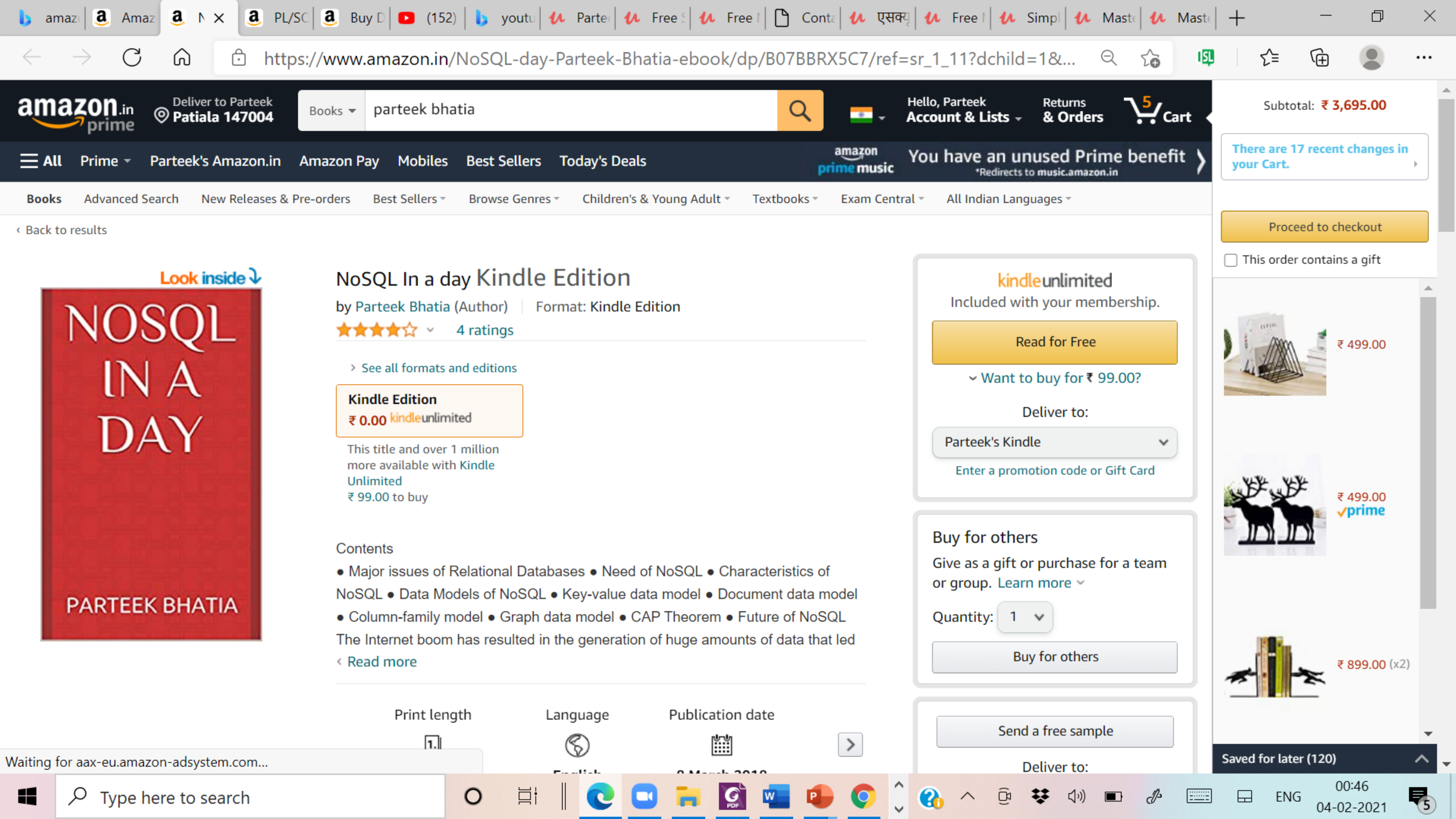
Share

Have one to sell?

Sell on Amazon

Waiting for aax-eu.amazon-adsystem.com...

Type here to search | ENG | 00:47 04-02-2021


https://www.amazon.in/NoSQL-day-Parteek-Bhatia-ebook/dp/B07BBRX5C7/ref=sr_1_11?dchild=1&...
amazon.in prime
Deliver to Parteek
Patiala 147004
Books
parteek bhatia
Hello, Parteek
Account & Lists
Returns
& Orders
5
Cart
All
Prime
Parteek's Amazon.in
Amazon Pay
Mobiles
Best Sellers
Today's Deals
amazon prime music
You have an unused Prime benefit
*Redirects to music.amazon.in
Books
Advanced Search
New Releases & Pre-orders
Best Sellers
Browse Genres
Children's & Young Adult
Textbooks
Exam Central
All Indian Languages
Back to results
Look inside
NOSQL IN A DAY
PARTEEK BHATIA
NoSQL In a day Kindle Edition
by Parteek Bhatia (Author) Format: Kindle Edition
4 ratings
See all formats and editions
Kindle Edition
₹ 0.00 kindleunlimited
This title and over 1 million more available with Kindle Unlimited
₹ 99.00 to buy
Contents
● Major issues of Relational Databases ● Need of NoSQL ● Characteristics of NoSQL ● Data Models of NoSQL ● Key-value data model ● Document data model ● Column-family model ● Graph data model ● CAP Theorem ● Future of NoSQL
The Internet boom has resulted in the generation of huge amounts of data that led
Read more
Print length
Language
Publication date
kindleunlimited
Included with your membership.
Read for Free
Want to buy for ₹ 99.00?
Deliver to:
Parteek's Kindle
Enter a promotion code or Gift Card
Buy for others
Give as a gift or purchase for a team or group. Learn more
Quantity: 1
Buy for others
Send a free sample
Deliver to:
Subtotal: ₹ 3,695.00
There are 17 recent changes in your Cart.
Proceed to checkout
This order contains a gift
₹ 499.00
₹ 499.00
prime
₹ 899.00 (x2)
Saved for later (120)
Waiting for aax-eu.amazon-adsystem.com...
Type here to search
ENG
00:46
04-02-2021

For more information

- Please visit: www.parteekbhatia.com
- For Feedback: parteek.bhatia@gmail.com
- For online Video Courses: https://www.udemy.com/user/parteek-kumar-4/
- For YouTube Channel:
- https://www.youtube.com/user/parteekbhatia
- For Books: Amazon.in : Parteek Bhatia
- For Workshops and invited talks, please write to:

parteek.bhatia@gmail.com

Thanks

Keys

Keys?

- A DBMS key is an attribute or set of an attribute which helps you to identify a row(tuple) in a relation(table).

- They allow you to find the relation between two tables.

- Keys help you uniquely identify a row in a table by a combination of one or more columns in that table.

Why we need a Key?

- Keys help you to identify any row of data in a table.
- In a real-world application, a table could contain thousands of records. Moreover, the records could be duplicated.
- Keys ensure that you can uniquely identify a table record despite these challenges.
- Allows you to establish a relationship between and identify the relation between tables Help you to enforce identity and integrity in the relationship.

Keys in Database Management System

- Super Key
- Primary Key
- Candidate Key
- Alternate Key
- Foreign Key
- Composite Key
- Surrogate Key/ Artificial Key

Super key?

- A superkey is a group of single or multiple keys which identifies rows in a table.

- A Super key may have additional attributes that are not needed for unique identification.

In this example, EmpSSN, EmpNum and Empname are superkeys.

Example:

EmpSSN	EmpNum	Empname
9812345098	AB05	Shown
9876512345	AB06	Roslyn
199937890	AB07	James

Primary Key?

- A column or group of columns in a table which helps us to uniquely identifies every row in that table is called a primary key.

- This can't be a duplicate. The same value can't appear more than once in the table.

- Rules for defining Primary key:

- Two rows can't have the same primary key value.
- It must for every row to have a primary key value.
- The primary key field cannot be null.
- The value in a primary key column can carefully be modified or updated, considering that if any foreign key refers to that primary key.

Example:

In the following example, StudID is a Primary Key.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com (mailto:abc@gmail.com)
2	12	Nick	Wright	xyz@gmail.com (mailto:xyz@gmail.com)
3	13	Dana	Natan	mno@yahoo.com (mailto:mno@yahoo.com)

Alternate key?

- All the keys which are not primary key are called an alternate key.
- It is a candidate key which is currently not the primary key.
- However, A table may have single or multiple choices for the primary key.

Example: In this table, **StudID, Roll No, Email** are qualified to become a primary key. But since **StudID is the primary key**, **Roll No, Email becomes the alternative key**.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com (mailto:abc@gmail.com)
2	12	Nick	Wright	xyz@gmail.com (mailto:xyz@gmail.com)
3	13	Dana	Natan	mno@yahoo.com (mailto:mno@yahoo.com)

Candidate Key?

- A super key with no repeated attribute is called candidate key.
- The Primary key should be selected from the candidate keys. Every table must have at least a single candidate key.

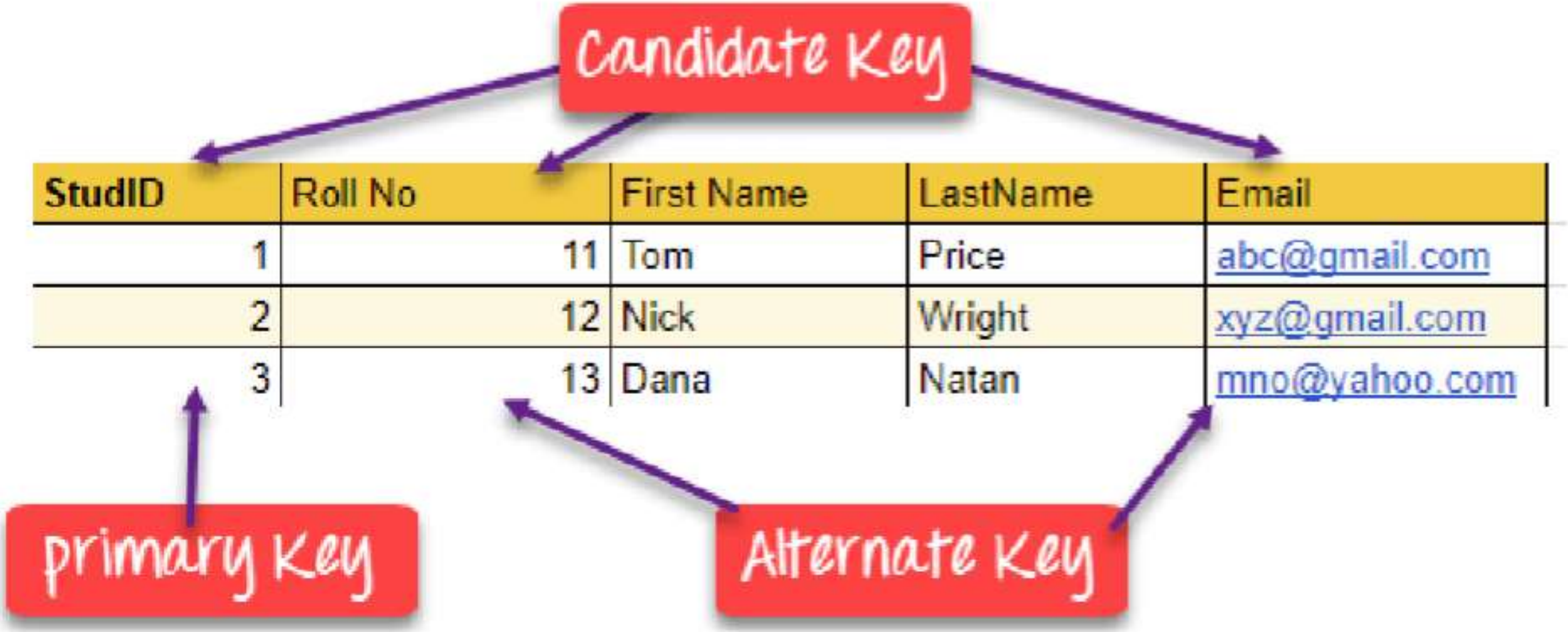
- Properties of Candidate key:

- It must contain unique values
- Candidate key may have multiple attributes
- Must not contain null values
- It should contain minimum fields to ensure uniqueness
- Uniquely identify each record in a table

Candidate Key?

- Example: In the given table **Stud ID, Roll No, and email** are candidate keys which help us to uniquely identify the student record in the table.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com (mailto:abc@gmail.com)
2	12	Nick	Wright	xyz@gmail.com (mailto:xyz@gmail.com)
3	13	Dana	Natan	mno@yahoo.com (mailto:mno@yahoo.com)



StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com
2	12	Nick	Wright	xyz@gmail.com
3	13	Dana	Natan	mno@yahoo.com

Foreign key?

- A foreign key is a column which is added to create a relationship with another table.
- Foreign keys help us to maintain data integrity and also allows navigation between two different instances of an entity.
- Every relationship in the model needs to be supported by a foreign key.

Example:

DeptCode	DeptName
001	Science
002	English
005	Computer

Teacher ID	Fname	Lname
B002	David	Warner
B017	Sara	Joseph
B009	Mike	Brunton

In this example, we have two table**, teacher and department** in a school. However, there is no way to see which search work in which department.

In this table, adding the foreign key in Deptcode to the Teacher, we can create a relationship between the two tables.

teacher and department tables

Example:

DeptCode	DeptName
001	Science
002	English
005	Computer

Teacher ID	Fname	Lname
B002	David	Warner
B017	Sara	Joseph
B009	Mike	Brunton

Teacher with DeptCode

Teacher ID	DeptCode	Fname	Lname
B002	002	David	Warner
B017	002	Sara	Joseph
B009	001	Mike	Brunton

This concept is also known as Referential Integrity.

Composite key?

- A key which has multiple attributes to uniquely identify rows in a table is called a composite key.

- It is type of Compound key has many fields which allow you to uniquely recognize a specific record.

- It is possible that each column may be not unique by itself within the database.

- However, when combined with the other column or columns the combination of composite keys become unique.

Composite or Compound key example

Example:

OrderNo	PorductID	Product Name	Quantity
B005	JAP102459	Mouse	5
B005	DKT321573	USB	10
B005	OMG446789	LCD Monitor	20
B004	DKT321573	USB	15
B002	OMG446789	Laser Printer	3

In this example, OrderNo and ProductID can't be a primary key as it does not uniquely identify a record. However, a compound key of Order ID and Product ID could be used as it uniquely identified each record.

Surrogate Key/Artificial key ?

- An artificial key which aims to uniquely identify each record is called a surrogate key.

- These kind of key are unique because they are created when you don't have any natural primary key.

- They do not lend any meaning to the data in the table.

- Surrogate key is usually an integer.

Student Table

Student_Name	Class
Amit	CSE
Amit	CSE
Rohit	ENC

Student Table with Sr_No as Surrogate Key/Artificial key ?

Sr_No	Student_Name	Class
1	Amit	CSE
2	Amit	CSE
3	Rohit	ENC

Difference Between Primary key & Foreign key

Primary Key	Foreign Key
Helps you to uniquely identify a record in the table.	It is a field in the table that is the primary key of another table.
Primary Key never accept null values.	A foreign key may accept multiple null values.
Primary key is a clustered index and data in the DBMS table are physically organized in the sequence of the clustered index.	A foreign key cannot automatically create an index, clustered or non-clustered. However, you can manually create an index on the foreign key.
You can have the single Primary key in a table.	You can have multiple foreign keys in a table.

Keys

- Let $K \subseteq R$
- *K* is a **superkey** of *R* if values for *K* are sufficient to identify a unique tuple of each possible relation *r(R)*
  - Example: {*ID*} and {ID,name} are both superkeys of *instructor*
- Superkey *K* is a **candidate key** if *K* is minimal
  - Example: {*ID*} is a candidate key for *Instructor*
- One of the candidate keys is selected to be the **primary key**
  - Which one?
- A **surrogate key** (or synthetic key) in a database is a unique identifier for either an *entity* in the modeled world or an *object* in the database
  - The surrogate key is *not* derived from application data, unlike a *natural* (or *business*) key which is derived from application data

Keys

- **Super Key**: Roll #, {Roll #, DoB}
- **Candidate Keys**: Roll #, {First Name, Last Name}, Passport #, Aadhaar #
  - Passport # cannot be a key. Why?
  - Null values are allowed for Passport # (a student may not have a passport)
- **Primary Key**: Roll #
- **Secondary / Alternate Key**: {First Name, Last Name}, Aadhaar #
- **Simple key**: Consists of a *single attribute*
- **Composite Key**: {First Name, Last Name}
  - Consists of more than one attribute to uniquely identify an entity occurrence
  - One or more of the attributes, which make up the key, are not simple keys in their own right

Roll #	First Name	Last Name	DoB	Passport #	Aadhaar #	Department
15CS10026	Lalit	Dubey	27-Mar-1997	L4032464	1728-6174-9239	Computer
16EE30029	Jatin	Chopra	17-Nov-1996	null	3917-1836-3816	Electrical
15EC10016	Smriti	Mongra	23-Dec-1996	G5432849	2045-9271-0914	Electronics
16CE10038	Dipti	Dutta	02-Feb-1997	null	5719-1948-2918	Civil
15CS30021	Ramdin	Minz	10-Jan-1997	X8811623	4928-4927-5924	Computer

Keys

- **Foreign key** constraint: Value in one relation must appear in another
  - **Referencing** relation
    - Enrolment: Foreign Keys – Roll #, Course #
  - **Referenced** relation
    - Students, Courses
- A **compound key** consists of *more than one attribute* to uniquely identify an entity occurrence
  - Each attribute, which makes up the key, is a simple key in its own right
  - {Roll #, Course #}

Students

Roll #	First Name	Last Name	DoB	Passport #	Aadhaar #	Department

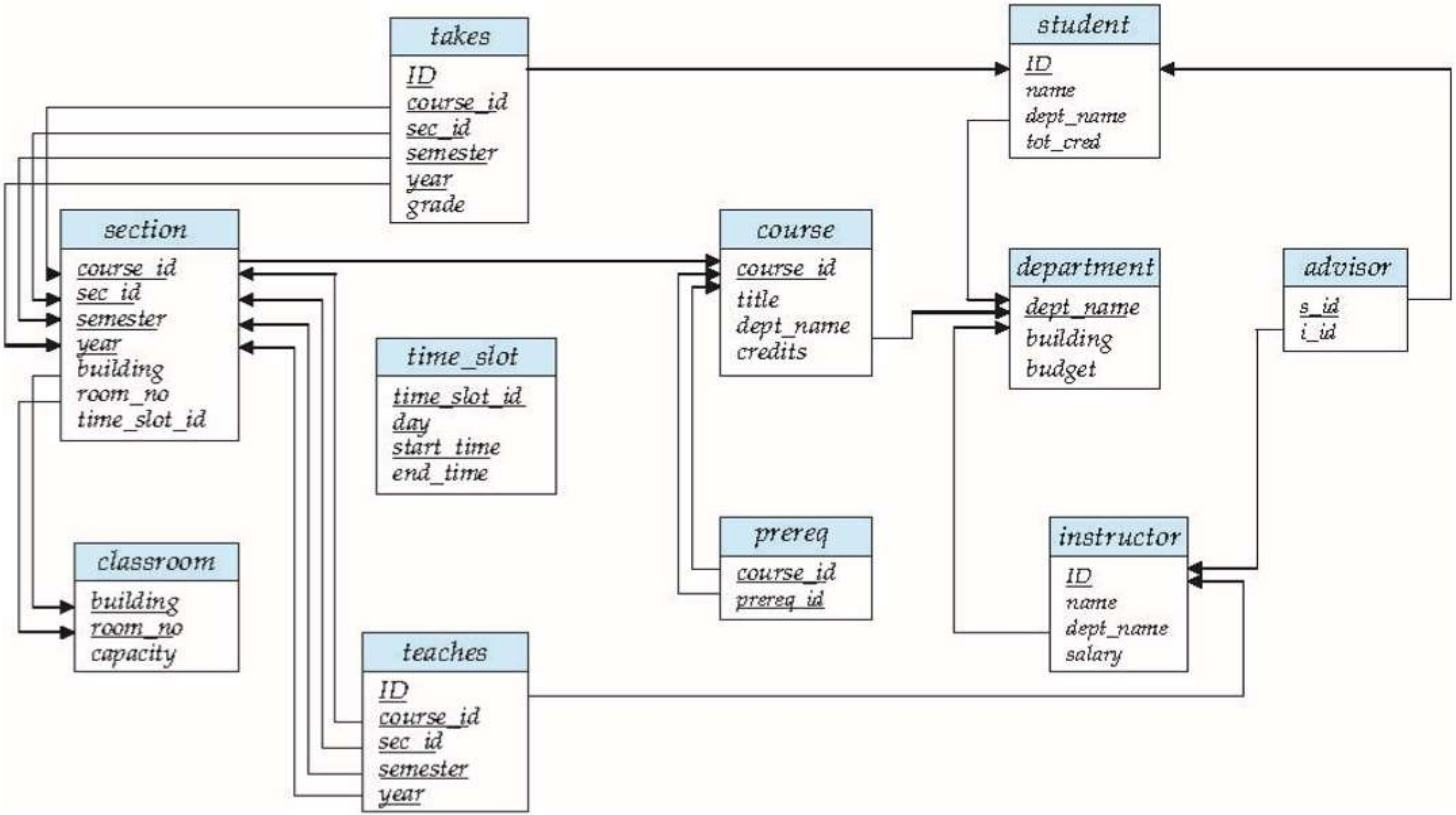
Courses

Course #	Course Name	Credits	L-T-P	Department

Enrolment

Roll #	Course #	Instructor ID

Schema Diagram for University Database



Relational Integrity constraints or RULES

Concept of Null or Unknown Value

- Null represent a value for an attribute that is currently unknown or it Is not applicable for this tuple.
- Null is not same as 0 or zero value or a text string.
- It represent absence of value
- At a particular time the following table might not have the value of age (of Rajesh) and Job (of Raja)

Name	Age	Job
Rajesh	-	Clerk
Raja	23	-
Amit	43	Sales

Difference between Null and Not Applicable

- Blank against Amit might indicate that pension is not applicable to this employee as per firm policy.

Name	Age	Job	Pension_Date
Rajesh	-	Clerk	-
Raja	23	-	01/01/1999
Amit	43	Sales	-

Best way to represent it as :

Name	Age	Job	Pension_Date
Rajesh	null	Clerk	null
Raja	23	null	01/01/1999
Amit	43	Sales	Not Applicable

Issues in basic STUDENT table

Basic table suffers from following anomalies:

- There is no check on storing duplicate records
- There is no mandatory column
- There is no check on validity of data

To ensure all these features, there is a need to apply constraints during creation of table.

Relational Integrity constraints/RULES

Relational Integrity constraints is referred to conditions which must be present for a valid relation. These integrity constraints are derived from the rules that the database represents.

There are many types of integrity constraints. Constraints on the Relational database management system is mostly divided into three main categories are:

1. **Domain constraints**
2. **Entity integrity or Key/primary key constraints**
3. **Referential integrity constraints**

Relational Integrity constraints or RULES

Data Integrity

The following categories of data integrity exist with each RDBMS:

- **Entity Integrity:** There are no duplicate rows in a table.
- **Domain Integrity:** Enforces valid entries for a given column by restricting the type, the format, or the range of values.
- **Referential integrity:** Rows cannot be deleted, which are used by other records.
- **User-Defined Integrity:** Enforces some specific business rules that do not fall into entity, domain or referential integrity.

Levels of Constraints

Column level constraints

Constraints which are applied on single column of table are known as column level constraints.

For example:

Roll_Number is Unique.

Table level constraints

Constraints which are applied on more than one column of a table are known as table level constraints.

For example:

Roll_Number and Class combination is unique in primary School database

[In primary school Roll number of students starts with one in each class]

Types of Constraints

- Not Null
- Unique
- Primary key
- Check
- Foreign key
- Default

Check Constraint

- Check constraints allow Oracle to verify the validity of data being entered on a table against a set of constants. These constants act as valid values.
- The Check constraint consists of the keyword CHECK followed by parenthesized conditions.
- Check constraints must be specified as a logical expression that evaluates either to TRUE or FALSE.
- Syntax:
- Columnname datatype (size) [constraint constraintname] CHECK (logical expression)

Example [Check: Column level constraint]

```
CREATE TABLE student
    (
    Roll_Number Number(4) ,  Name Char(15),
    Class Char(10) Check (class in ('BE','ME','MCA')),
    Marks Number(4) Check (marks>=0), DOB Date
    );
```

OR

```
    CREATE TABLE student
    (
    Roll_Number Number(4) ,  Name Char(15),
    Class Char(10) Constraint class_check Check (class in ('BE','ME','MCA')),
    Marks Number(4) Check (marks>=0), DOB Date
    );
```

Example [Check: Table level constraint]

```
CREATE TABLE library
(
Roll_Number Number(4) Primary Key,
Book_Number Number(10) NOT NULL,
DOI Date, DOR Date,
Check (DOR>DOI);
```

OR

```
CREATE TABLE library
(
Roll_Number Number(4) Primary Key,
Book_Number Number(10) NOT NULL,
DOI Date, DOR Date,
Constraint Date_Check Check (DOR>DOI);
```

Example

For example, the following program creates a new table called CUSTOMERS and adds five columns. Here, we add a CHECK with AGE column, so that you cannot have any CUSTOMER who is below 18 years.

```
CREATE TABLE CUSTOMERS(
  ID   INT              NOT NULL,
  NAME VARCHAR (20)     NOT NULL,
  AGE  INT              NOT NULL CHECK (AGE >= 18),
  ADDRESS  CHAR (25) ,
  SALARY   DECIMAL (18, 2),
  PRIMARY KEY (ID)
);;
```

If the CUSTOMERS table has already been created, then to add a CHECK constraint to AGE column, you would write a statement like the one given below.

```
ALTER TABLE CUSTOMERS
   MODIFY AGE INT NOT NULL CHECK (AGE >= 18 );
```

Or

```
ALTER TABLE CUSTOMERS
   ADD CONSTRAINT myCheckConstraint CHECK(AGE >= 18);
```

DROP a CHECK Constraint

```
ALTER TABLE CUSTOMERS
   DROP CONSTRAINT myCheckConstraint;
```

Limitations of Check Constraint

Rno	Name	Class	Marks	DOB
1	Ram	BE	60	12-DEC-1980
2	Rajesh	MCA	70	13-JAN-1989
3	Surinder	ME	78	10-JUN-1980

Check class in (‘BE’, ‘ME’, ‘MCA’)

Limitations of Check Constraint

Rno	Name	Class	Marks	DOB
1	Ram	BE	60	12-DEC-1980
2	Rajesh	MCA	70	13-JAN-1989
3	Surinder	ME	78	10-JUN-1980
4	Rahat	MBA	76	15-MAY-1980

Check class in (‘BE’, ‘ME’, ‘MCA’)
It will not be allowed…

Limitations of Check Constraint

Rno	Name	Class	Marks	DOB
1	Ram	BE	60	12-DEC-1980
2	Rajesh	MCA	70	13-JAN-1989
3	Surinder	ME	78	10-JUN-1980
4	Rahat	MBA	76	15-MAY-1980

Check class in ('BE', 'ME', 'MCA')
To allow it we have to add it in check list by changing schema

Limitations of Check Constraint

- It will not be suitable if check values need to be changed in future.
- Class can be added or removed in future, for check constraint is not desirable.
- Apply check constraint only if its value is not going to change in future.
- For example,
- Marks number(3) check (marks>=0)
- Gender

  Gender char(10) check (gender in

  ('male', 'female', 'transgender'));

NOT NULL Constraint

- By default, a column can hold NULL values.

- If you do not want a column to have a NULL value, then you need to define such a constraint on this column specifying that NULL is now not allowed for that column.

- A NULL is not the same as no data, rather, it represents unknown data.

NOT NULL Constraint

- Specifies if the column must contain value or might not contain any.
- By default all columns in a table allow nulls.
- NOT NULL specifies that all rows in the table to have value for specified column.
- Syntax:

  Columnname datatype (size) [constraint constraintname] NOT NULL

Example [Not Null: Column level constraint]

```
CREATE TABLE student
(
Roll_Number Number(4) Not Null,
Name Char(15), Class Char(10), Marks Number(4),
DOB Date
);
```

Example

```
CREATE TABLE CUSTOMERS(
   ID   INT                                    NOT NULL,
   NAME VARCHAR (20)                 NOT NULL,
   AGE  INT                                    NOT NULL,
   ADDRESS  CHAR (25) ,
   SALARY   DECIMAL (18, 2),
   PRIMARY KEY (ID)
);
```

If CUSTOMERS table has already been created, then to add a NOT NULL constraint to the SALARY column in

ALTER TABLE CUSTOMERS MODIFY SALARY DECIMAL (18, 2) NOT NULL;

UNIQUE Constraint

- **The UNIQUE Constraint prevents two records from having identical values in a column.**
- E.g. In the CUSTOMERS table, for example, you might want to prevent two or more people from having an identical age.

  - Enforce uniqueness of the column or group of columns.
  - Syntax:

    Columnname datatype (size) [constraint constraintname] UNIQUE

Example [Unique: Column level constraint]

```
CREATE TABLE student
(
Roll_Number Number(4) Unique,
Name Char(15), Class Char(10), Marks Number(4),
DOB Date
);
```

OR

```
CREATE TABLE student
(
Roll_Number Number(4) Constraint RNO_Unique Unique,
Name Char(15), Class Char(10), Marks Number(4),
DOB Date
);
```

Example

For example, the following SQL query creates a new table called CUSTOMERS and adds five columns.
Here, the AGE column is set to UNIQUE, so that you cannot have two records with the same age.

```
CREATE TABLE CUSTOMERS(
   ID   INT              NOT NULL,
   NAME VARCHAR (20)     NOT NULL,
   AGE  INT              NOT NULL UNIQUE,
   ADDRESS  CHAR (25) ,
   SALARY   DECIMAL (18, 2),
   PRIMARY KEY (ID)
);
```

Adding Constraint

```
ALTER TABLE CUSTOMERS
   ADD CONSTRAINT myUniqueConstraint UNIQUE(AGE, SALARY);
```

DROP a UNIQUE Constraint

To drop a UNIQUE constraint, use the following SQL query.

```
ALTER TABLE CUSTOMERS
   DROP CONSTRAINT myUniqueConstraint;
```

Example [Unique:Table level constraint]

```
CREATE TABLE primary_student
(
Roll_Number Number(4),  Name Char(15), Class Char(10),
Marks Number(4), DOB Date,
Unique(Roll_Number, Class)
);
```

OR

```
CREATE TABLE primary_student
(
Roll_Number Number(4), Name Char(15), Class Char(10), Marks Number(4),        DOB
Date,
Constraint RNO_Class_Unique  Unique(Roll_Number, Class)
);
```

DEFAULT constraint

The DEFAULT constraint provides a default value to a column when the INSERT INTO statement does not provide a specific value.

- The default value constraint allows the user to insert the values in the columns where the user do not want to insert the value.
- This is not actually a constraint and used to insert default value if value is missing for this column.
- The datatype of the default value should match the datatype of the column.
- Syntax:
- Columnname datatype (size) default value

Example [Default: Column level constraint]

```
CREATE TABLE student
  (
  Roll_Number Number(4) Primary key,
  Name Char(15),
  Class Char(10) Check (class in ('BE','ME','MCA')),
  Marks Number(4) default 60,
  DOB Date
  );
```

DEFAULT constraint

Example

For example, the following SQL creates a new table called CUSTOMERS and adds five columns.

Here, the **SALARY column is set to 5000.00 by default,** so in case the INSERT INTO statement does not provide a value for this column, then by default this column would be set to 5000.00.

```
CREATE TABLE CUSTOMERS(
   ID   INT              NOT NULL,
   NAME VARCHAR (20)     NOT NULL,
   AGE  INT              NOT NULL,
   ADDRESS  CHAR (25) ,
   SALARY   DECIMAL (18, 2) DEFAULT 5000.00,
   PRIMARY KEY (ID)
);
```

if the CUSTOMERS table has already been created, then to add a DEFAULT constraint to the SALARY column, you would write a query like the one which is shown in the code block below.

```
ALTER TABLE CUSTOMERS
MODIFY SALARY  DECIMAL (18, 2) DEFAULT 5000.00;
```

Drop Default Constraint

To drop a DEFAULT constraint, use the following SQL query.

```
ALTER TABLE CUSTOMERS
   ALTER COLUMN SALARY DROP DEFAULT;
```

Entity Integrity OR Key/Primary Key constraints

- **An attribute that can uniquely identify a tuple in a relation is called the key of the table.**
- The value of the attribute for different tuples in the relation has to be unique.
- <u>Entity integrity constraints are rules for primary keys:</u>

- **The primary key cannot have a null value.**
- **If the primary key is a composite key, none of the fields in the key can contain a null value.**

Entity Integrity OR Key/Primary Key constraints

Example:

- In the given table, CustomerID is a key attribute of Customer Table.
- It is most likely to have a single key for one customer, CustomerID =1 is only for the CustomerName =" Google".

CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Inactive

Primary Key Constraint

- It is a combination of both the unique key constraint as well as the not null constraint.
- A primary key is used to identify each row of the table uniquely.
- A primary key may be either consists of a single field (Simple key) or group of fields (Composite Primary Key).
- Oracle enforces all PRIMARY KEY constraints using indexes.
- Syntax:

  Columnname datatype (size) [constraint constraintname] PRIMARY KEY

Example [Primary Key: Column level Constraint]

```
CREATE TABLE student
(
Roll_Number Number(4) Primary Key,
Name Char(15), Class Char(10), Marks Number(4),
DOB Date
);
```

OR

```
CREATE TABLE student
(
Roll_Number Number(4) Constraint RNO_PK Primary Key,
Name Char(15), Class Char(10), Marks Number(4),
DOB Date
);
```

Example [Primary Key: Table level Constraint]

```
CREATE TABLE primary_student
(
Roll_Number Number(4), Name Char(15), Class Char(10),
Marks Number(4), DOB Date,
Primary Key(Roll_Number, Class)
);
```

OR

```
CREATE TABLE primary_student
(
Roll_Number Number(4), Name Char(15), Class Char(10), Marks Number(4),        DOB
Date,
Constraint RNO_Class_PK  Primary Key(Roll_Number, Class)
);
```

Referential integrity constraints

- Referential integrity constraints is base on the **concept of Foreign Keys.**
- A foreign key is an important attribute of a relation **which should be referred to in other relationships.**
- Referential integrity constraint state happens where relation refers to a key attribute of a different or same relation. However, that key element must exist in the table.

Referential Integrity

- Referential Integrity can help to avoid data input errors, voids data inconsistency and quality problems.
- In a relational database, **referential integrity means that a foreign key value cannot be entered in one table unless it matches an existing primary key in another table.**
- You cannot change a primary key that has matching child table records
  - A child table that has a foreign key for a different record
- Referential integrity also can prevent the deletion of a record has a primary key that matches foreign keys in another table.

CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Inactive

Customer

Billing

InvoiceNo	CustomerID	Amount
1	1	$100
2	1	$200
3	2	$150

In the above example, we have 2 relations, Customer and Billing.

Tuple for CustomerID =1 is referenced twice in the relation Billing. So we know CustomerName=Google has billing amount $300

Foreign Key Constraint

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30

FK

PK

Department table

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical

Only those values are allowed which exist in dno of Department table

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

FK

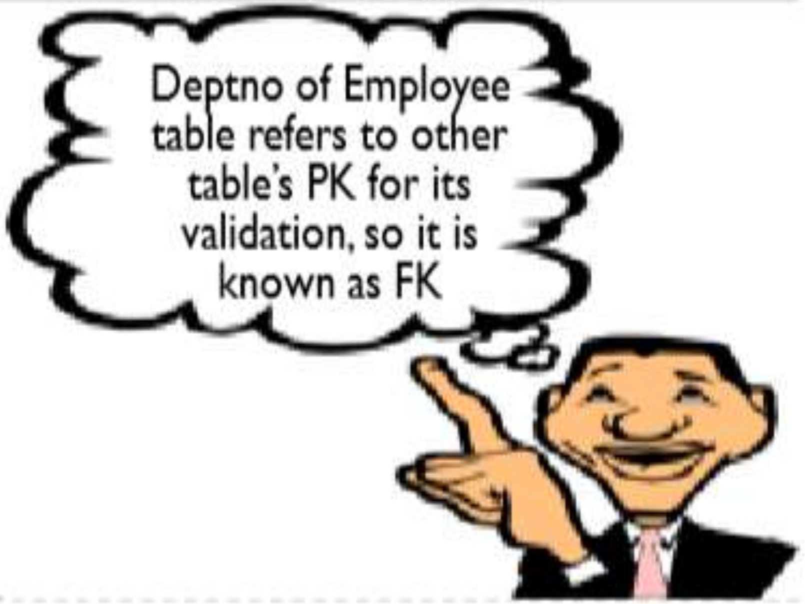
Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30
6	Rupinder	Prof	40

No corresponding department exists in University. Not allowed.

PK

Department table

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical



Foreign key Constraint

- Used to enforce referential integrity of data and establishes a relationship among tables.
- A foreign key may be a single column or the combinations of columns, which derive their values, based on the primary key of some other table.
- A table in which foreign key is present is known as child table and to which it refers is known as master or parent table.
- Its Syntax is:

  Columnname datatype (size) references tablename [(columnname)]

  [ON DELETE RESTRICT/ON DELETE CASCADE/ON DELETE SET NULL]

Syntax: Case Study EMP-DEPT

Employee table

FK

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30

Department table

PK

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical

1

```
Create table department
(deptno number(2) primary key,
dname char(20));
```

2

```
Create table employee
(eno number(2) primary key,
ename char(20),job char(10),
deptno number(2) references department(deptno));
```

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

FK

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30

PK

Department table

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical

What will happen if we try to delete deptno 30 record from department table with

Delete department where deptno=30;

It should not be allowed, Oracle achieves this by enforcing ON DELETE RESTRICT constraint on FK.

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

FK

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30

Department table

PK

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical

What will be the solution if we have to delete department number 30 from department table

Solution: Apply ON DELETE CASCADE constraint during creation of table so that department and corresponding employees records removed together.

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30

FK

PK

Department table

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical

Result of Delete department where deptno=30 ; will be …..

With ON DELETE CASCADE constraint

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20

FK

PK

Department table

Deptno	Dname
10	Computer
20	Chemical

Record of department number 30 has been removed from both tables.

With ON DELETE CASCADE constraint corresponding employees have been fired

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	30

FK

PK

Department table

Deptno	Dname
10	Computer
20	Chemical
30	Mechanical

What will be the solution if we have to delete department number 30 from department table, without firing corresponding employees?

Solution: Apply ON DELETE SET NULL
Corresponding employees records will not be removed and their dno will be set to NULL

Concept of Foreign Key: Case Study EMP-DEPT

Employee table

FK

Eno	Ename	Job	Deptno
1	Ram	Prof	10
2	Rajesh	Asst Prof	20
3	Suresh	Asst Prof	10
4	Ramesh	Prof	20
5	Surinder	Asst Prof	

Department table

PK

Deptno	Dname
10	Computer
20	Chemical

Record of department number 30 has been removed from department table and corresponding employee record set to NULL

With ON DELETE SET NULL Constraint corresponding employees records has not been removed and set to NULL

Creating Master and Child tables with ON DELETE RESTRICT (By default)

- Parent Table
- SQL>CREATE TABLE department
  (DEPTNO NUMBER(2) PRIMARY KEY,
  DNAME CHAR(10));
- Child table
- SQL>CREATE TABLE employee(
  empno NUMBER(4) PRIMARY KEY,
  ename CHAR(20), job CHAR(10),
  deptno NUMBER(2) REFERENCES department(deptno)
  ON DELETE RESTRICT);

Creating Master and Child tables with ON DELETE CASCADE

- Parent Table
- SQL>CREATE TABLE department
  (DEPTNO NUMBER(2) PRIMARY KEY,
  DNAME CHAR(10));
- Child table
- SQL>CREATE TABLE employee(
  empno NUMBER(4) PRIMARY KEY,
  ename CHAR(20), job CHAR(10),
  deptno NUMBER(2) REFERENCES department(deptno)
  ON DELETE CASCADE);

Creating Master and Child tables with ON DELETE SET NULL

- Parent Table
- SQL>CREATE TABLE department
  (DNO NUMBER(2) PRIMARY KEY,
  DNAME CHAR(10));
- Child table
- SQL>CREATE TABLE employee(
  empno NUMBER(4) PRIMARY KEY,
  ename CHAR(20), job CHAR(10),
  dno NUMBER(2) REFERENCES department(deptno)
  ON DELETE SET NULL);

USER_CONSTRAINTS and USER_CONS_COLUMNS tables to list constraints details

- To see the name of the constraints on a table you can query USER_CONSTRINTS table (maintains internally by Oracle).

To query it:

- SELECT * FROM USER_CONSTRAINTS WHERE

  TABLE_NAME='NAME_OF_TABLE';

For example:

- SELECT * FROM USER_CONSTRAINTS WHERE TABLE_NAME='STUDENT';
- SELECT * FROM USER_CONS_COLUMNS WHERE TABLE_NAME='STUDENT';

Viewing Constraints

Query the `USER_CONSTRAINTS` table to view all constraint definitions and names.

```
SELECT   constraint_name, constraint_type,
         search_condition
FROM     user_constraints
WHERE    table_name = 'EMPLOYEES';
```

CONSTRAINT_NAME	C	SEARCH_CONDITION
EMP_LAST_NAME_NN	C	"LAST_NAME" IS NOT NULL
EMP_EMAIL_NN	C	"EMAIL" IS NOT NULL
EMP_HIRE_DATE_NN	C	"HIRE_DATE" IS NOT NULL
EMP_JOB_NN	C	"JOB_ID" IS NOT NULL
EMP_SALARY_MIN	C	salary > 0
EMP_EMAIL_UK	U	

...

Viewing the Columns Associated with Constraints

View the columns associated with the constraint names in the `USER_CONS_COLUMNS` view.

```
SELECT   constraint_name, column_name
FROM     user_cons_columns
WHERE    table_name = 'EMPLOYEES';
```

CONSTRAINT_NAME	COLUMN_NAME
EMP_DEPT_FK	DEPARTMENT_ID
EMP_EMAIL_NN	EMAIL
EMP_EMAIL_UK	EMAIL
EMP_EMP_ID_PK	EMPLOYEE_ID
EMP_HIRE_DATE_NN	HIRE_DATE
EMP_JOB_FK	JOB_ID
EMP_JOB_NN	JOB_ID

...

Adding a Constraint Syntax

Use the `ALTER TABLE` statement to:

- Add or drop a constraint, but not modify its structure
- Enable or disable constraints
- Add a `NOT NULL` constraint by using the `MODIFY` clause

```
ALTER TABLE  table
ADD [CONSTRAINT constraint] type (column);
```

Adding a Constraint

Add a FOREIGN KEY constraint to the EMPLOYEES table indicating that a manager must already exist as a valid employee in the EMPLOYEES table.

```
ALTER TABLE      employees
ADD CONSTRAINT   emp_manager_fk
  FOREIGN KEY(manager_id)
  REFERENCES employees(employee_id);
Table altered.
```

Cascading Constraints

- The `CASCADE CONSTRAINTS` clause is used along with the `DROP COLUMN` clause.
- The `CASCADE CONSTRAINTS` clause drops all referential integrity constraints that refer to the primary and unique keys defined on the dropped columns.
- The `CASCADE CONSTRAINTS` clause also drops all multicolumn constraints defined on the dropped columns.

Disabling Constraints

- Execute the `DISABLE` clause of the `ALTER TABLE` statement to deactivate an integrity constraint.
- Apply the `CASCADE` option to disable dependent integrity constraints.

```
ALTER TABLE          employees
DISABLE CONSTRAINT   emp_emp_id_pk CASCADE;
Table altered.
```

Enabling Constraints

- Activate an integrity constraint currently disabled in the table definition by using the `ENABLE` clause.

```
ALTER TABLE          employees
ENABLE CONSTRAINT    emp_emp_id_pk;
Table altered.
```

- A `UNIQUE` or `PRIMARY KEY` index is automatically created if you enable a `UNIQUE` key or `PRIMARY KEY` constraint.

Entity-Relationship Model

Dr. Geeta Kasana
Assistant Professor
CSED ,Thapar Institute of Engineering & Technology

How to design the database?

There are two approaches

- E-R Modeling:
  Identifying entity and relations
- Normalization: Refinement of database designing

Entity-Relation Model

E-R Model

- The Entity-Relationship (ER) model was originally proposed by Peter in 1976
- The ER model is a conceptual data model that views the real world as entities and relationships.
- A basic component of the model is the Entity-Relationship diagram, which is used to visually represent data objects.

Basic Constructs of E-R Modeling

- **Entities**
- Entities are the principal data object about which information is to be collected. Entities are usually recognizable concepts, either concrete or abstract, such as person, places things events, which have relevance to the database. Some specific examples of entities are EMPLOYEES, PROJECTS, and INVOICES. An entity is analogous to a table in the relational model.

Basic Constructs of E-R Modeling

Tangible Entity: Tangible Entities are those entities which exist in the real world physically. ***Example:*** Person, car, etc.

Intangible Entity: Intangible Entities are those entities which exist only logically and have no physical existence. ***Example:*** Bank Account, etc.

Basic Constructs of E-R Modeling

Strong Entity

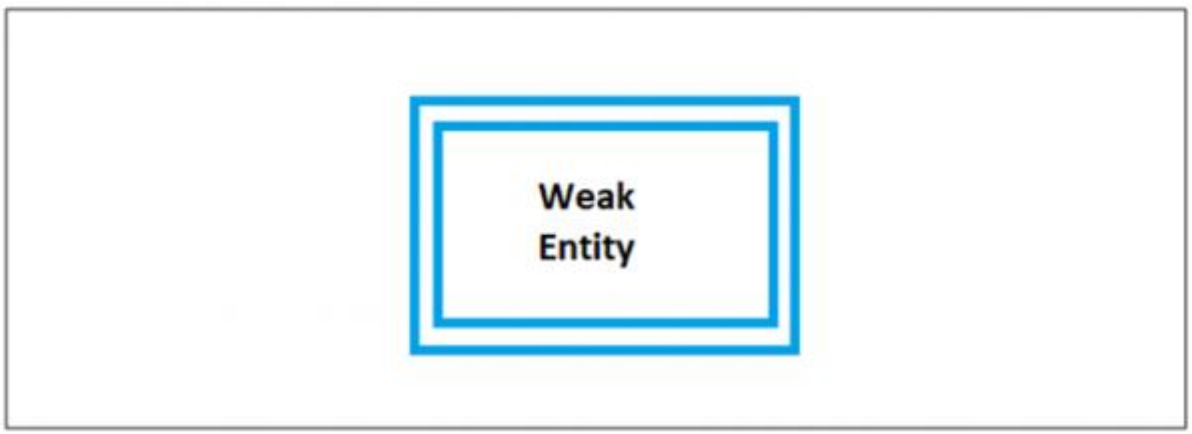
The strong entity has a primary key. Weak entities are dependent on strong entity. Its existence is not dependent on any other entity.

Basic Constructs of E-R Modeling

Weak Entity

The weak entity in DBMS do not have a primary key and are dependent on the parent entity. It mainly depends on other entities.

Weak Entity is represented by double rectangle



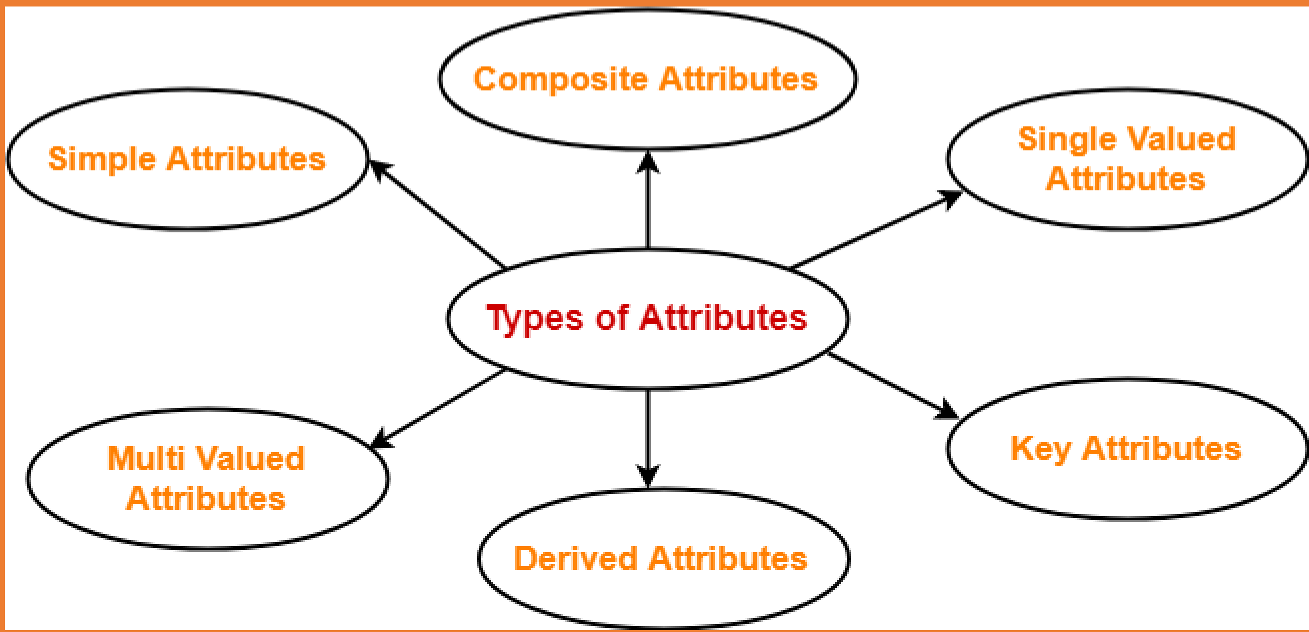
Attributes

Attributes describe the properties of the entity of which they are associated. A particular instance of an attribute is a value. For example, "Ram" is one value of the attribute Name. The domain of an attribute is the collection of all possible values an attribute can have.

We can classify attributes as following:

- Simple
- Composite
- Single-values
- Multi-values
- Derived

Attributes

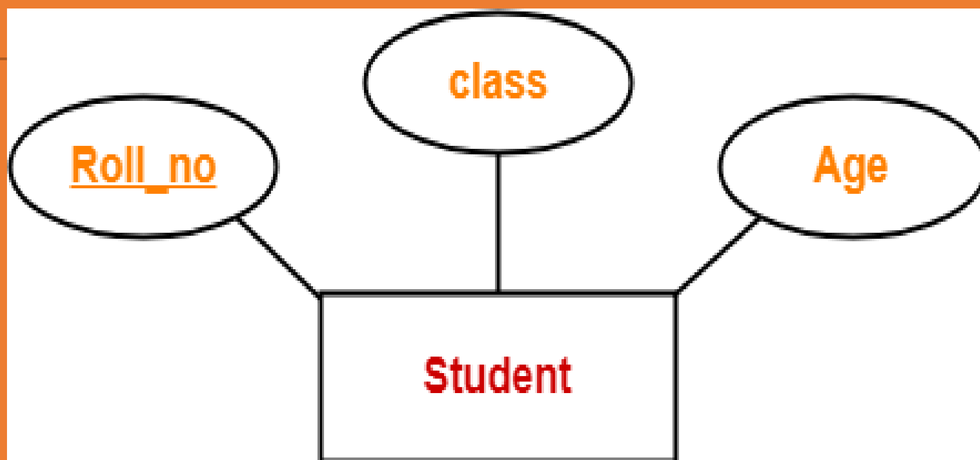


Composite Attributes
Simple Attributes
Single Valued Attributes
Types of Attributes
Multi Valued Attributes
Key Attributes
Derived Attributes

1. Simple Attributes

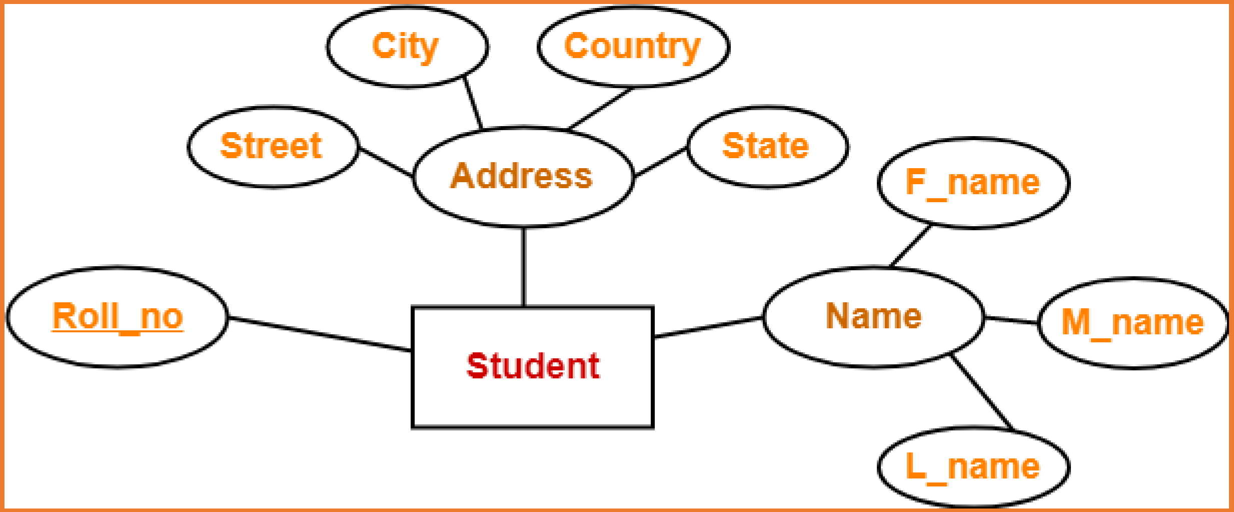
Simple attributes are those attributes which can not be divided further.

Example-



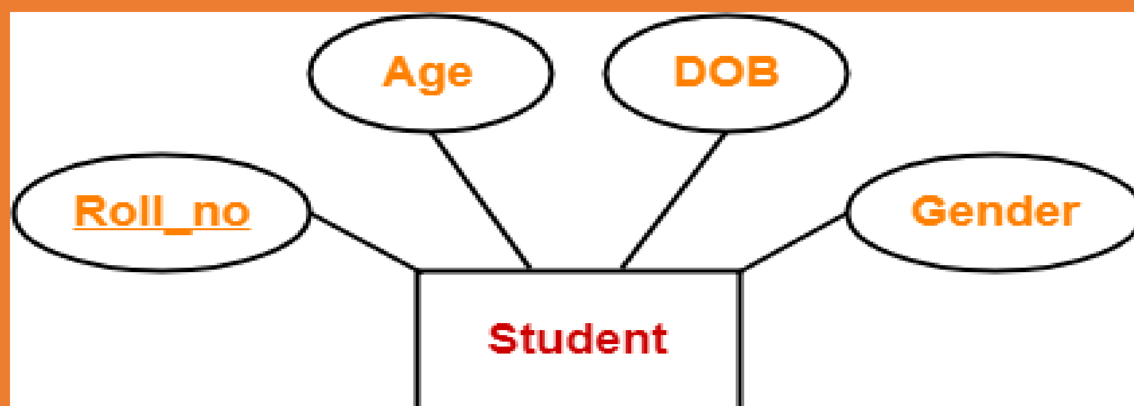
2. Composite Attributes

Composite attributes are those attributes which are composed of many other simple attributes.



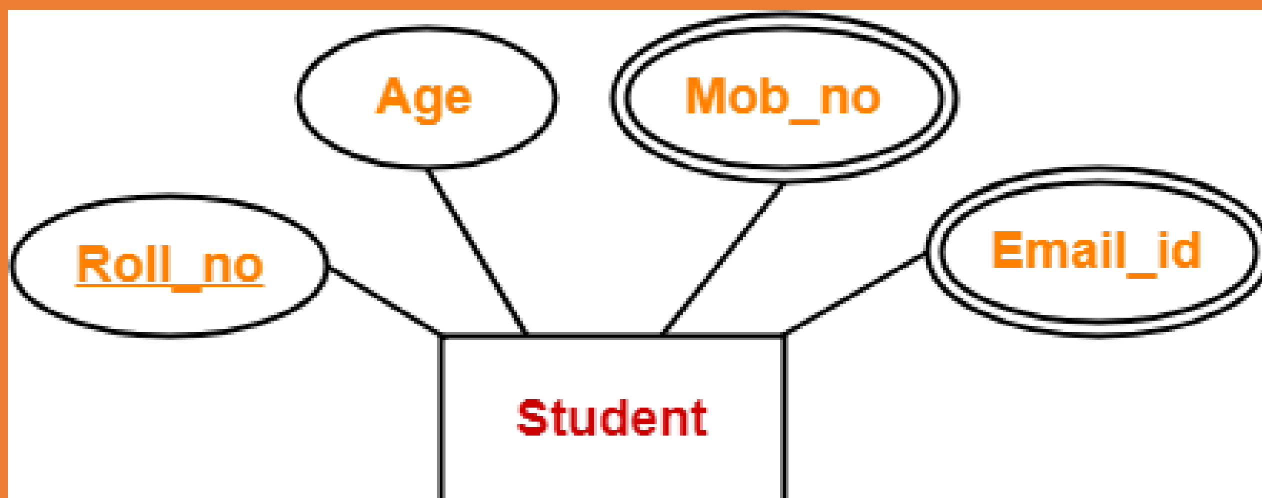
3. Single Valued **Attributes**

Single valued attributes are those attributes which can take only one value for a given entity from an entity set.



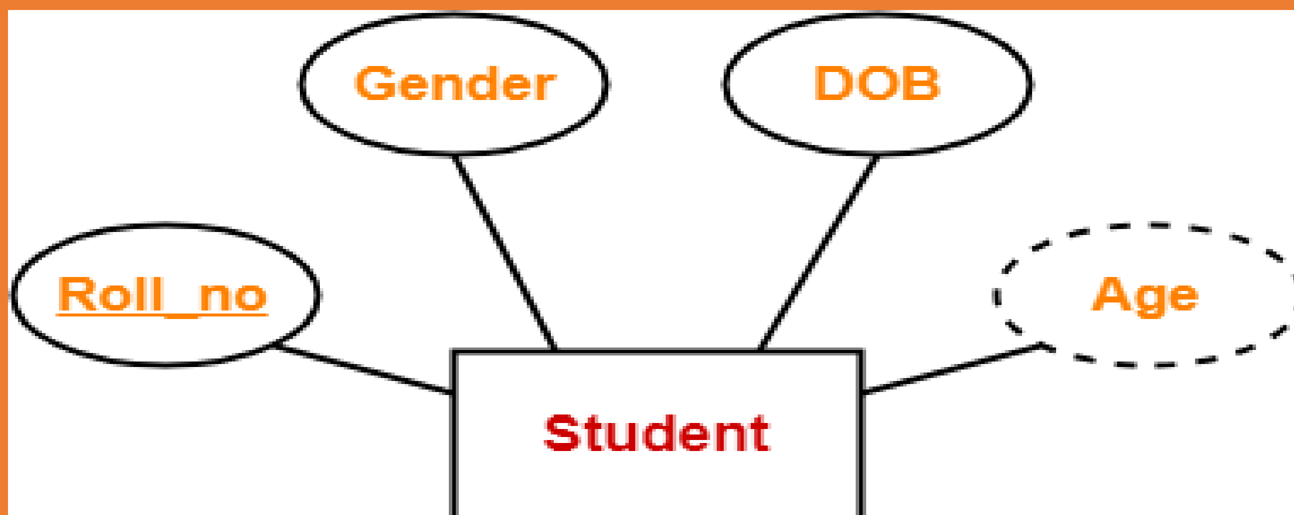
4. Multi Valued **Attributes**

Multi valued attributes are those attributes which can take more than one value for a given entity from an entity set.



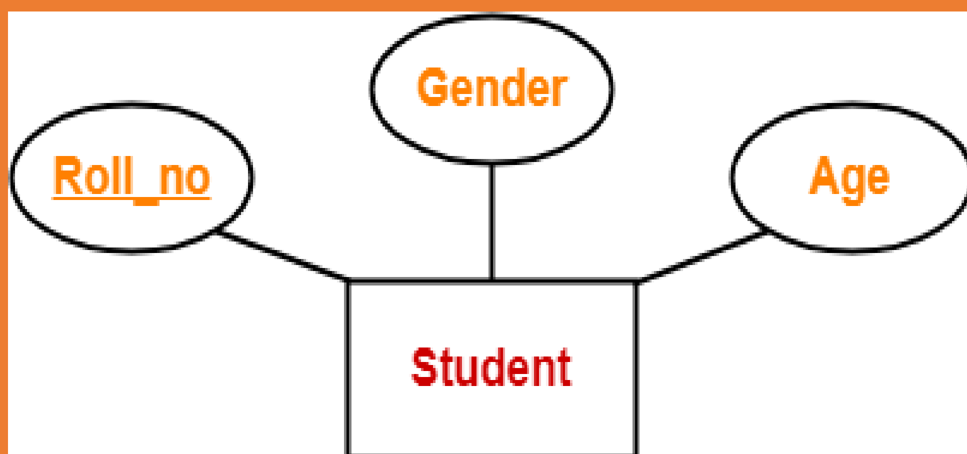
5. Derived **Attributes**

Derived attributes are those attributes which can be derived from other attribute(s).



6. Key **Attributes**

Key attributes are those attributes which can identify an entity uniquely in an entity set.



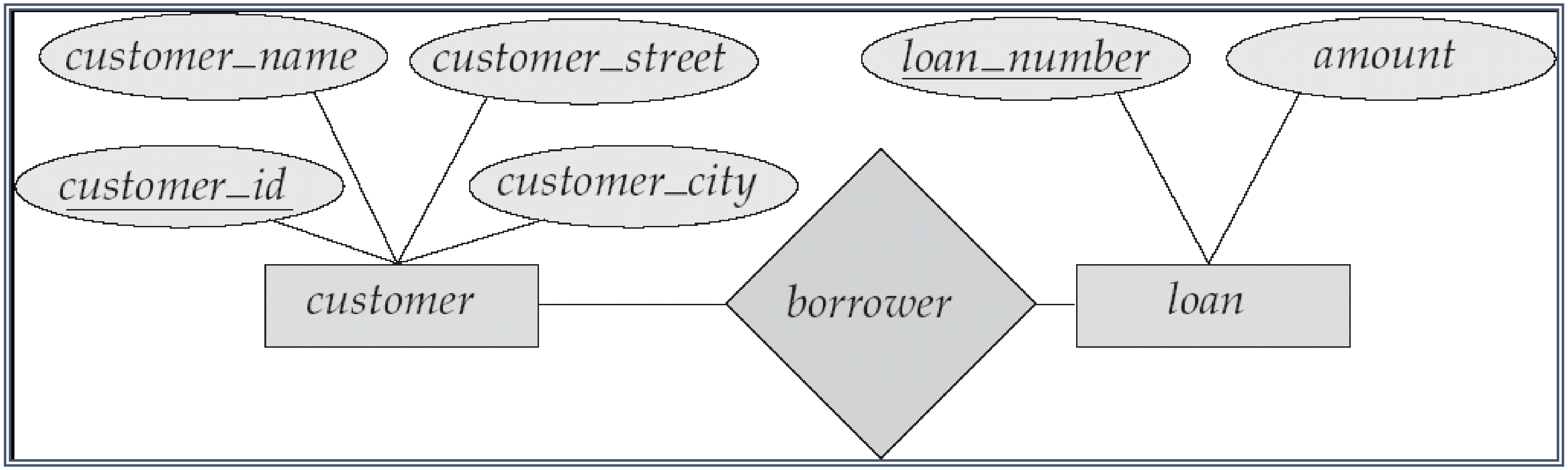
Entity

Student → **Entity Type**

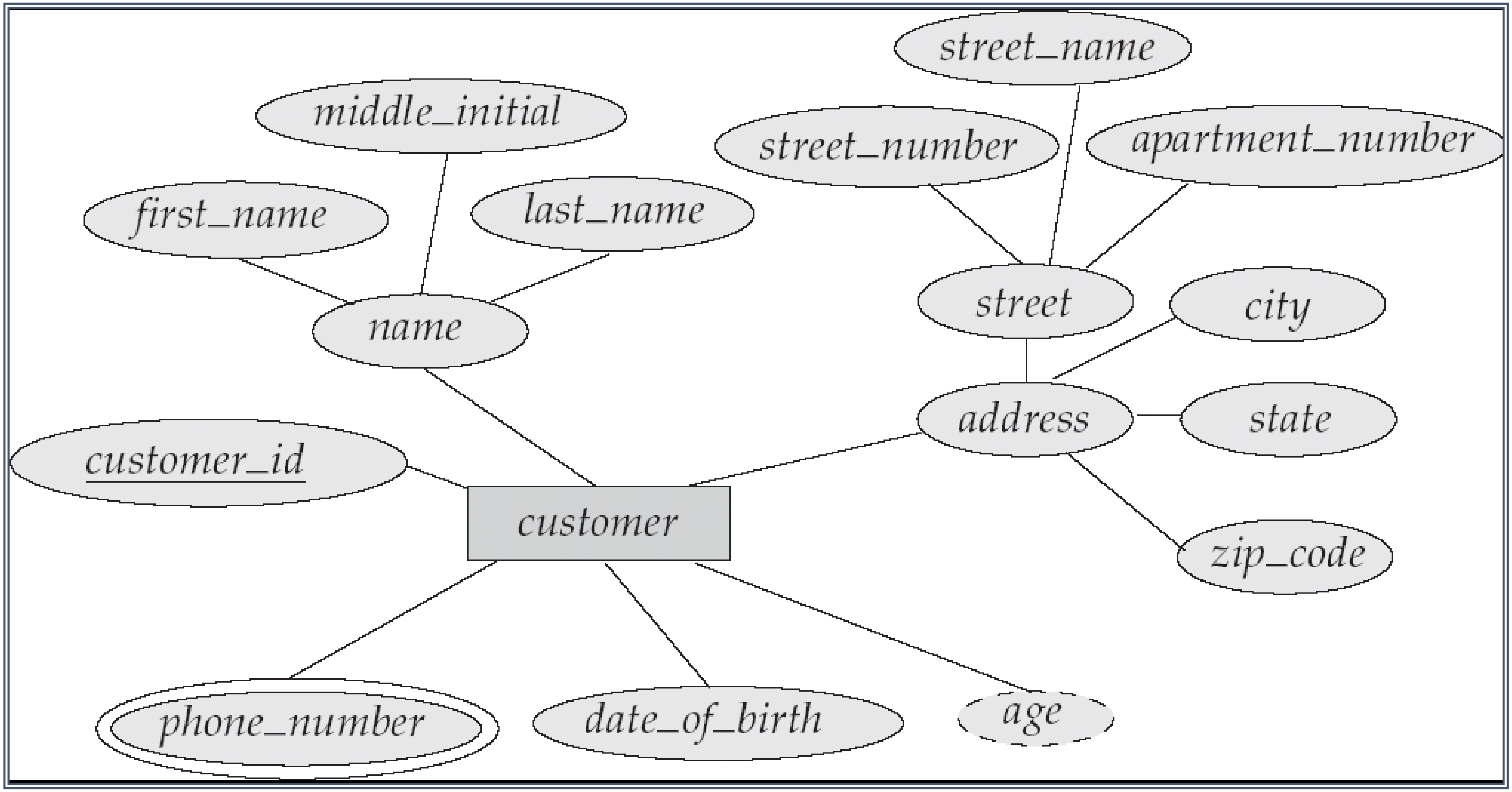
Roll_no	Student_name	Age	Mobile_no
1	Andrew	18	7089117222
2	Angel	19	8709054568
3	Priya	20	9864257315
4	Analisa	21	9847852156

→ **Entity**

E-R Diagrams


customer_name
customer_street
loan_number
amount
customer_id
customer_city
customer
borrower
loan

E-R Diagram With Composite, Multivalued, and Derived Attributes



Relationships

A Relationship represents an association between two or more entities. Relationships are classified in terms of degree, connectivity, cardinality, and existence. An example of a relationship would be:

¨ Employees are assigned to projects

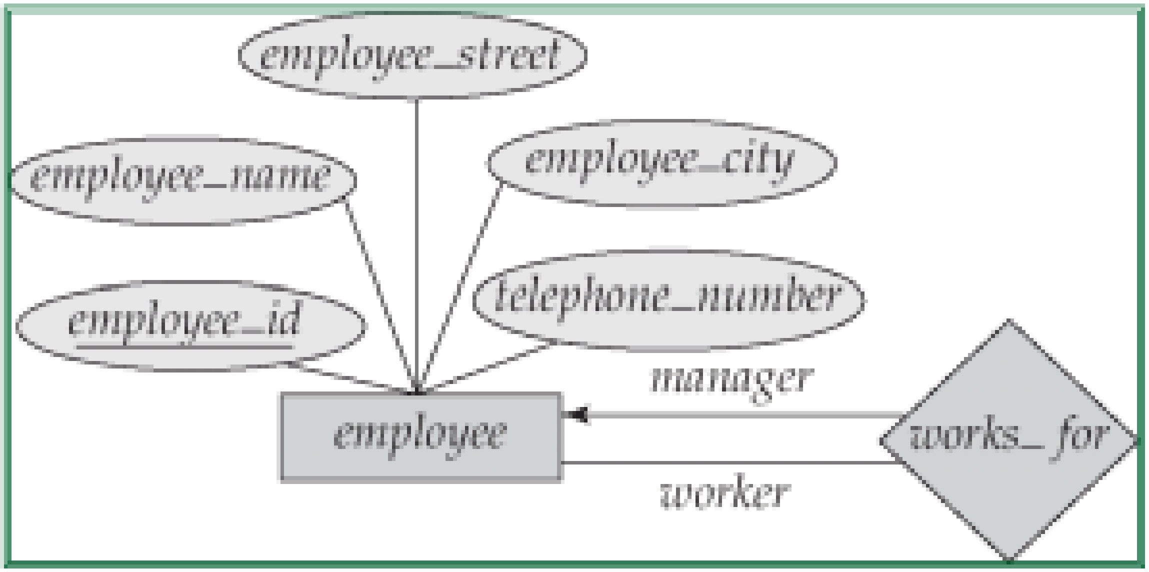
¨ Projects have subtasks

¨ Departments manage one or more projects

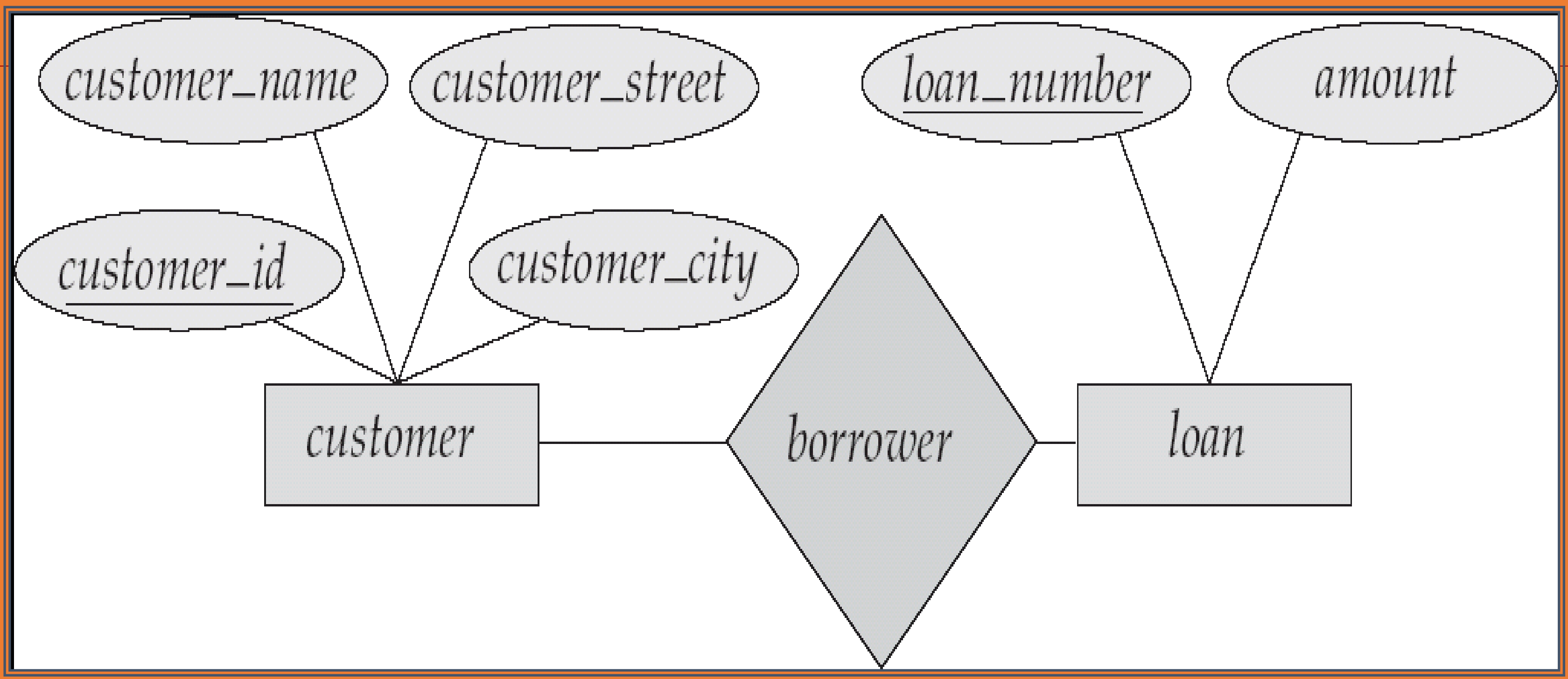
Degree of a Relationship

- The degree of a relationship is the number of entities associated with the relationship. The n-ary relationship is the general form for degree n. Special cases are the binary, and ternary, where the degree is 2, and 3, respectively.
- Binary relationships, the association between two entities are the most common type in the real world.
- A recursive binary relationship occurs when an entity is related to itself. An example might be "some employees are married to other employees".
- A ternary relationship involves three entities and is used when a binary relationship is inadequate. Many modeling approaches recognize only binary relationships. Ternary or n-ary relationships are

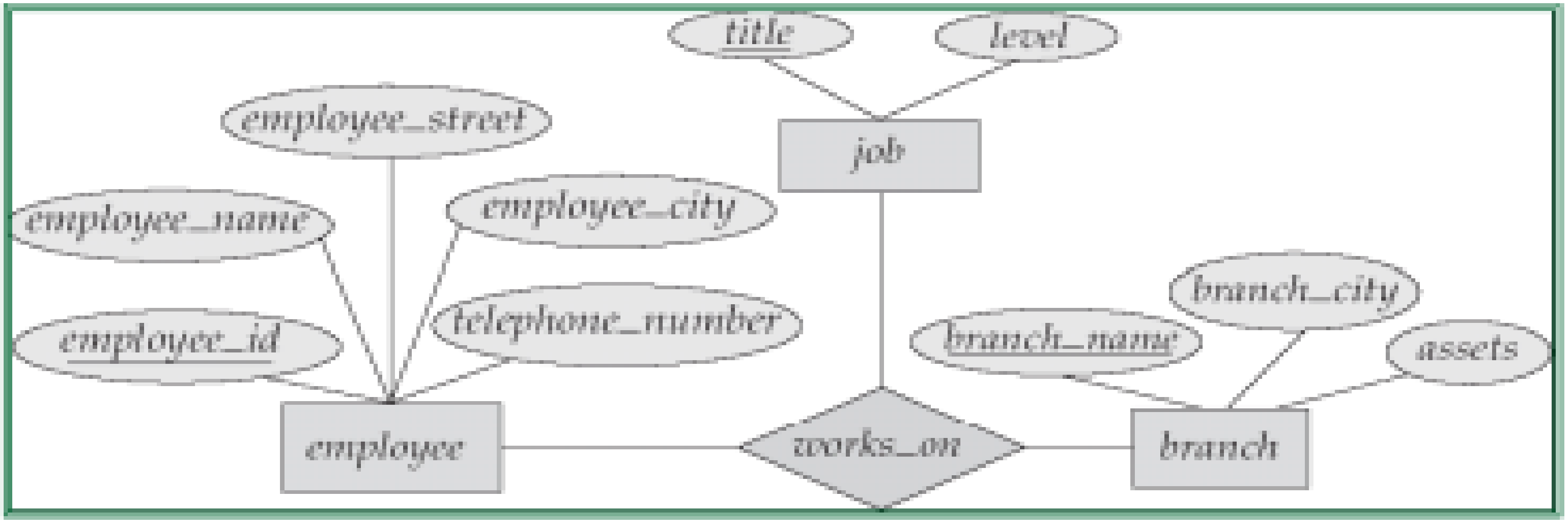
Recursive Relationship



Binary Relationship



Ternary Relationship



Connectivity and Cardinality

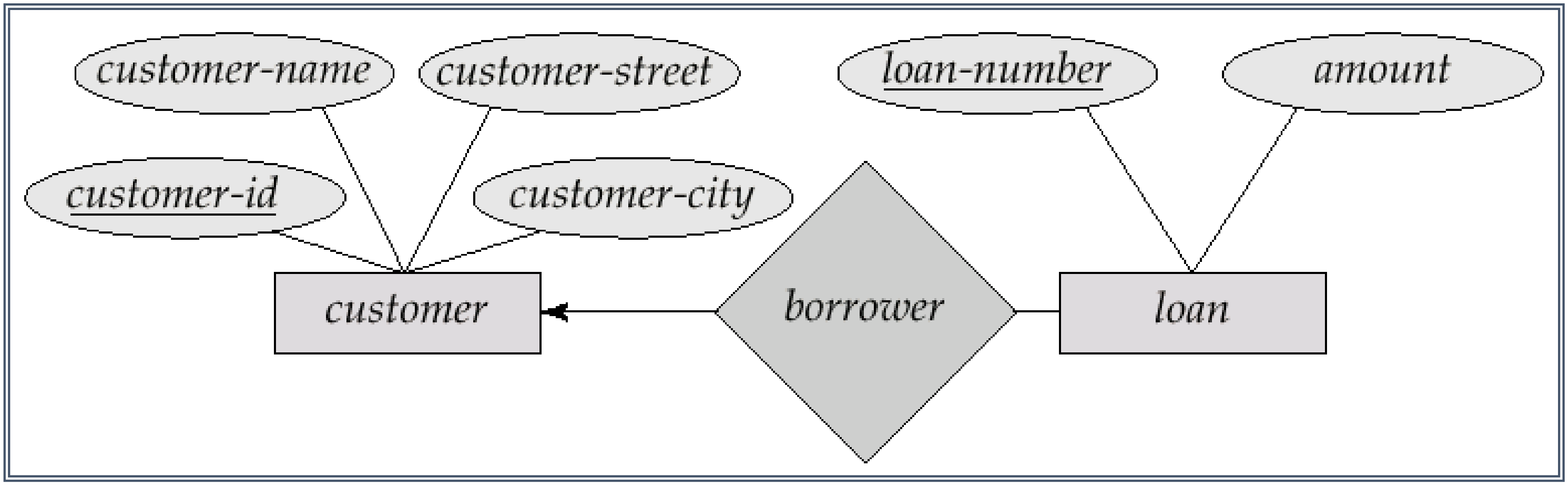
The connectivity of a relationship describes the mapping of associated entity instances in the relationship. The values of connectivity are "one" or "many". The cardinality of a relationship is the actual number of related occurrences for each of the two entities.

The basic types of connectivity for relations are:

- One to One (1:1)
- One to Many (1:M)
- Many to One (M:1)
- Many to Many (M:M)

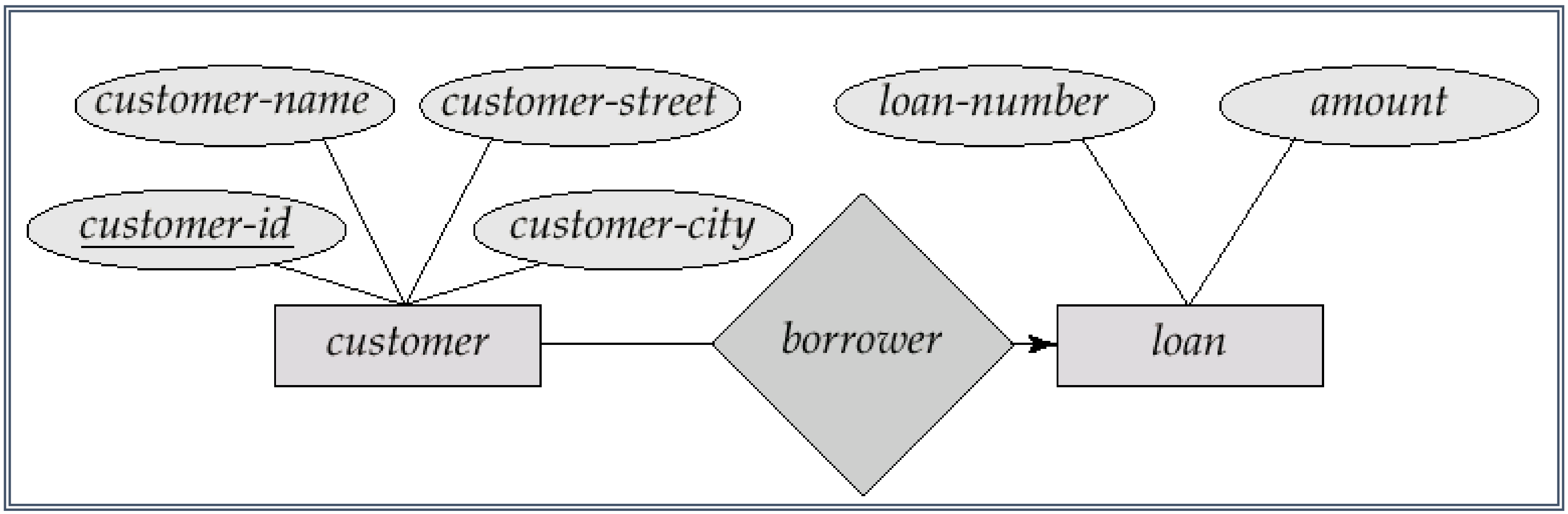
One-To-Many Relationship

- In the one-to-many relationship a loan is associated with at most one customer via *borrower*, a customer is associated with several (including 0) loans via *borrower*



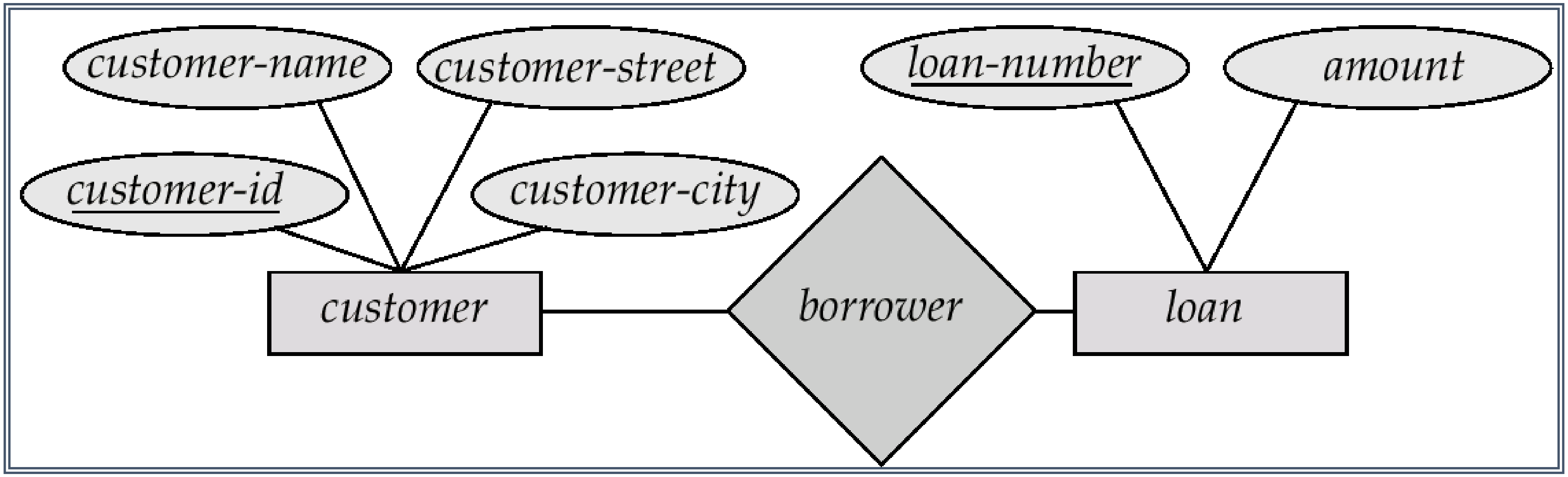
Many-To-One Relationships

- In a many-to-one relationship a loan is associated with several (including 0) customers via *borrower*, a customer is associated with at most one loan via *borrower*



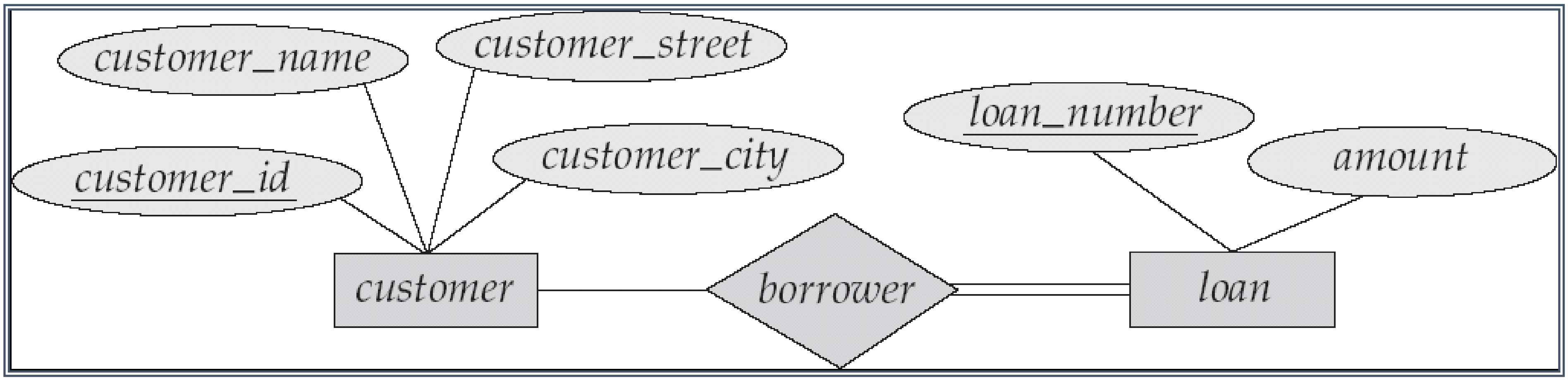
Many-To-Many Relationship

- A customer is associated with several (possibly 0) loans via borrower
- A loan is associated with several (possibly 0) customers via borrower



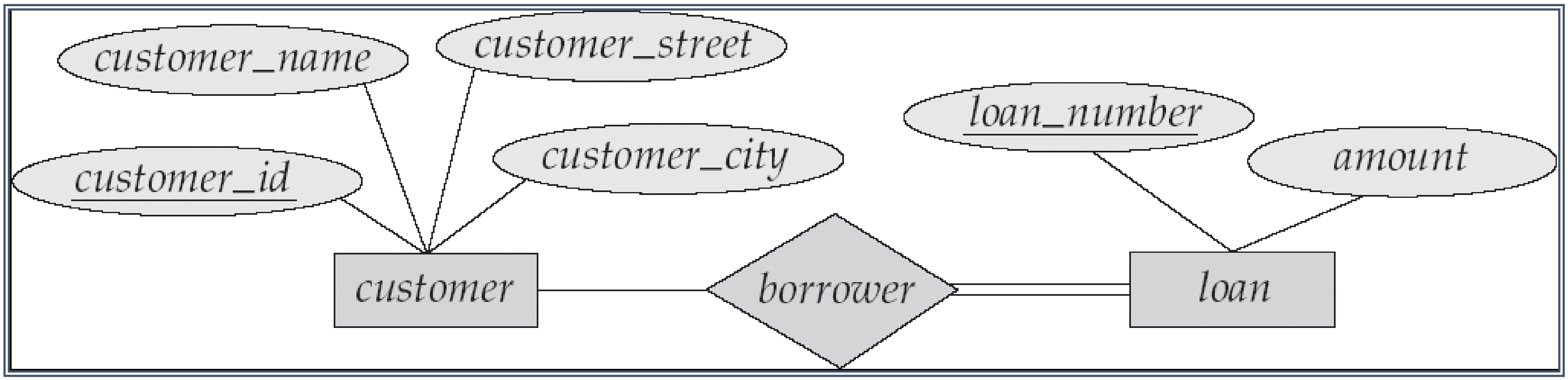
Participation of an Entity Set in a Relationship Set

- **Total participation** (indicated by double line): every entity in the entity set participates in at least one relationship in the relationship set
  - E.g. participation of loan in borrower is total
    - every loan must have a customer associated to it via borrower



Participation of an Entity Set in a Relationship Set

- **Partial participation** (indicated by single line): some entities in the entity set may not participates in any relationship in the relationship set
  - Example: participation of customer in borrower is partial

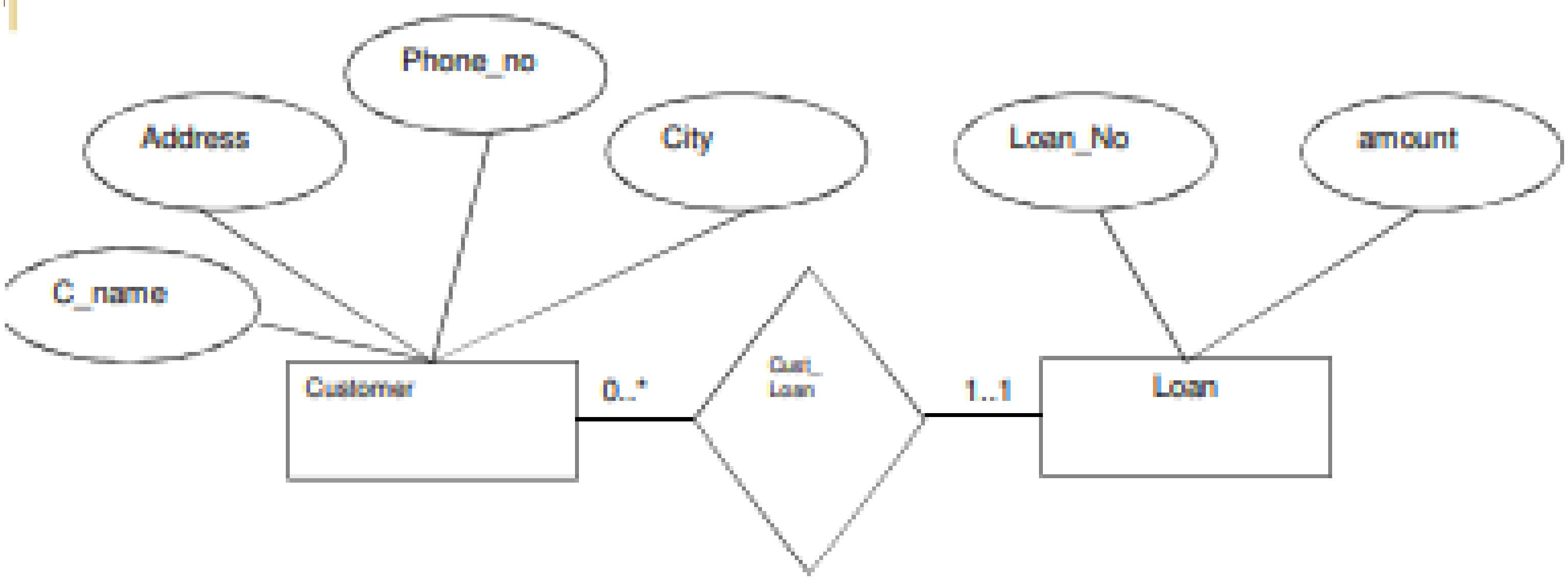


Represenation of Cardinality in ER

The cardinality of a relationship is the actual number of related occurrences for each of the two entities. E-R diagrams also provide a way to indicate cardinality of the relationship. An edge between an entity set and a relationship set can have an associated minimum and maximum cardinality, shown in the form l..h, where l is the minimum and h the maximum cardinality. A minimum value of 1 indicates total participation of the entity set in the relationship set. A maximum value of 1 indicates that the entity participates in at most one relationship, while a maximum value * indicates no limit. The label 1..* on an edge is equivalent to a double line

Represenation of Cardinality in ER

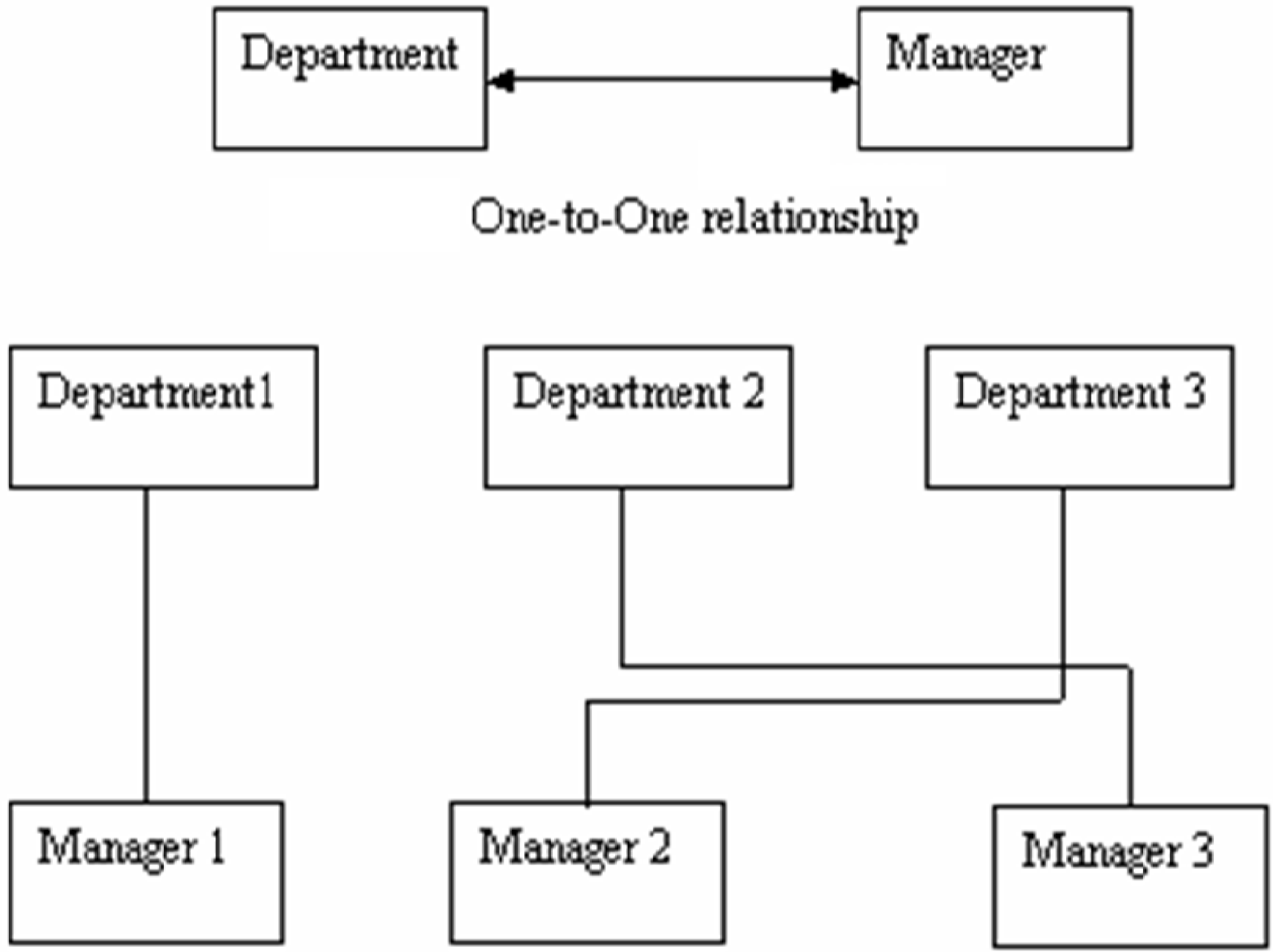
the edge between loan and Cust_Loan has a cardinality constraint of 1..1, meaning the minimum and the maximum cardinality are both 1. That is, each loan must have exactly one associated customer. The limit 0..* on the edge from customer to Cust_Loan indicates that a customer can have zero or more loans. Thus, the relationship Cust_Loan is one to many from customer to loan, and further the participation of loan in Cust_Loan is total.



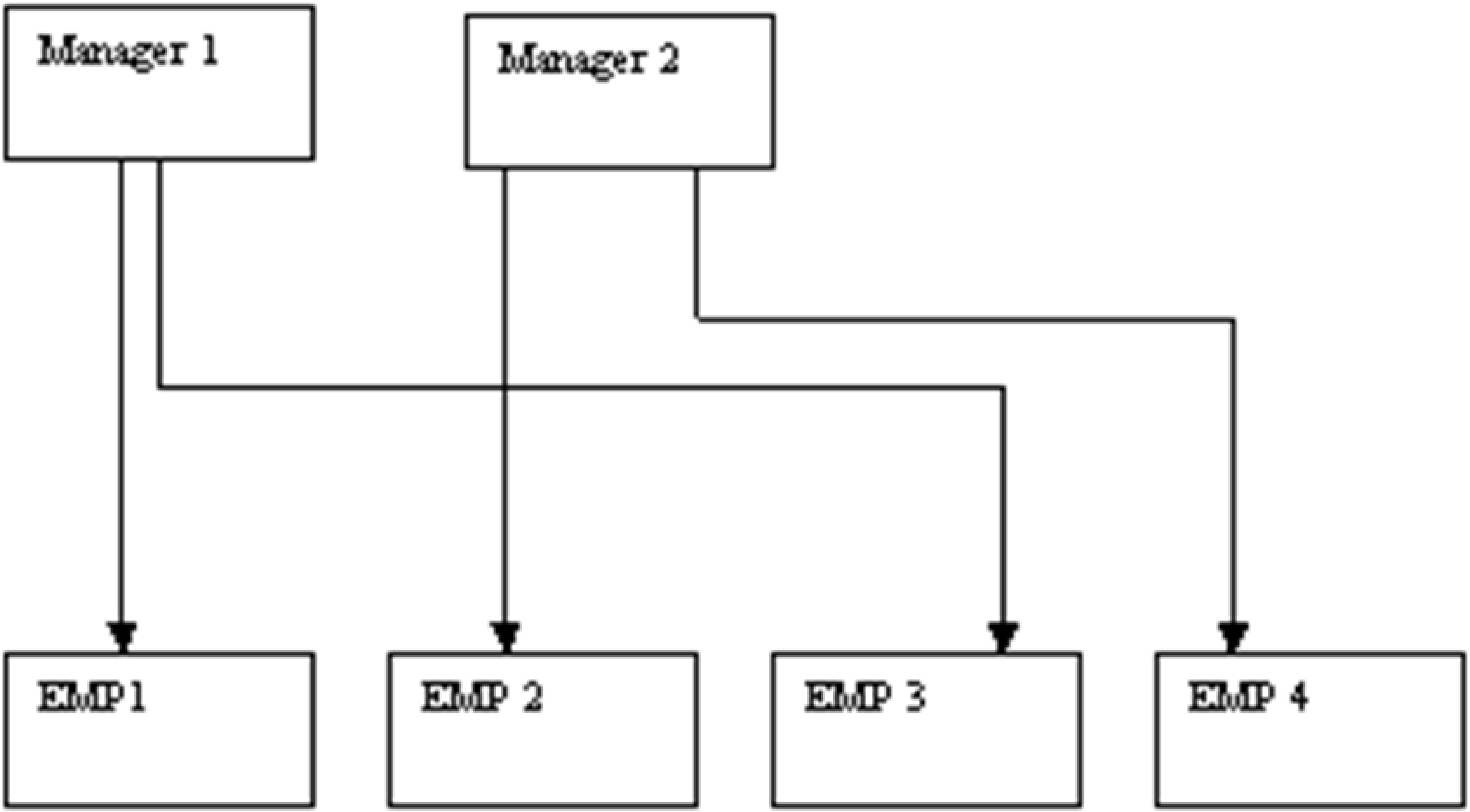
Direction

The direction of a relationship indicates the originating entity of a relationship. The entity from which a relationship originates is the parent entity; the entity where the relationship terminates is the child entity.

The type of the relation is determined by the direction of line connecting relationship component and the entity. To distinguish different types of relation, we draw either a directed line or an undirected line between the relationship set and the entity set. Directed line is used to indicate one occurrence and undirected line is used to indicate many occurrences in a relation as shown in next case.


Department
Manager
One-to-One relationship
Department1
Department 2
Department 3
Manager 1
Manager 2
Manager 3

One-to-Many relationship



E-R Notation

¨ Entities are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.

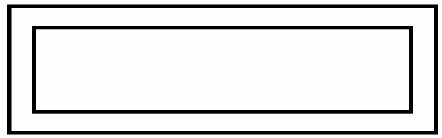
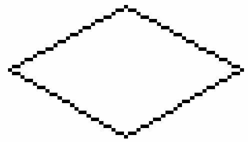
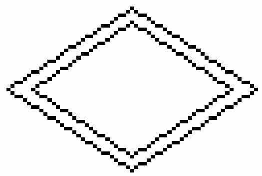
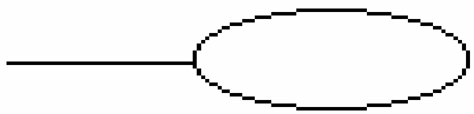
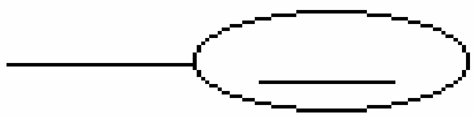
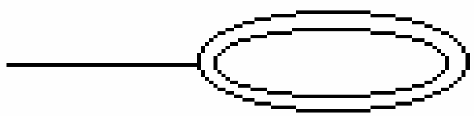
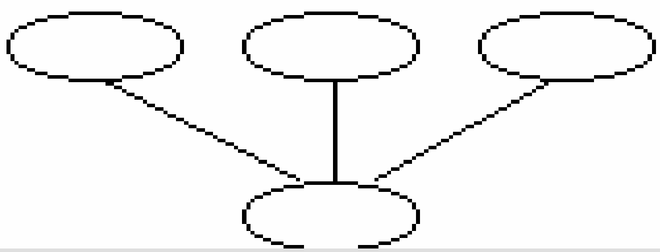
¨ Attributes are represented by Ellipses.

¨ A solid line connecting two entities represents relationships. The name of the relationship is written above the line. Relationship names should be verbs and diamonds sign is used to represent relationship sets.

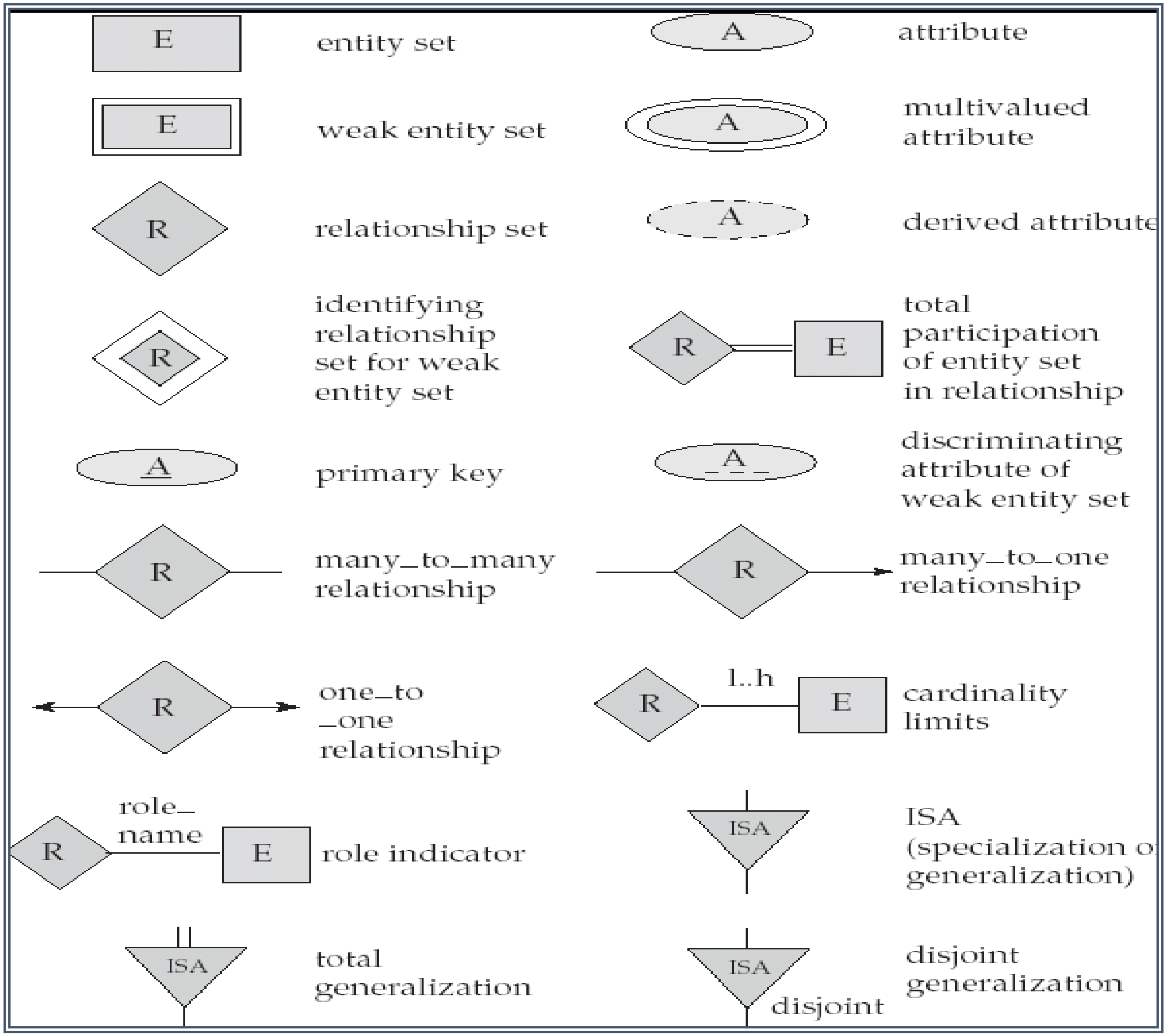
¨ Attributes, when included, are listed inside the entity rectangle. Attributes, which are identifiers, are underlined. Attribute names should be singular nouns.

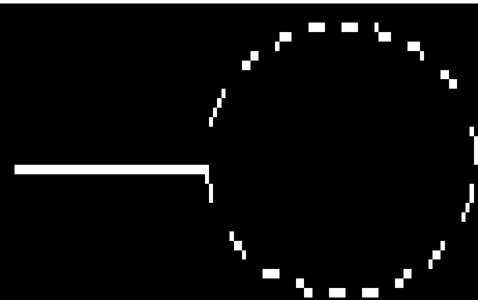
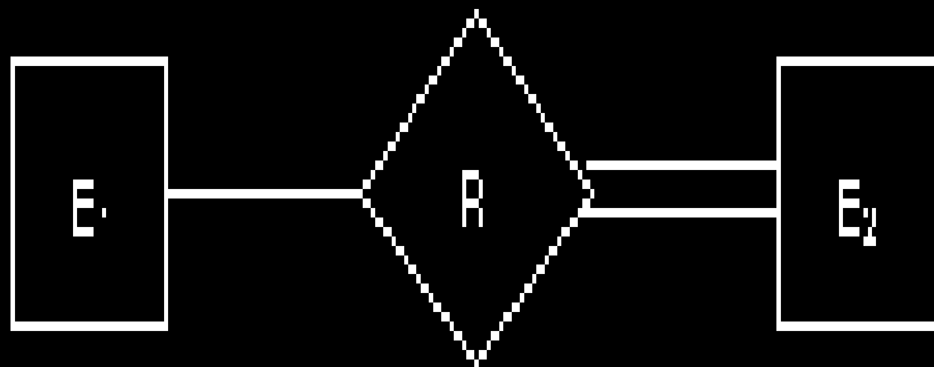
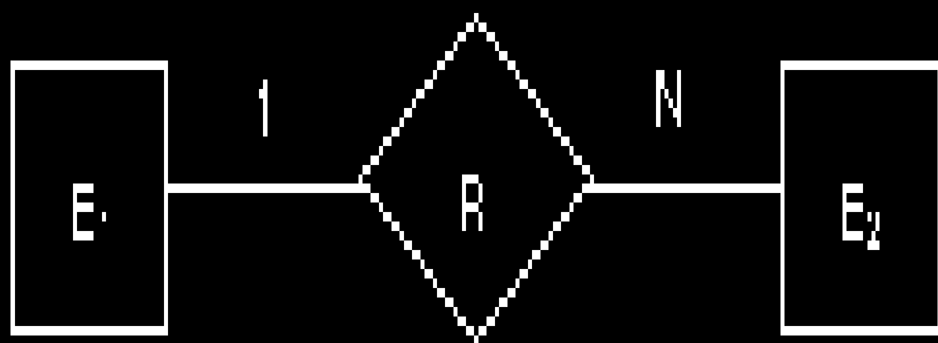
¨ Multi-valued attributes are represented by double ellipses.

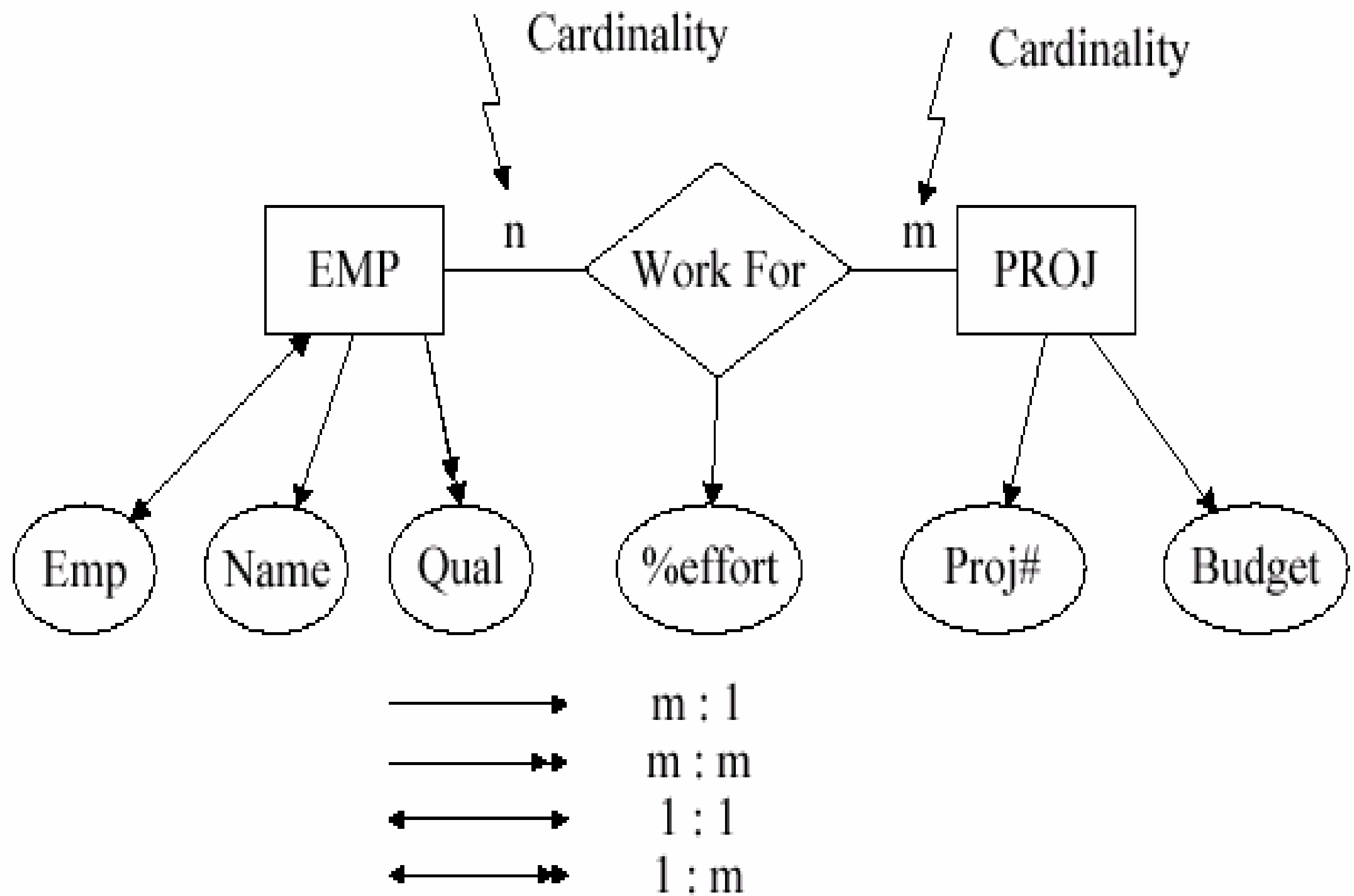
¨ Directed line is used to indicate one occurrence and undirected line is used to indicate many occurrences in a relation.

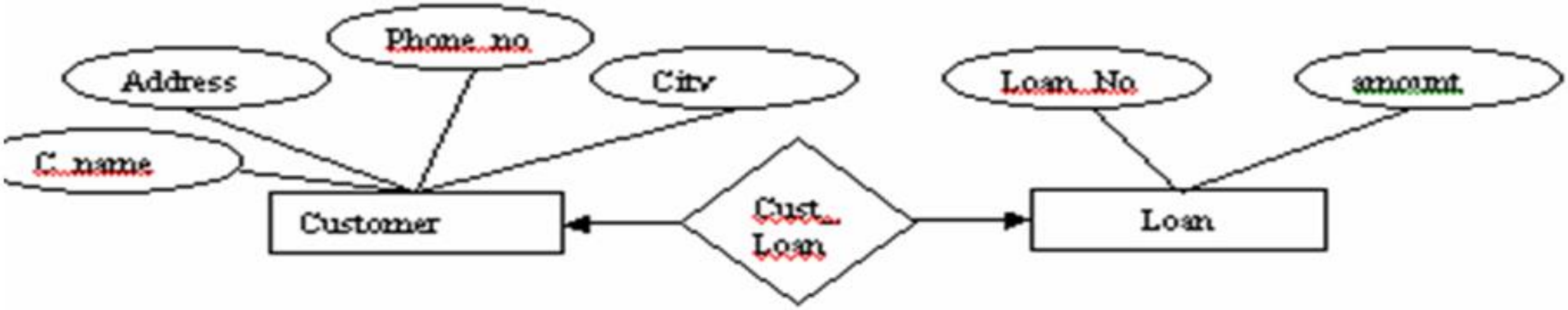
Symbol	Meaning
	Entity Type
	Weak Entity Type
	Relationship Type
	Identifying Relationship Type
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute

Summary of Symbols Used in E-R Notation

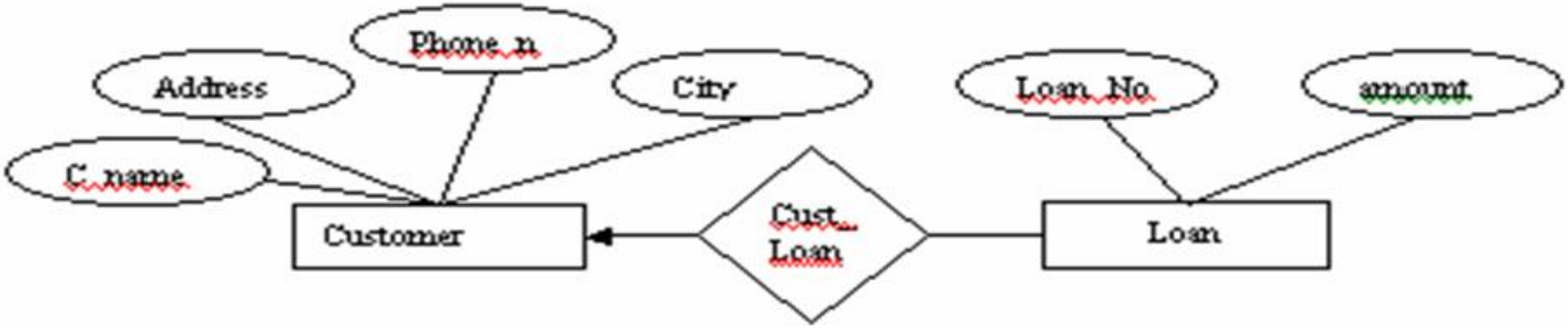


	Derived Attribute
	Total Participation of E2 in R
	Cardinality Relation 1:N for E1:E2 in R

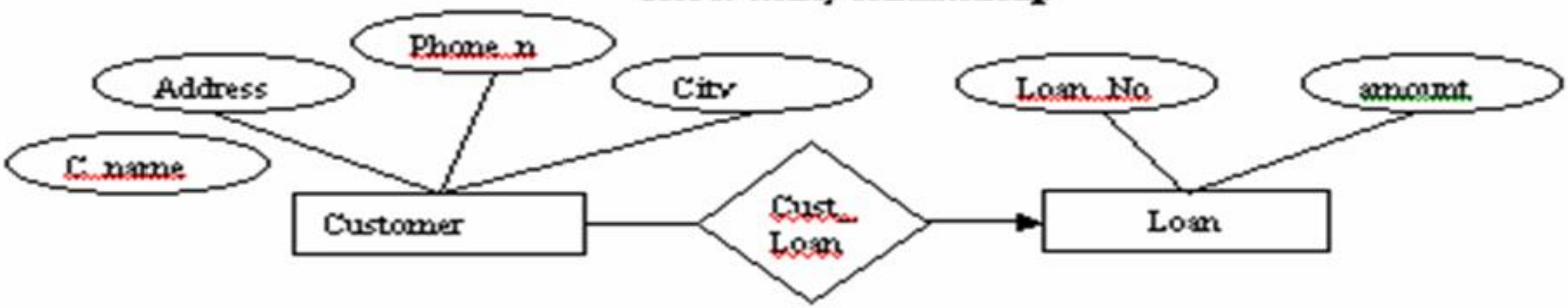

Cardinality
Cardinality
n
m
EMP
Work For
PROJ
Emp
Name
Qual
%effort
Proj#
Budget
m : 1
m : m
1 : 1
1 : m



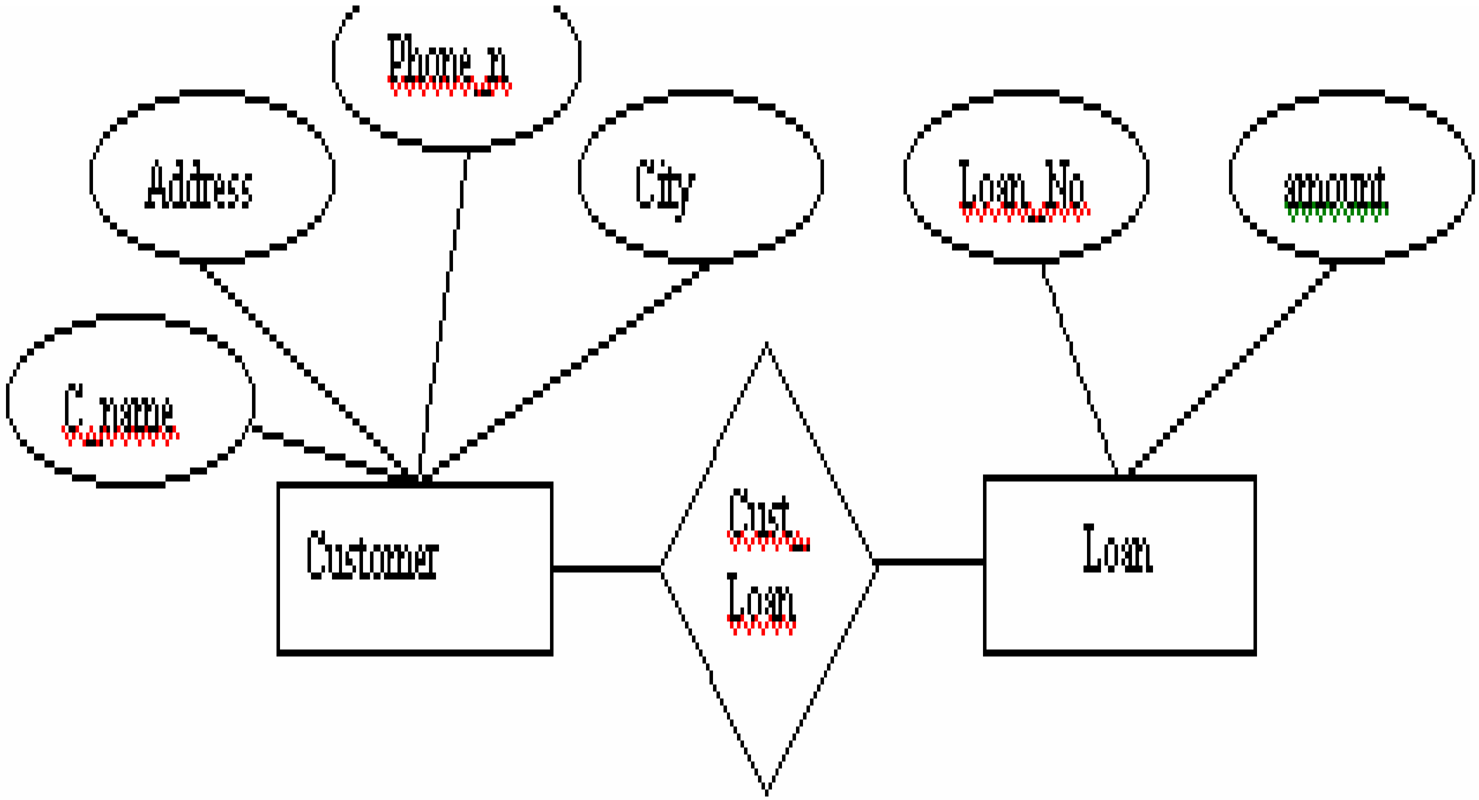
One-to-One Relationship



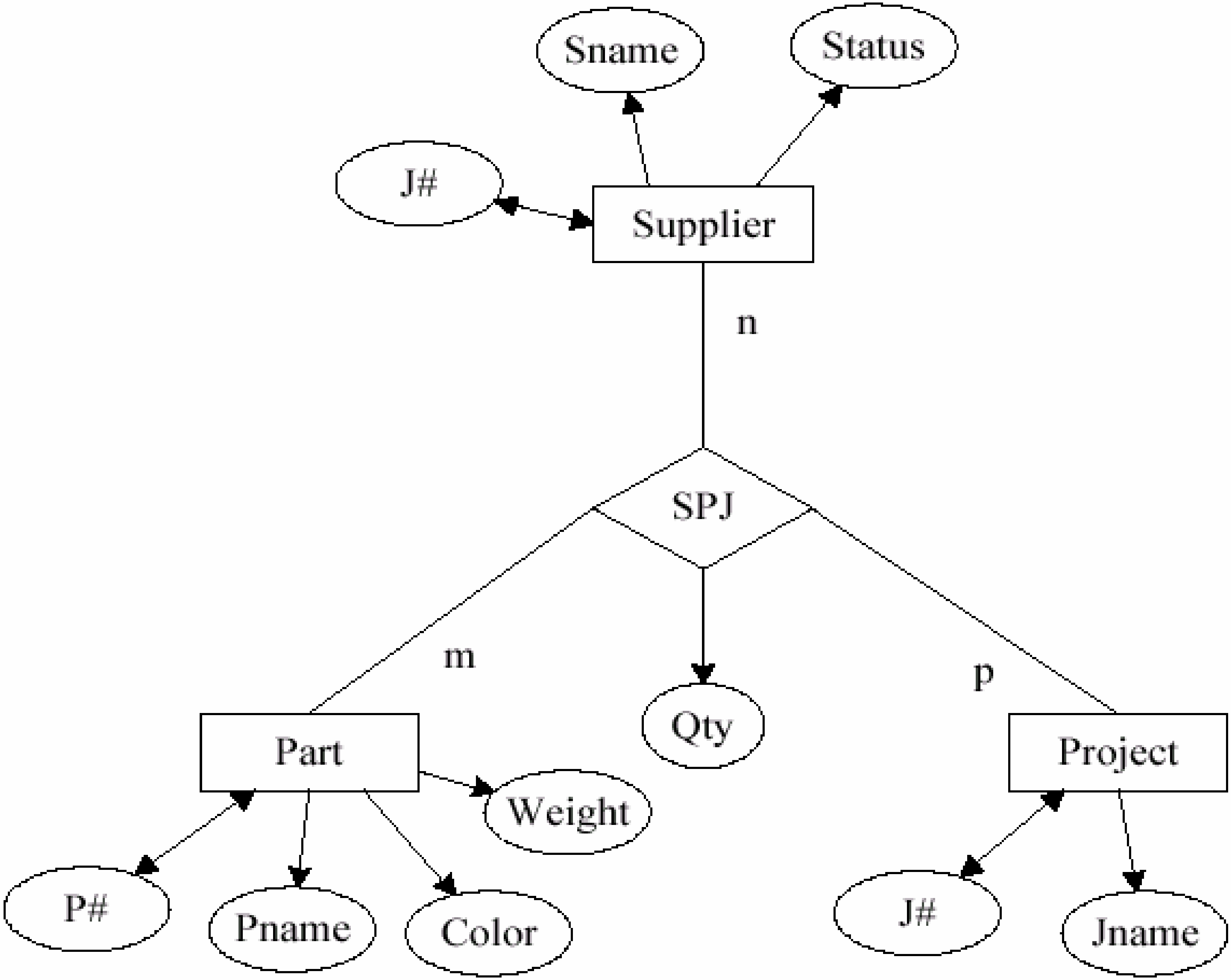
One-to-Many Relationship



Many-to-One Relationship



Many-to-Many Relationship



Sname
Status
J#
Supplier
n
SPJ
m
p
Qty
Part
Project
Weight
P#
Pname
Color
J#
Jname

Strong and Weak Entity Sets

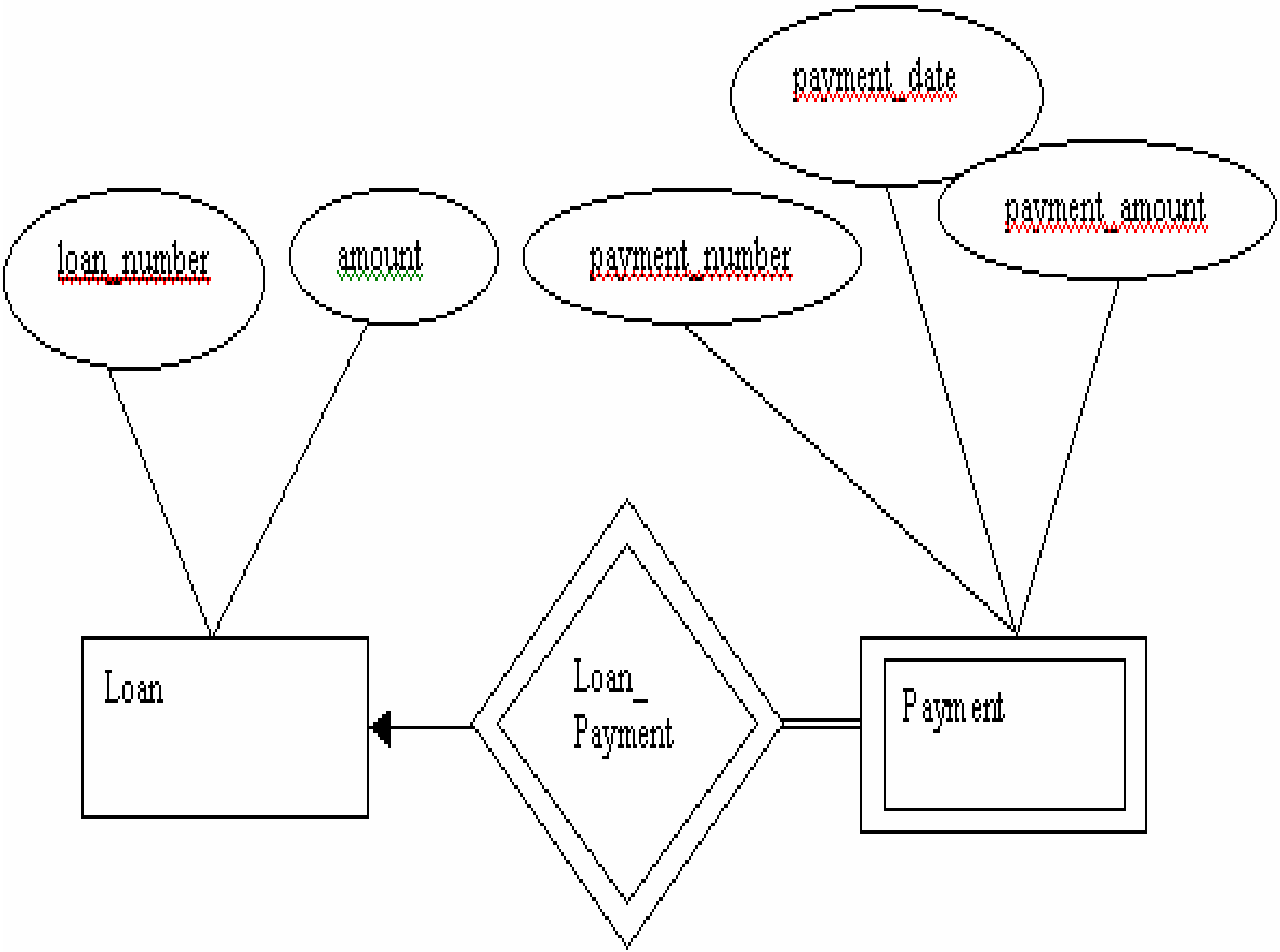
The entity set which does not has sufficient attributes to form a primary key is called as weak entity set.

An entity set that has a primary key is called as Strong entity set.

Consider an entity set Payment which has three attributes: payment_number, payment_date and payment_amount. Although each payment entity is distinct but payment for different loans may share the same payment number. Thus, this entity set does not have a primary key and it is a weak entity set. Each weak set must be a part of one-to-many relationship set.

Strong and Weak Entity Sets

A member of a strong entity set is called dominant entity and member of weak entity set is called as subordinate entity. A weak entity set does not have a primary key but we need a means of distinguishing among all those entries in the entity set that depend on one particular strong entity set. The discriminator of a weak entity set is a set of attributes that allows this distinction to be made. For example, payment_number is acts as discriminator for payment entity set. It is also called as the Partial key of the entity set.



payment_date
payment_amount
loan_number
amount
payment_number
Loan
Loan_
Payment
Payment

Steps to design E-R diagram

1. Identify the Entities
2. To find relationships among these entities
3. To identify the key attributes for every Entity.
4. To identify other relevant attributes
5. To draw the complete e-r diagram with all attributes including primary key

Case Study 1

Consider, a university contains many departments. Each department can offer any number of courses. Many teachers can work in a department. A teacher can work only in one department. For each department there is a Head. A teacher can be head of only one department. Each teacher can take any number of courses. A student can enroll for any number of courses. Each course can have any number of students.

Case Study 1

1. To identify the Entities

University, Department, Course, Teacher, Student

Consider University as single instance

2 Relationship:

Each course belong to one department(1:M)

Department & teacher(1:M)

Department Head and teacher(1:1)

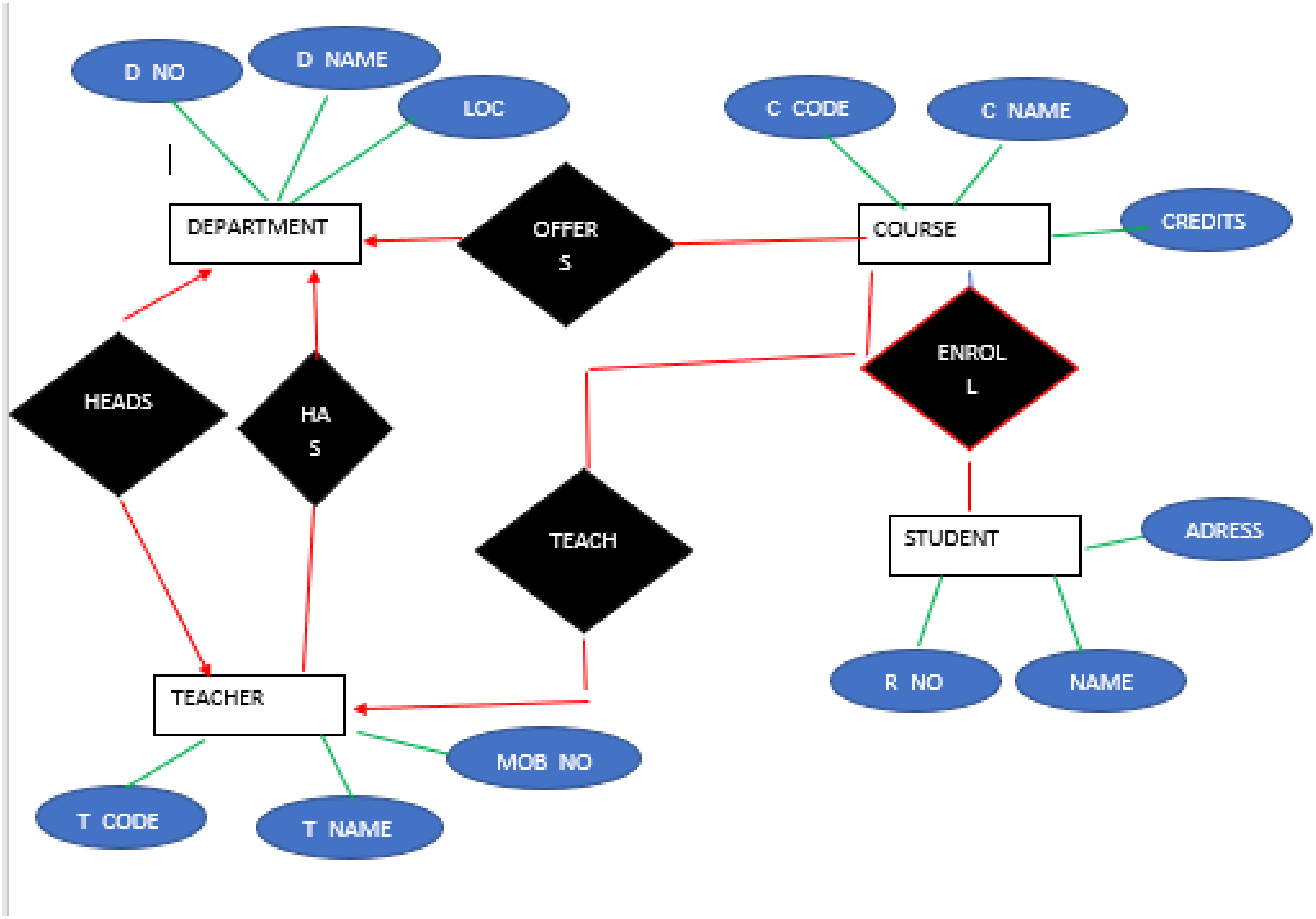
Teacher and course(1:M)

Student & course(M:M)

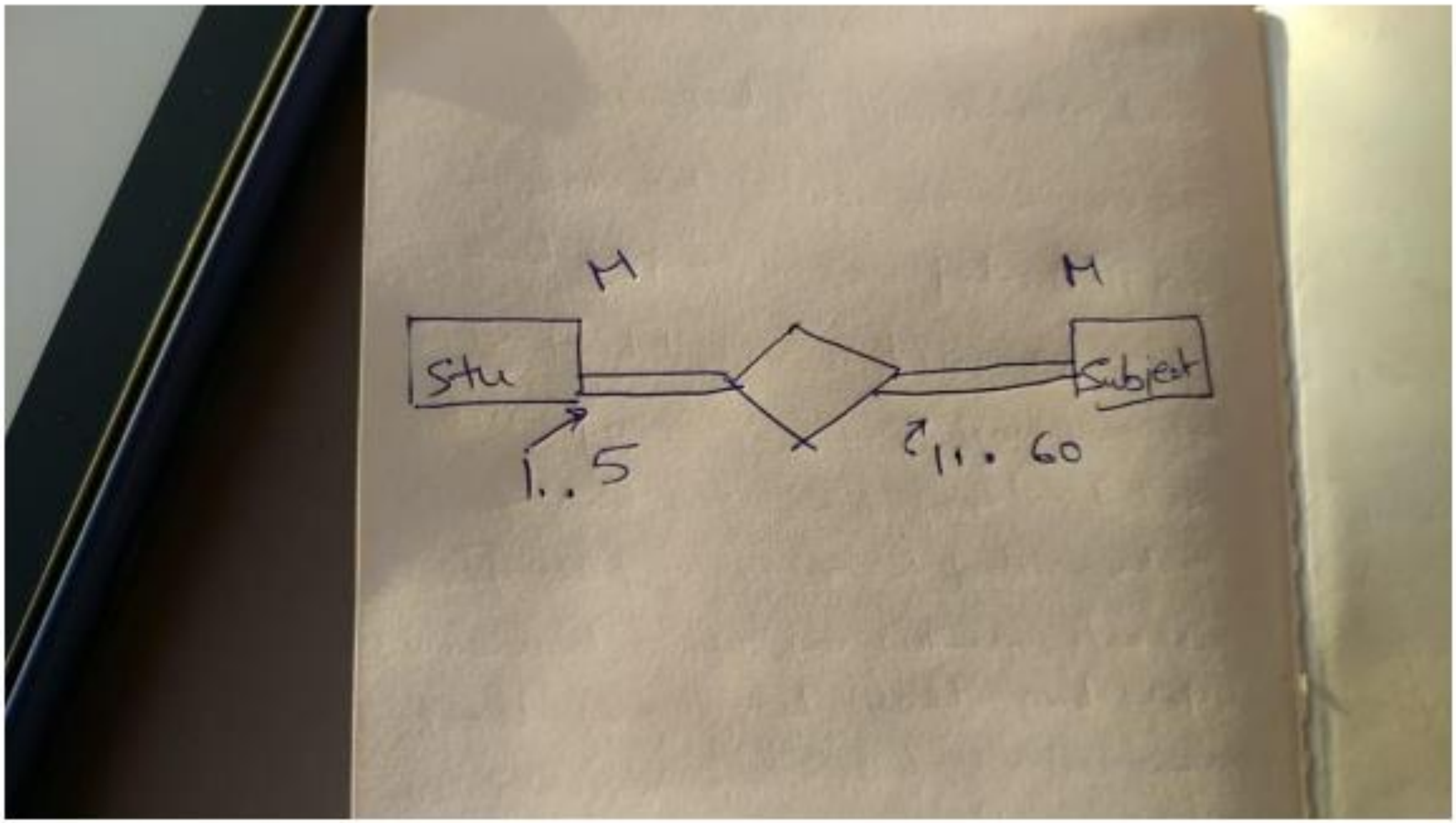
3.Key Attributes:D_no,C_code,Roll_no,t_code

4. Relevant Attributes:d_name,loc,c_name,credits,t_name,mob_no,name,address

Case Study 1 (Solution)



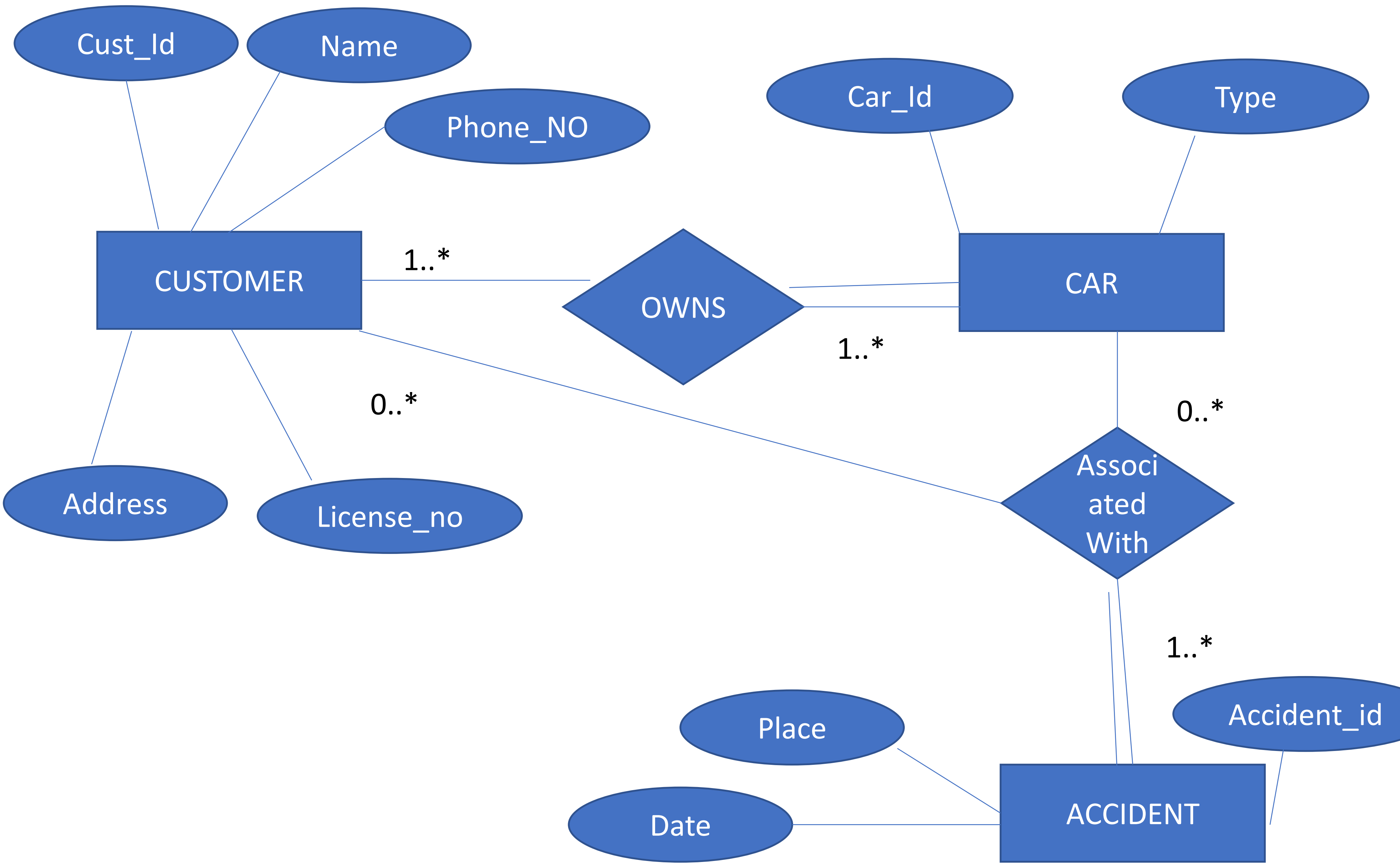
One student can take max of 5 subjects where as one subject can be taken by max of 60 students . Student must take subject and there should be no subject left behind.



Case Study 2

Construct an ER diagram for a car insurance company whose customers own one or more cars each. Each car has associated with it zero to any number of recorded accidents.

Case Study 2 solution



Enhanced ER MODEL

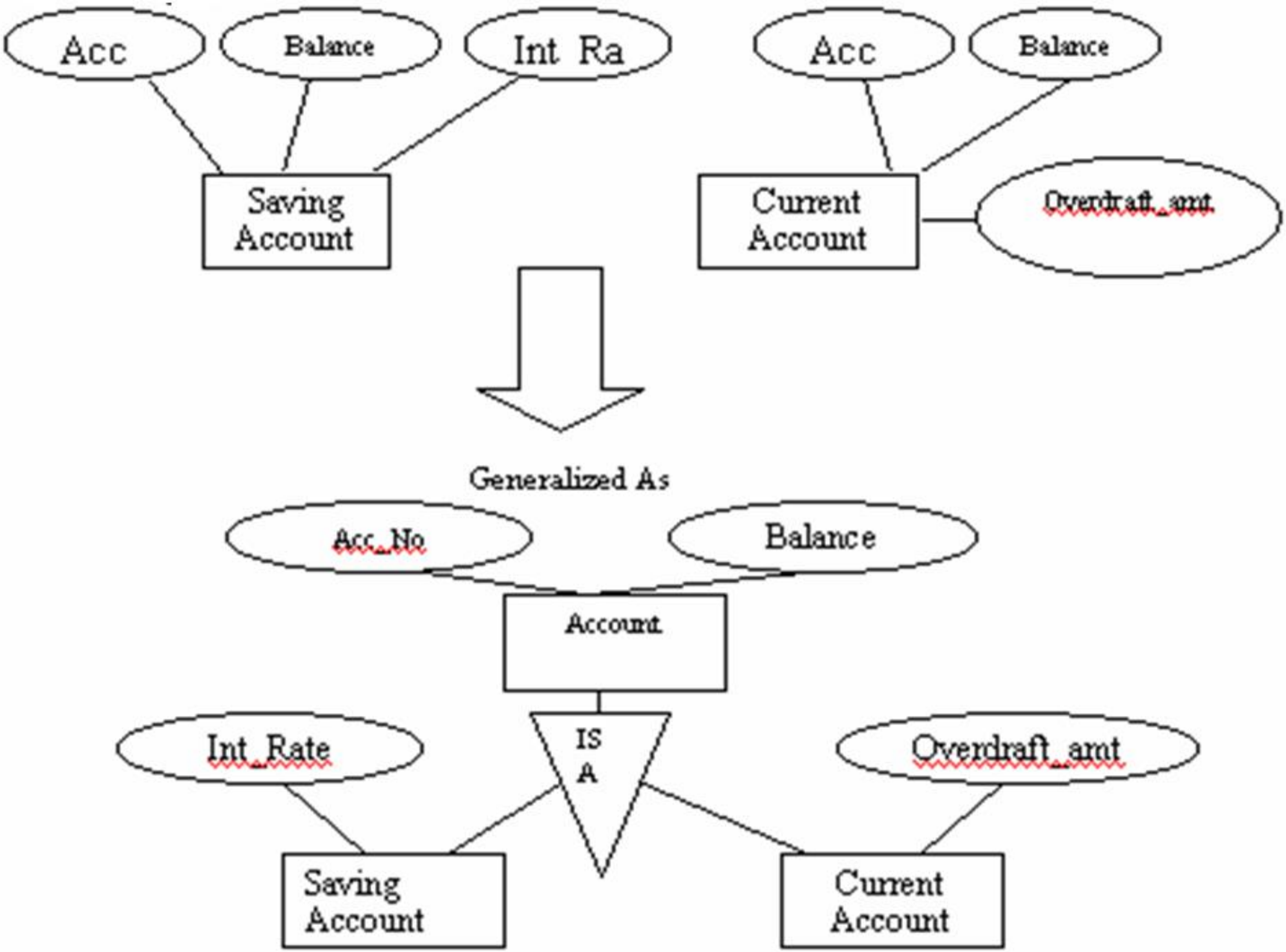
Generalization

A generalization hierarchy is a form of abstraction that specifies that two or more entities that share common attributes can be generalized into a higher-level entity type called a supertype or generic entity. The lower level of entities becomes the subtype, or categories, to the super type. Subtypes are dependent entities.

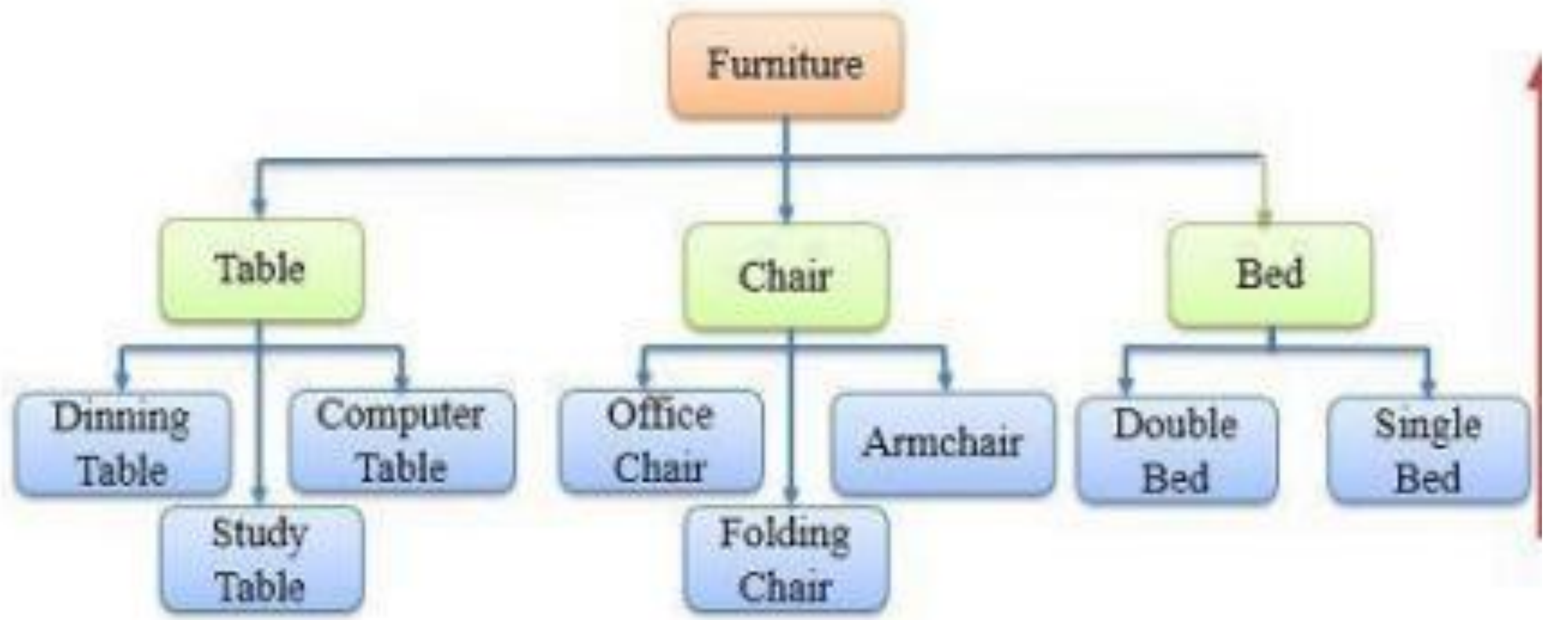
Generalization is used to emphasize the similarities among lower-level entity sets and to hide differences.

It makes ER diagram simpler because shared attributes are not repeated. Generalization is denoted through a triangle component labeled 'IS A",

Bottom Up Approach

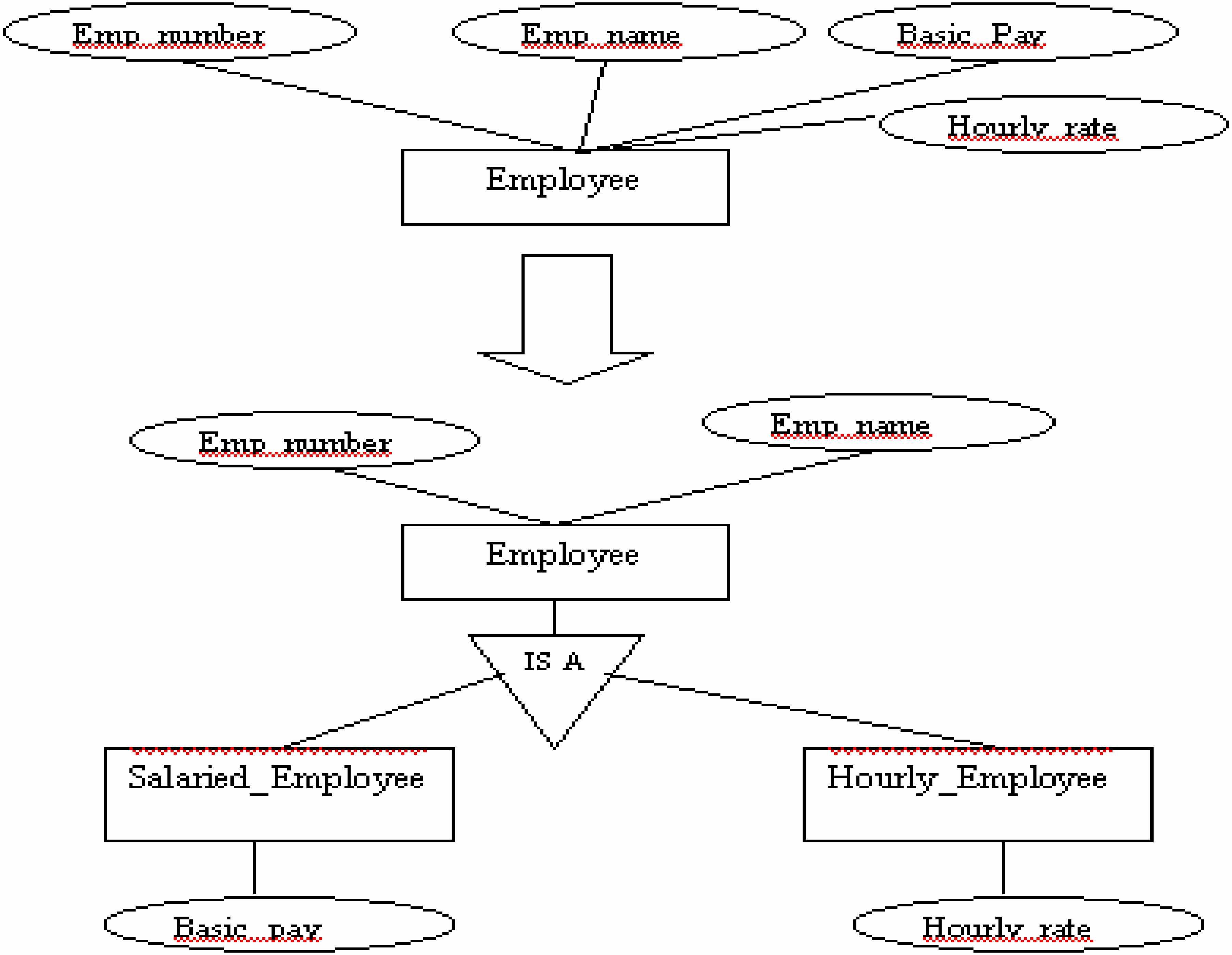


Acc
Balance
Int Ra
Acc
Balance
Saving Account
Current Account
Overdraft_amt
Generalized As
Acc_No
Balance
Account
IS A
Int_Rate
Overdraft_amt
Saving Account
Current Account


Furniture
Table
Chair
Bed
Dinning Table
Computer Table
Study Table
Office Chair
Armchair
Folding Chair
Double Bed
Single Bed

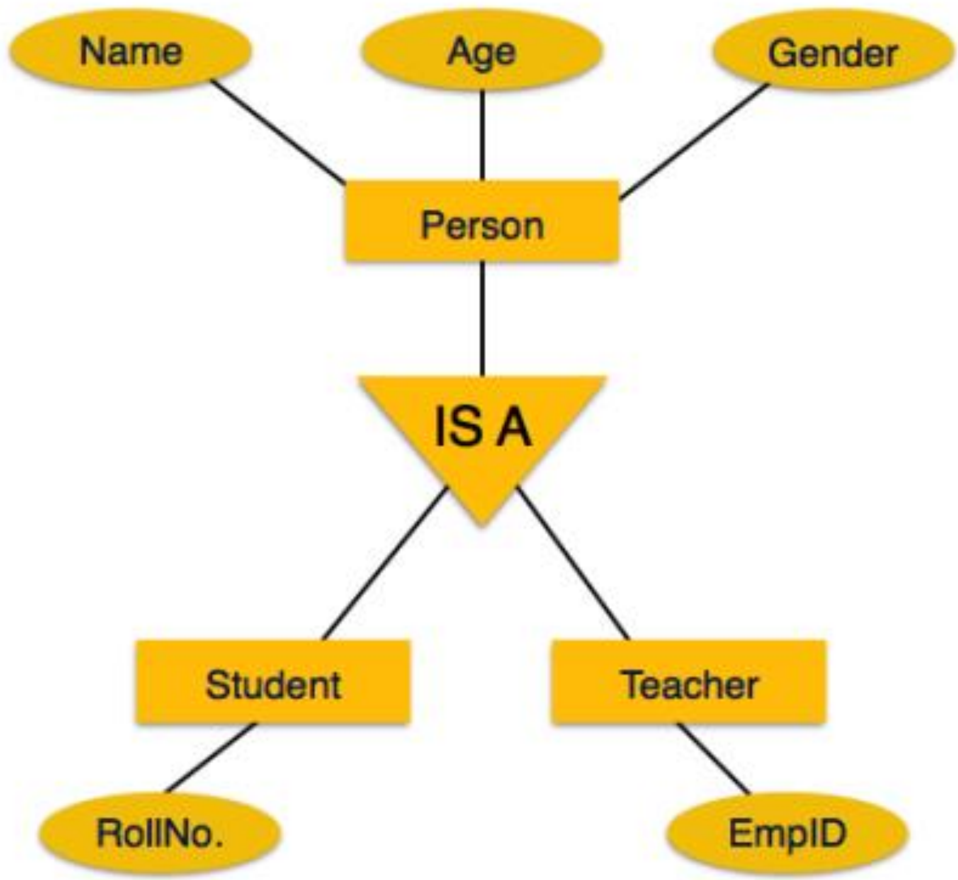
Specialization

- Specialization is the process of taking subsets of a higher-level entity set to form lower level entity sets. It is a process of defining a set of subclasses of an entity type, which is called as superclas of the specialization. The process of defining subclass is based on the basis of some distinguish characteristics of the entities in the super class.
- For example, specialization of the Employee entity type may yield the set of subclasses namely Salaried_Employee and Hourly_Employee on the method of pay
- Top Down Approach



Emp_number
Emp_name
Basic_Pay
Hourly_rate
Employee
Emp_number
Emp_name
Employee
IS A
Salaried_Employee
Hourly_Employee
Basic_pay
Hourly_rate

Inheritance is an important feature of Generalization and Specialization. It allows lower-level entities to inherit the attributes of higher-level entities.



For example, the attributes of a Person class such as name, age, and gender can be inherited by lower-level entities such as Student or Teacher.

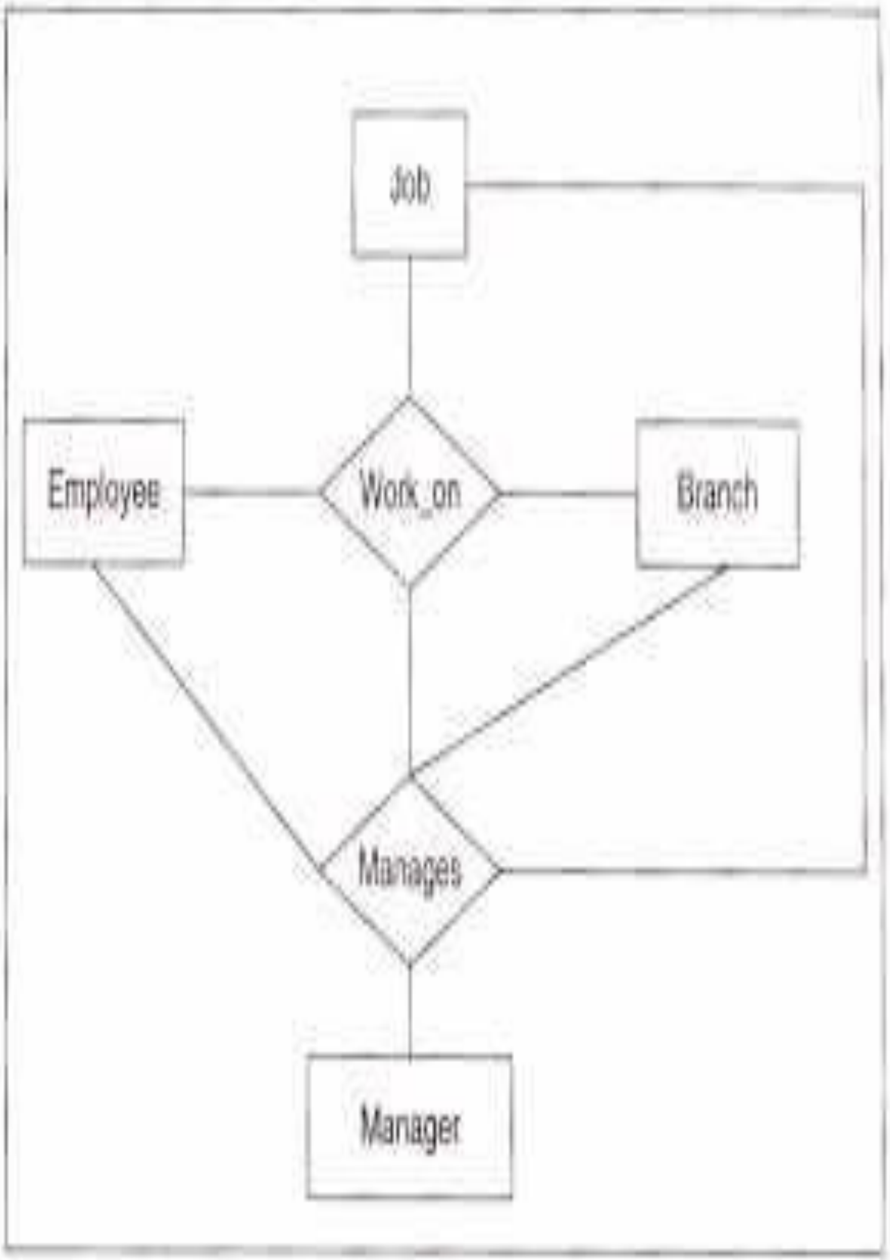
Difference between Specialization and Generalization

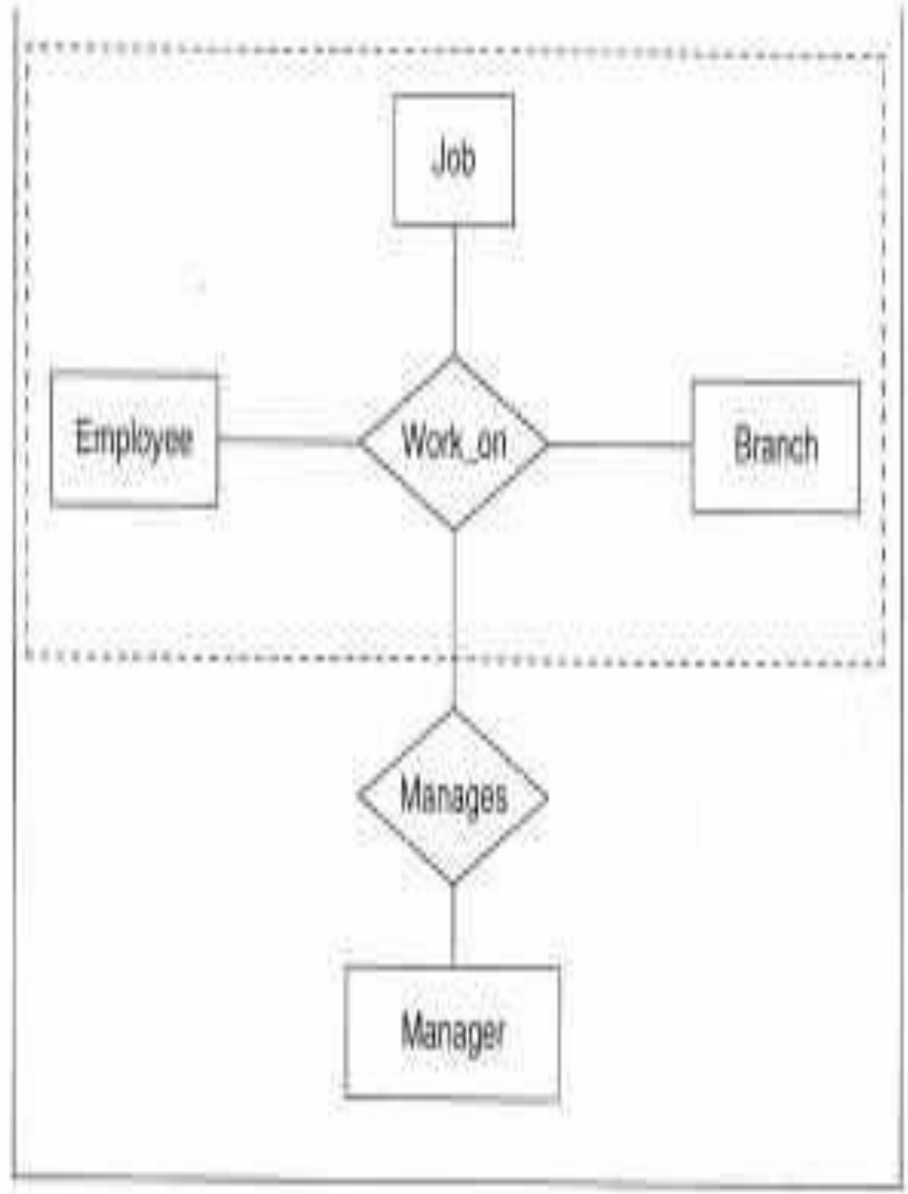
1.Generalization is a bottom-up approach. However, specialization is a top-down approach.
2. Generalization club all the entities that share some common properties to form a new entity. On the other hands, specialization spilt an entity to form multiple new entities that inherit some properties of the spiltted entity.
3. In generalization, a higher entity must have some lower entities whereas, in specialization, a higher entity may not have any lower entity present.
4. Generalization helps in reducing the size of schema whereas, specialization is just opposite it increases the number of entities thereby increasing the size of a schema.
5. Generalization is always applied to the group of entities whereas, specialization is always applied on a single entity.
6. Generalization results in a formation of a single entity whereas, Specialization results in the formation of multiple new

Aggregation

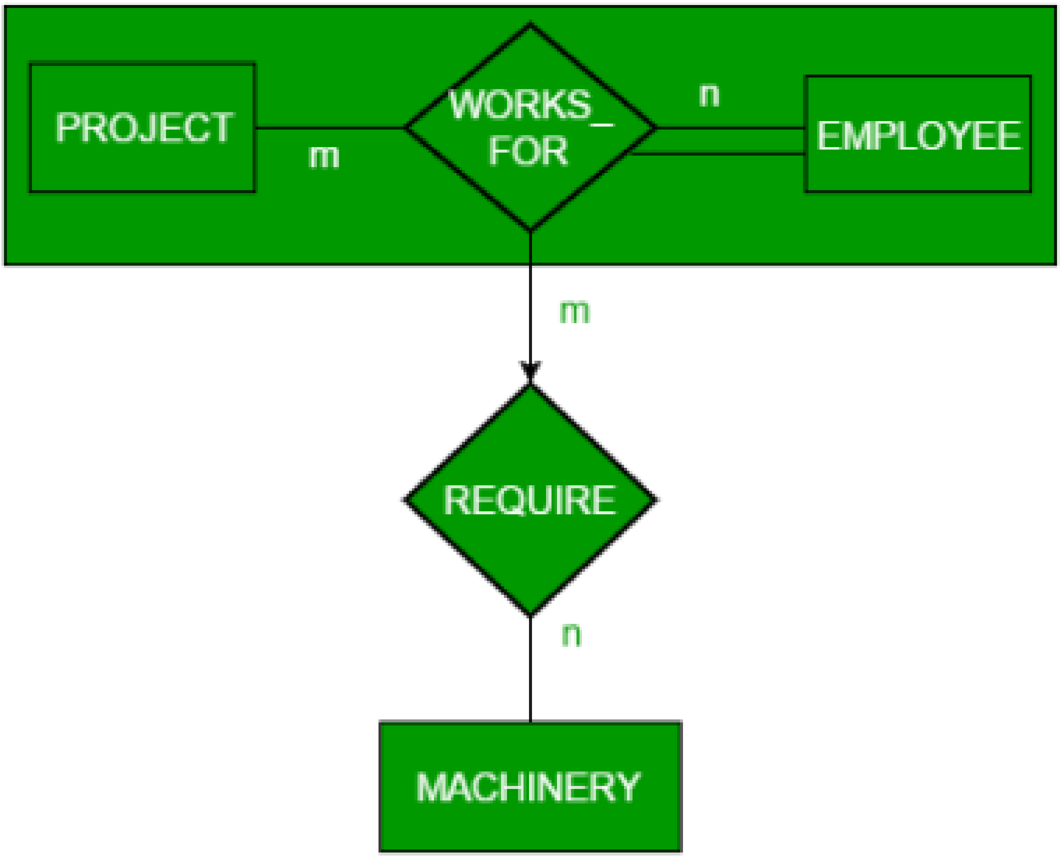
One limitation of the E-R model is that it cannot express relationships among relationships.

The best way to model a situation like this is by the use of aggregation. Thus, the relationship set work_on relating the entity sets Employee, Branch and Job is a higher-level entity set. Such an entity set is treated in the same manner, as is any other entity set. We can then create a binary relationship Manages between work_on and Manager to represent who manages what tasks.


Job
Employee
Work_on
Branch
Manages
Manager


Job
Employee
Work_on
Branch
Manages
Manager

Aggregation



Representing aggregation via schema- To represent aggregation, create a schema containing:

1. primary key of the aggregated relationship
2. primary key of the associated entity set
3. descriptive attribute, if exists.

Thank
You

CONVERSION FROM ER to TABLE

Dr. Geeta Kasana
Assistant Professor
CSED
Thapar Institute of Engineering & Technology

CONVERSION FROM ER to TABLE

- For each entity set and relationship set there is a unique table, which is assigned the name of the corresponding entity set or relationship set.

- Each table has a number of columns (generally corresponding to attributes), which have unique names.

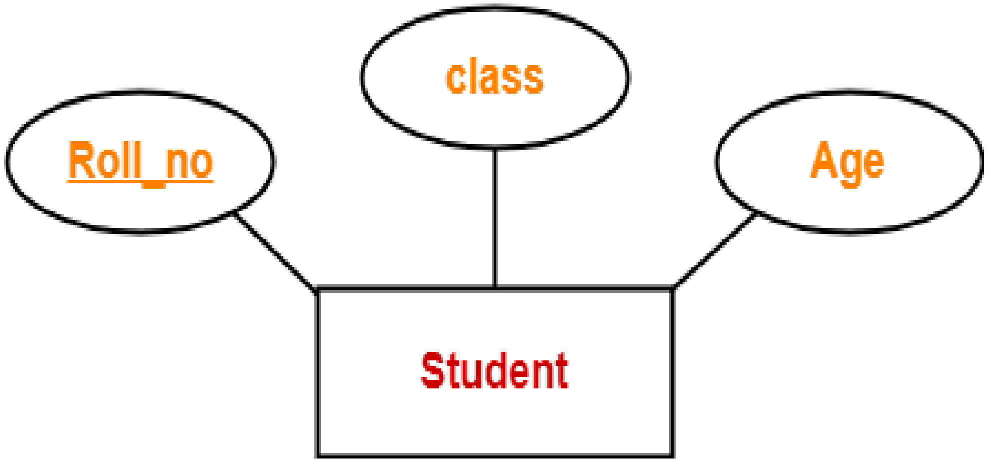
- Primary keys allow entity sets and relationship sets to be expressed uniformly as tables, which represent the contents of the database

CONVERSION FROM ER to TABLE

Rule-01: For Strong Entity Set With Only Simple Attributes

A strong entity set with only simple attributes will require only one table in relational model.

- Attributes of the table will be the attributes of the entity set.
- The primary key of the table will be the key attribute of the entity set

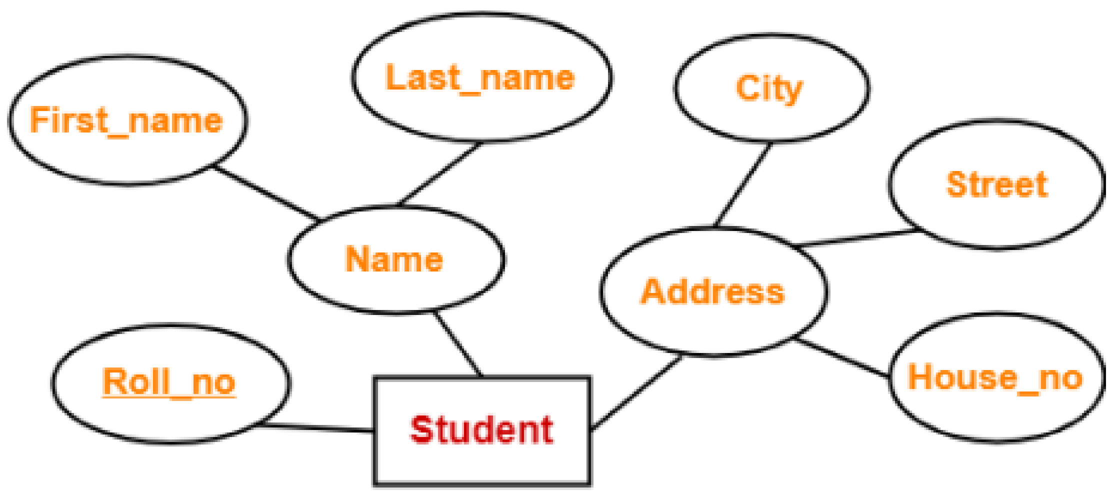


Schema : Student (Roll_no , Class , Age)

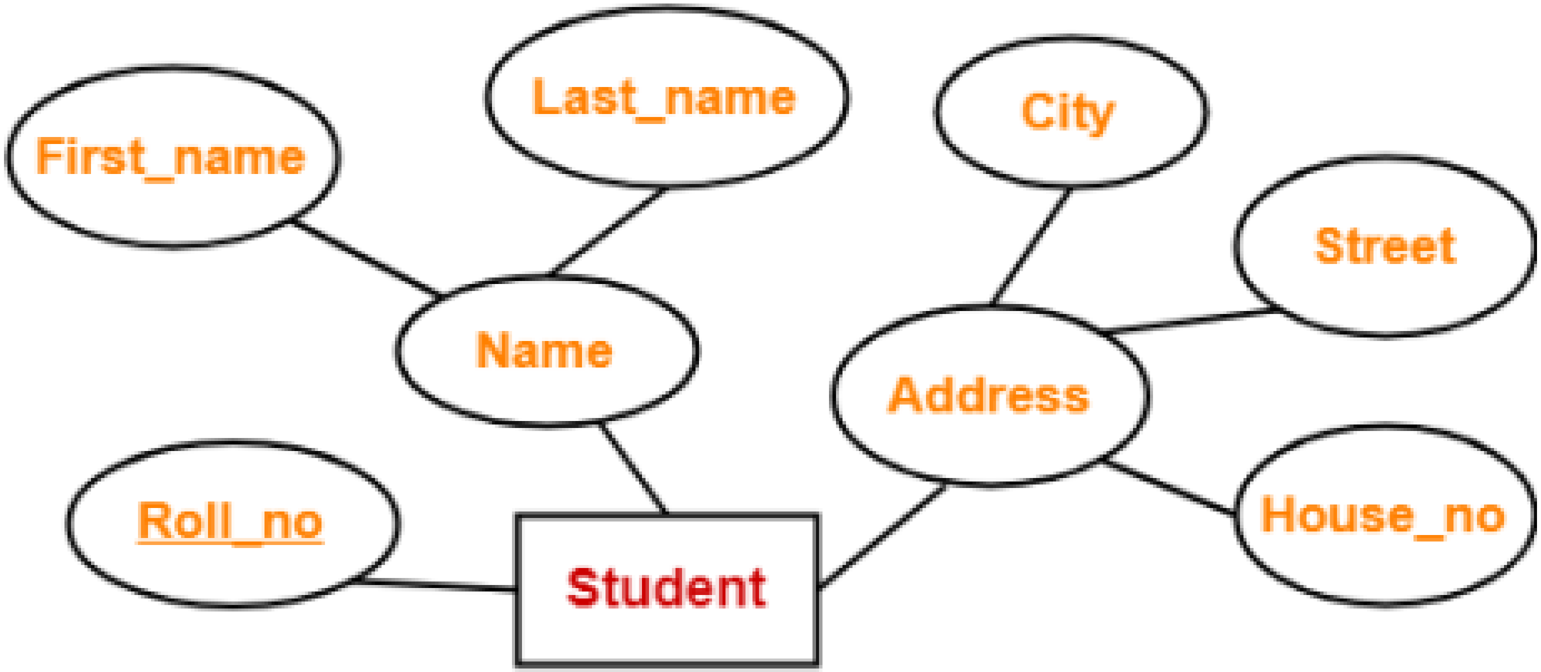
CONVERSION FROM ER to TABLE

Rule-02: For Strong Entity Set With Composite Attributes-

- A strong entity set with any number of composite attributes will require only one table in relational model.
- While conversion, simple attributes of the composite attributes are taken into account and not the composite attribute itself.



CONVERSION FROM ER to TABLE



Roll_no	First_name	Last_name	House_no	Street	City

Schema : Student (Roll_no , First_name , Last_name , House_no , Street , City)

CONVERSION FROM ER to TABLE

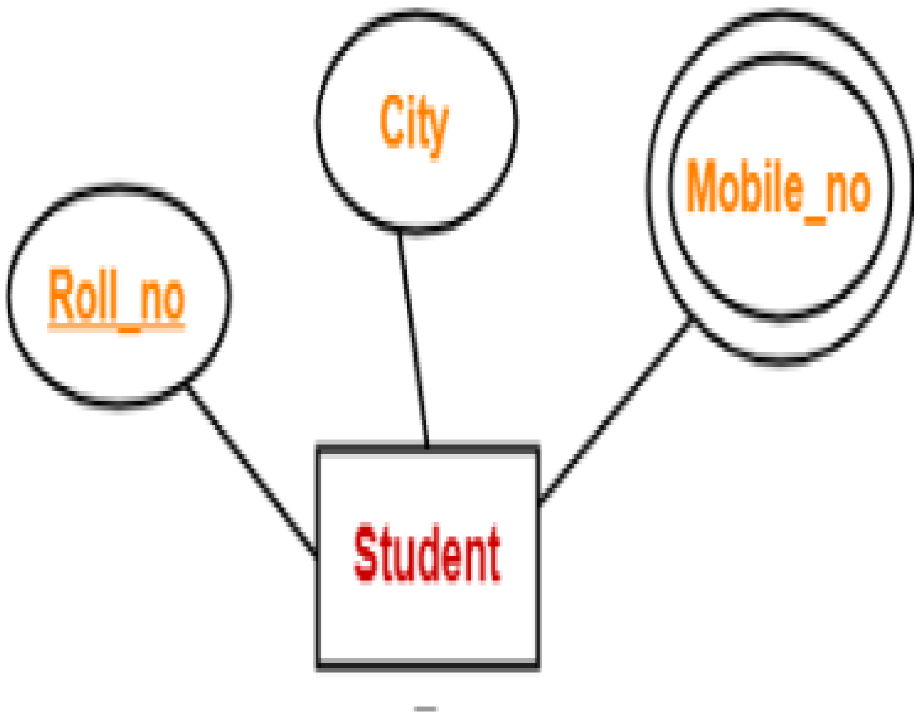
Rule-03: For Strong Entity Set With Multi Valued Attributes

A strong entity set with any number of multi valued attributes will require two tables in relational model.

- One table will contain all the simple attributes with the primary key.
- Other table will contain the primary key and all the multi valued attributes.

CONVERSION FROM ER to TABLE

Rule-03: For Strong Entity Set With Multi Valued Attributes



Roll_no	City

Roll_no	Mobile_no

CONVERSION FROM ER to TABLE

Rule-04: Translating Relationship Set into a Table

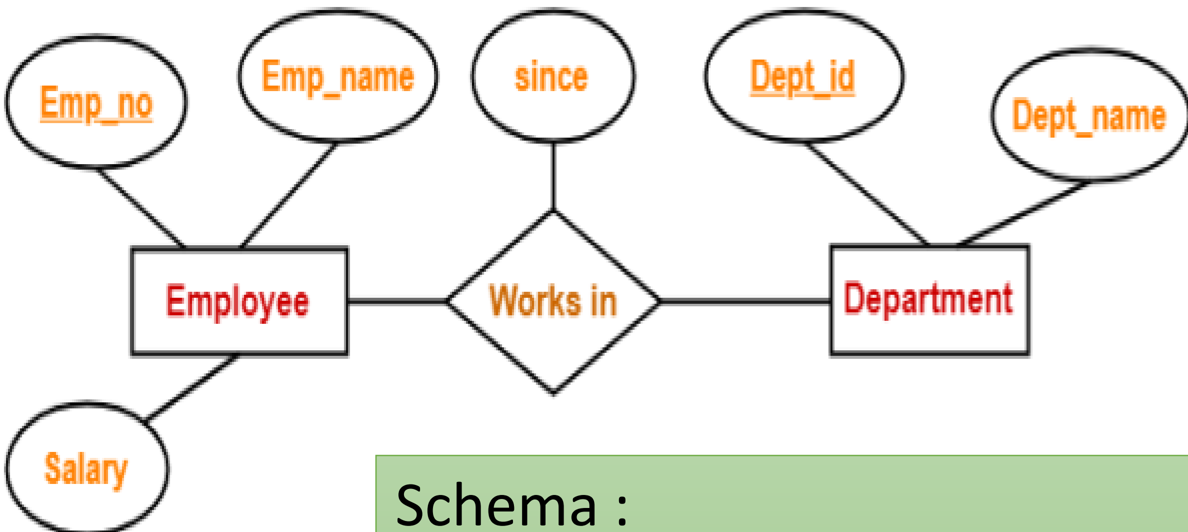
A relationship set will require one table in the relational model.

Attributes of the table are-

- Primary key attributes of the participating entity sets
- Its own descriptive attributes if any. Set of non-descriptive attributes will be the primary key

CONVERSION FROM ER to TABLE

Rule-04: Translating Relationship Set into a Table



Schema :

Works in (Emp_no , Dept_id , since)

Employee(emp_no,name.salary)

Department(id,name)

CONVERSION FROM ER to TABLE

Rule-05: For Binary Relationships With Cardinality Ratios

The following four cases are possible

Case-01: Binary relationship with cardinality ratio m:n

Case-02: Binary relationship with cardinality ratio 1:n

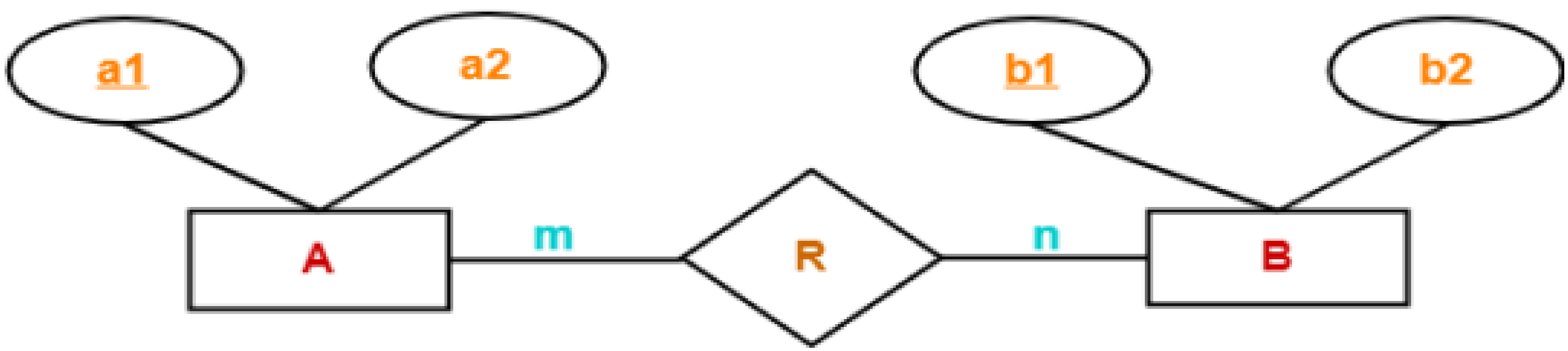
Case-03: Binary relationship with cardinality ratio m:1

Case-04: Binary relationship with cardinality ratio 1:1

CONVERSION FROM ER to TABLE

Rule-05: For Binary Relationships With Cardinality Ratios

Case-01: For Binary Relationship With Cardinality Ratio m:n



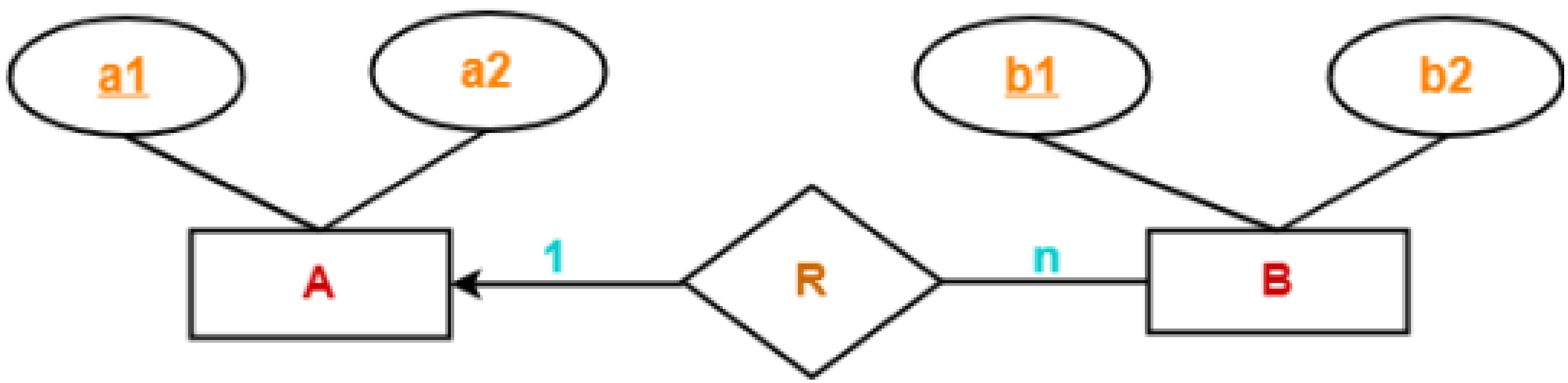
Here, three tables will be required-

1. A (<u>a1</u> , a2)
2. R (<u>a1</u> , <u>b1</u>)
3. B (<u>b1</u> , b2)

CONVERSION FROM ER to TABLE

Rule-05: For Binary Relationships With Cardinality Ratios

Case-02: For Binary Relationship With Cardinality Ratio 1:n



Here, two tables will be required-

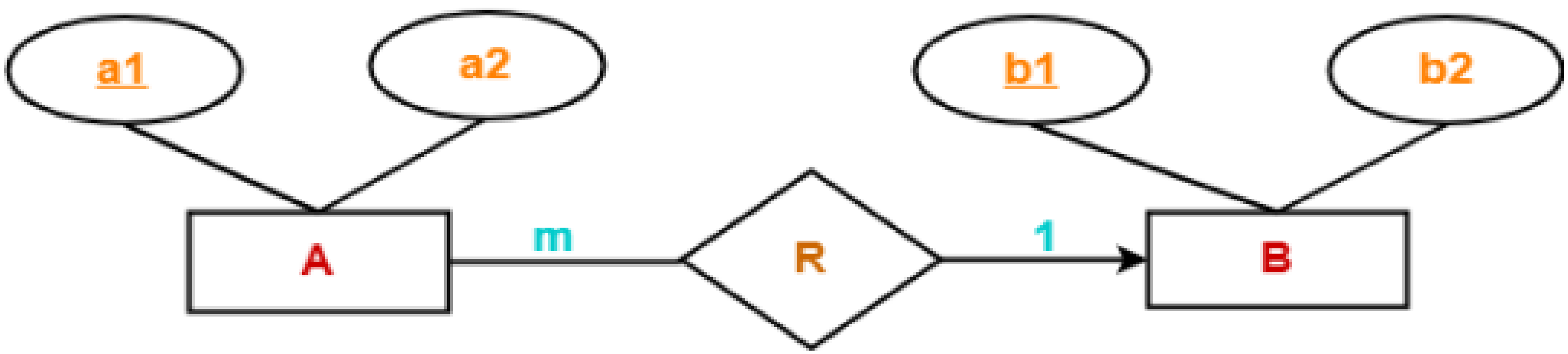
1. A (a1 , a2)
2. BR (a1 , b1 , b2)

NOTE- Here, combined table will be drawn for the entity set B and relationship set R.

CONVERSION FROM ER to TABLE

Rule-05: For Binary Relationships With Cardinality Ratios

Case-03: Binary relationship with cardinality ratio m:1



Here, two tables will be required-

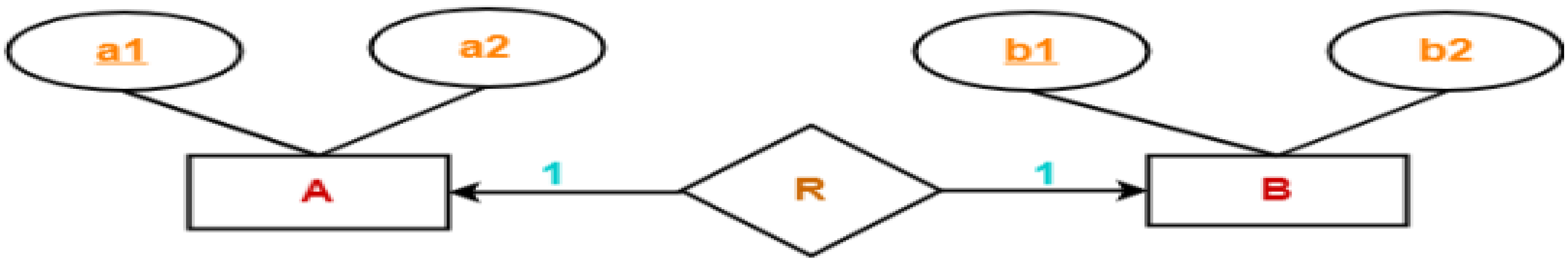
1. AR (a1 , a2 , b1)
2. B (b1 , b2)

NOTE- Here, combined table will be drawn for the entity set A and relationship set R.

CONVERSION FROM ER to TABLE

Rule-05: For Binary Relationships With Cardinality Ratios

Case-04: For Binary Relationship With Cardinality Ratio 1:1



Here, two tables will be required. Either combine 'R' with 'A' or 'B'

Way-01:

1. AR (a1 , a2 , b1)
2. B (b1 , b2)

Way-02:

1. A (a1 , a2)
2. BR (a1 , b1 , b2)

CONVERSION FROM ER to TABLE

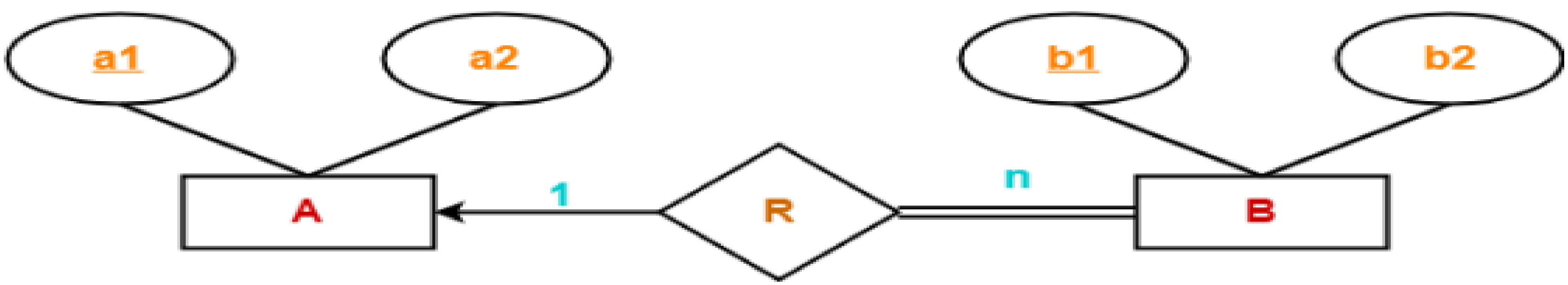
Rule-06: For Binary Relationship With Both Cardinality Constraints and Participation Constraints-

- Cardinality constraints will be implemented as discussed in Rule-05.
- Because of the total participation constraint, foreign key acquires NOT NULL constraint i.e. now foreign key can not be null.

CONVERSION FROM ER to TABLE

Rule-06: For Binary Relationship With Both Cardinality Constraints and Participation Constraints-

Case-01: For Binary Relationship With Cardinality Constraint and Total Participation Constraint From One Side-



Because cardinality ratio = 1 : n , so we will combine the entity set B and relationship set R.

Then, two tables will be required-

1. A (a1 , a2)
2. BR (a1 , b1 , b2)

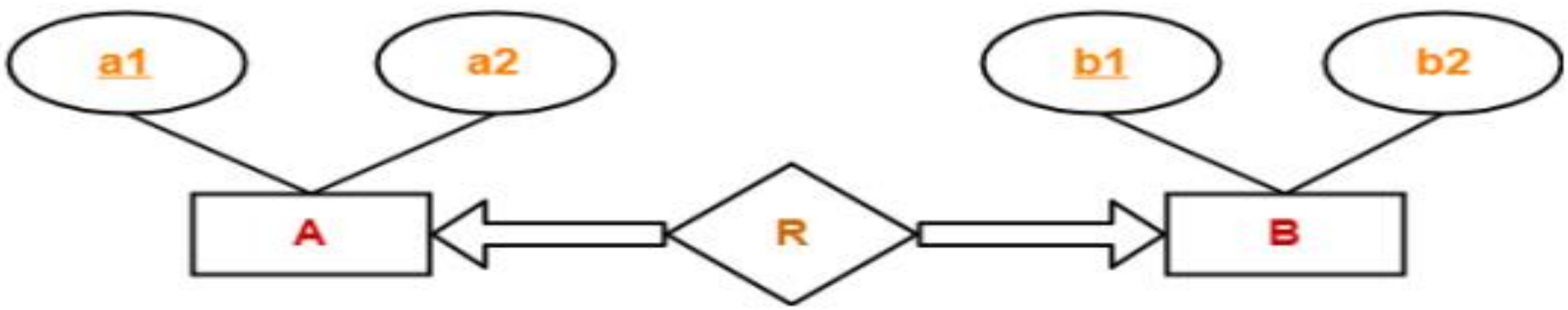
Because of total participation, foreign key a1 has acquired NOT NULL constraint, so it can't be null now.

CONVERSION FROM ER to TABLE

Rule-06: For Binary Relationship With Both Cardinality Constraints and Participation Constraints-
only one table is required

Case-02: For Binary Relationship With Cardinality Constraint and Total Participation Constraint From Both Sides-

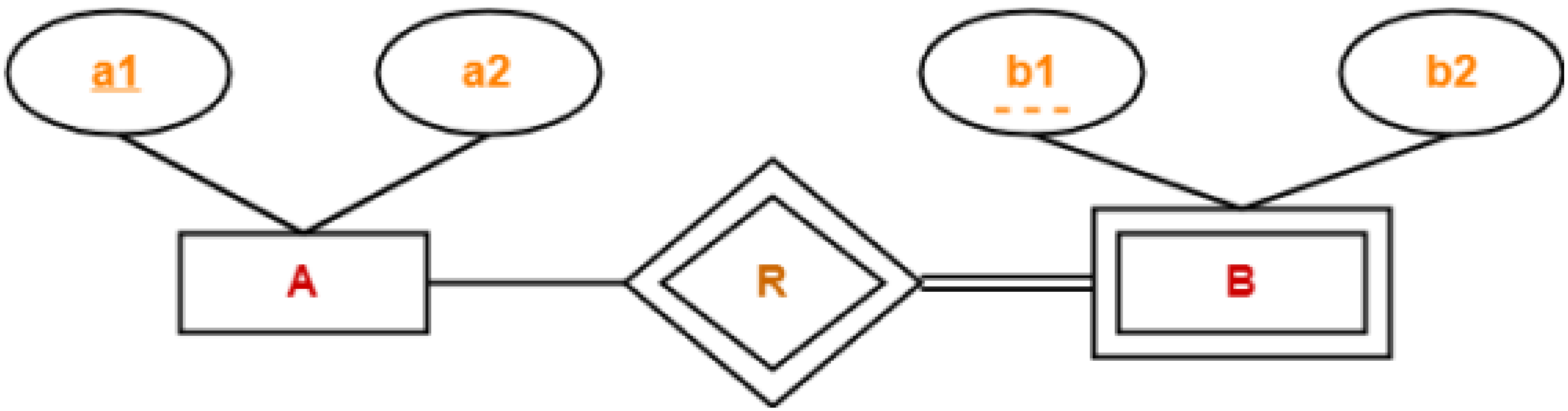
If there is a key constraint from both the sides of an entity set with total participation, then that binary relationship is represented using only single table.



ARB (a1 , a2 , b1 , b2)

CONVERSION FROM ER to TABLE

Rule-07: For Binary Relationship With Weak Entity Set
Weak entity set always appears in association with identifying relationship with total participation constraint.



Here, two tables will be required-

1. A (a1 , a2)
2. BR (a1 , b1 , b2)

CONVERSION FROM ER to TABLE

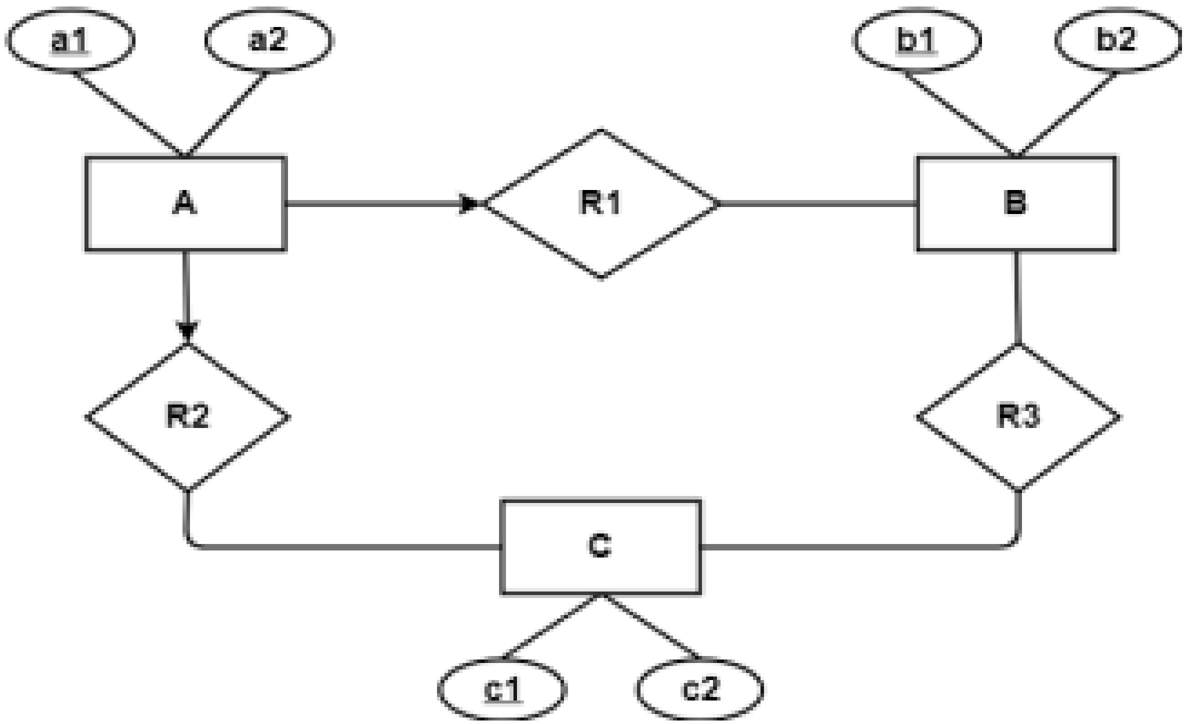
While determining the minimum number of tables required for binary relationships with given cardinality ratios, following thumb rules must be kept in mind-

- For binary relationship with cardinality ration m : n , separate and individual tables will be drawn for each entity set and relationship.
- For binary relationship with cardinality ratio either m : 1 or 1 : n , always remember "many side will consume the relationship" i.e. a combined table will be drawn for many side entity set and relationship set.
- For binary relationship with cardinality ratio 1 : 1 , two tables will be required. You can combine the relationship set with any one of the entity sets.

CONVERSION FROM ER to TABLE

Problem -1

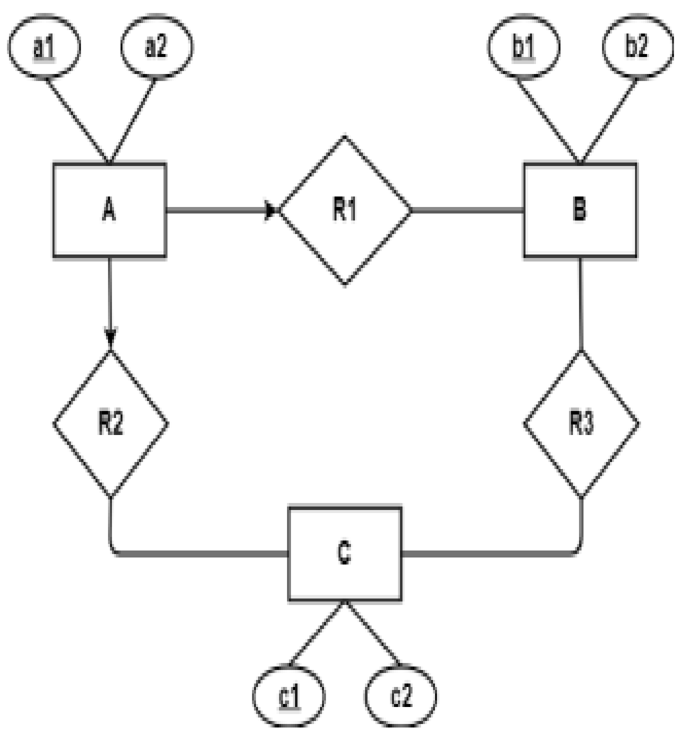
Find the minimum number of tables required to represent the given ER diagram in relational model



CONVERSION FROM ER to TABLE

Problem -1

Find the minimum number of tables required to represent the given ER diagram in relational model



Solution

Applying the rules, minimum 4 tables will be required-

- AR1R2 (a1 , a2 , b1 , c1)
- B (b1 , b2)
- C (c1 , c2)
- R3 (b1 , c1)

or

- A(a1,a2), B(b1,b2,a1), C(c1,c2,a1) R3(b1,c1)

Thank
You