www.BrainKart.com

Anna University

for Affilated Engineering College - 2021 Regulation

AID (Artificial Intelligence & Data Science Engineering)

1st Semester | 2nd Semester | 3rd Semester | 4th Semester

5th Semester | 6th Semester | 7th Semester | 8th Semester

Click on Subject/Paper under Semester to enter.

1st Semester

- Professional English - I - HS3152
- Matrices and Calculus - MA3151
- Engineering Physics - PH3151
- Engineering Chemistry - CY3151
- Problem Solving and Python Programming - GE3151

2nd Semester

- Professional English - II - HS3252
- Statistics and Numerical Methods - MA3251
- Engineering Graphics - GE3251
- Physics for Information Science - PH3256
- Basic Electrical and Electronics Engineering - BE3251
- Data Structures Design - AD3251

3rd Semester

- Discrete Mathematics - MA3354
- Digital Principles and Computer Organization - CS3351
- Database Design and Management - AD3391
- Design and Analysis of Algorithms - AD3351
- Data Exploration and Visualization - AD3301
- Artificial Intelligence - AL3391

4th Semester

- Environmental Sciences and Sustainability - GE3451
- Probability and Statistics - MA3391
- Operating Systems - AL3452
- Machine Learning - AL3451
- Fundamentals of Data Science and Analytics - AD3491
- Computer Networks - CS3591

5th Semester

- Deep Learning - AD3501
- Data and Information Security - CW3551
- Distributed Computing - CS3551
- Big Data Analytics - CCS334
- Elective 1
- Elective 2

6th Semester

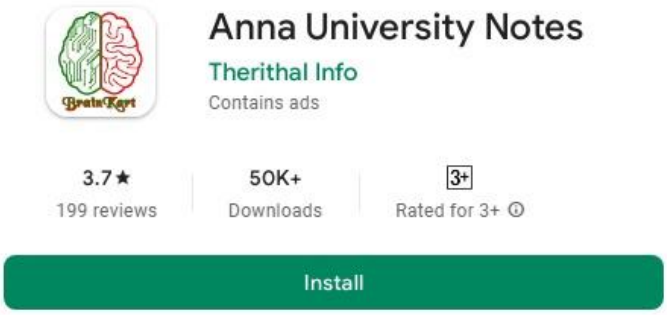
- Embedded Systems and IoT - CS3691
- Open Elective-1
- Elective-3
- Elective-4
- Elective-5
- Elective-6

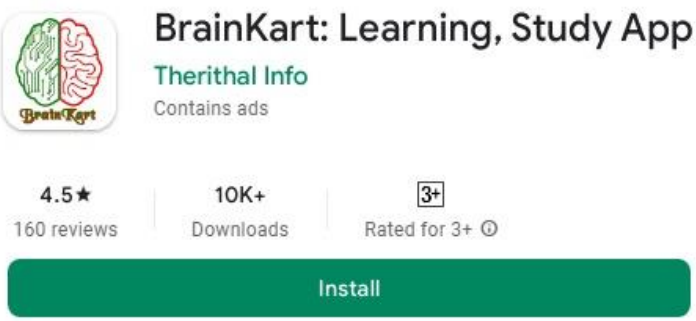
7th Semester

- Human Values and Ethics - GE3791
- Open Elective 2
- Open Elective 3
- Open Elective 4
- Management Elective

8th Semester

- Project Work / Intership





All Computer Engg Subjects - [B.E., M.E.,]		(Click on Subjects to enter)
Programming in C	Computer Networks	Operating Systems
Programming and Data Structures I	Programming and Data Structure II	Problem Solving and Python Programming
Database Management Systems	Computer Architecture	Analog and Digital Communication
Design and Analysis of Algorithms	Microprocessors and Microcontrollers	Object Oriented Analysis and Design
Software Engineering	Discrete Mathematics	Internet Programming
Theory of Computation	Computer Graphics	Distributed Systems
Mobile Computing	Compiler Design	Digital Signal Processing
Artificial Intelligence	Software Testing	Grid and Cloud Computing
Data Ware Housing and Data Mining	Cryptography and Network Security	Resource Management Techniques
Service Oriented Architecture	Embedded and Real Time Systems	Multi - Core Architectures and Programming
Probability and Queueing Theory	Physics for Information Science	Transforms and Partial Differential Equations
Technical English	Engineering Physics	Engineering Chemistry
Engineering Graphics	Total Quality Management	Professional Ethics in Engineering
Basic Electrical and Electronics and Measurement Engineering	Problem Solving and Python Programming	Environmental Science and Engineering

GRACE COLLEGE OF ENGINEERING
(Approved by AICTE, New Delhi & Affiliated to ANNA University, Chennai)
MULLAKKADU, THOOTHUKUDI - 628 005

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Anna University Regulation: 2021

AD3391- Database Design and Management

II Year/ III Semester

Question Bank

UNIT-II

Prepared By,

Mrs. S. KIRUTHIKA, AP/AIDS

PART – A

1. What is Relational Model?

Relational Model (RM)represents the database as a collection of relations. A relation isnothing but a table of values. Every row in the table represents a collection of related data values.These rows in the table denote a real-world entity or relationship. The table name and column names are helpful to interpret the meaning of values in each row.The data are represented as a

set of relations. In the relational model, data are stored as tables.However, the physical storage of the data is independent of the way the data are logicallyorganized. Some popular

Relational Database management systems are: DB2 and Informix Dynamic Server – IBM Oracle and RDB – Oracle SQL Server and Access – Microsoft

2. Relational Model Concepts in DBMS

1. Attribute:Each column in a Table. Attributes are the properties which define a relation.e.g., Student_Rollno, NAME,etc.
2. Tables – In the Relational model the, relations are saved in the table format. It is storedalong with its entities. A table has two properties rows and columns. Rows represent records andcolumns represent attributes.
3. Tuple – It is nothing but a single row of a table, which contains a single record.
4. Relation Schema:A relation schema represents the name of the relation with its attributes.
5. Degree:The total number of attributes which in the relation is called the degree of therelation.
6. Cardinality:Total number of rows present in the Table.
7. Column:The column represents the set of values for a specific attribute.
8. Relation instance – Relation instance is a finite set of tuples in the RDBMS system.Relation instances never have duplicate tuples.
9. Relation key – Every row has one, two or multiple attributes, which is called relation key.

10. Attribute domain – Every attribute has some pre-defined value and scope which isknown as attribute domain.

3. Relational Integrity Constraints

Relational Integrity constraints in DBMS are referred to conditions which must be present for avalid relation. These Relational constraints in DBMS are derived from the rules in the mini-world that the database represents. There are many types

of Integrity Constraints in DBMS. Constraints on the Relational databasemanagement system is mostly divided into three main categories are: 1. Domain Constraints2. Key Constraints3. Referential Integrity Constraints

4. Domain Constraints

Domain constraints can be violated if an attribute value is not appearing in the correspondingdomain or it is not of the appropriate data type. Domain constraints specify that within each tuple, and the value of each attribute must beunique. This is specified as data types which include standard data types integers, real numbers,characters, Booleans, variable length strings, etc. Example: Create DOMAIN CustomerNameCHECK (value not NULL)

5. Key Constraints

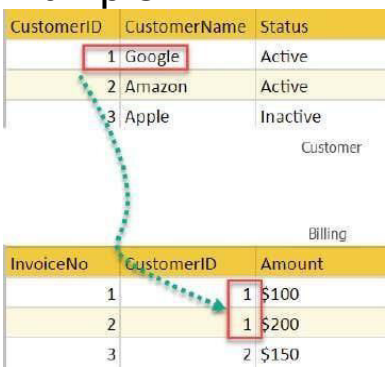
An attribute that can uniquely identify a tuple in a relation is called the key of the table. The value of the attribute for different tuples in the relation has to be unique. Example: In the given table, CustomerID is a key attribute of Customer Table. It is most likely to have a single key for one customer, CustomerID =1 is only for theCustomerName =" Google".

CustomerID	Customer Name	Status
1	Google	active
2	Amazon	active
3	Apple	inactive

6. Referential Integrity Constraints

Referential Integrity constraints in DBMS are based on the concept of Foreign Keys. A foreignkey 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 differentor

same relation. However, that key element must exist in the table.
Example:



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 $30.

7. Operations in Relational Model

Four basic update operations performed on relational database model areInsert, update, delete and select.

Insert is used to insert data into the relation Delete is used to delete tuples from the table.

Modify allows you to change the values of some attributes in existing tuples.

Select allows you to choose a specific range of data.

8. Advantages of Relational Database Model

Simplicity: A Relational data model in DBMS is simpler than the hierarchical andnetwork model.

Structural Independence: The relational database is only concerned with data and notwith a structure. This can improve the performance of the model.

Easy to use: The Relational model in DBMS is easy as tables consisting of rows andcolumns are quite natural and simple to understand

Query capability: It makes possible for a high-level query language like SQLto avoidcomplex database navigation.

Data independence: The Structure of Relational database can be changed without havingto change any application.

Scalable: Regarding a number of records, or rows, and the number of fields, a database should be enlarged to enhance its usability.

9. Disadvantages of Relational Model Few relational databases have

limits on field lengths which can't be exceeded.
Relational databases can sometimes become complex as the

amount of data grows, andthe relations between pieces of data become more complicated. Complex relational database systems may lead to isolated databases where the information cannot be shared from one system to another.

10. Integrity Constraints in the Relational Model Definition. Integrity constraints are logical statements that restrict the set of allowable relations in database
Example. database schema, R = {EMP, DEPT}, with schema(EMP) = {ENAME, DNAME, ADDRESS, POSTCODE, LOC}
schema(DEPT) = {DNAME, MNAME, NO EMPS, LOC}.
database, d = {r1, r2} OVER R, r1 is a relation over EMP, and r2 is a relation over DEPT
11. Functional Dependencies
ENAME is a key of EMP, means that no two distinct tuples in r1 have thesame ENAME. Stating that DNAME is a key of DEPT, means that no two distinct tuples in r2 have the same DNAME.? Keys are special cases of Functional Dependencies (FDs). An example of an FD which is not the result of a key, is the constraint that an ADDRESS has a unique POSTCODE.
12. SQL statements are divided into two major categories: Data Definition Language (DDL)and Data Manipulation Language (DML). Both of thesecategories contain far more statements than we can present here, and each of the statements is farmore complex than we show in this introduction. If you want to master this material, we stronglyrecommend that you find a SQL reference for your own database software as a supplement tothese pages. Few relational databases have limits on field lengths which can't be exceeded. Relational databases can sometimes become complex as the amount of data grows, andthe relations between pieces of data become more complicated.

Complex relational database systems may lead to isolated databases

where theinformation cannot be shared from one system to another.

13. Data definition language

DDL statements are used to build and modify the structure of your tables and other objects in thedatabase. When you execute a DDL statement, it takes effect immediately.The create table statement does exactly that:CREATE TABLE<tablename> (<attribute name1> <datatype1>,...<attribute name n> <datatype n>);thedata typesthat you will use most frequently are character strings, which might be calledVARCHAR or CHAR for variable or fixed length strings; numeric types such asNUMBER or INTEGER, which will usually specify a precision; and DATE or relatedtypes. Data type syntax is variable from system to system; the only way to be sure is toconsult the documentation for your own software.

14. Data manipulation language

DML statements are used to work with the data in tables.

When you are connected to most multi-user databases (whether in a client program or by a connection from a Web page script), you arein effect working with a private copy of your tables that can't be seen by anyone else until you are finished (or tell the system that you are finished). You have already seen the SELECTstatement; it is considered to be part of DML even though it just retreives data rather thanmodifying it.

The insert statement is used, obviously, to add new rows to a table.INSERT INTO<tablename>VALUES(<value1>, ... <valuen>);

15. Database views

A database view is a subset of a database and is based on a query that runs on one or more database tables. Database views are saved in the database as named queries and can be used to save frequently used, complex queries. There are two types of database views: dynamic views and static views.

Dynamic views can contain data from one or two tables and automatically include all of the columns from the specified table or tables.

Dynamic views are automatically updated when related objects or extended objects are created or changed.
Static views can contain data from multiple tables and required columns from these tables must be specified in the SELECT and WHERE clauses of the static view. Static views must be manually updated when related objects or extended objects are created or changed.

16. SQL technique: views and indexes

A view is simply any SELECT query that has been given a name and saved in the database. Forthis reason, a view is sometimes called anamed queryor astored query. To create a view, youuse the SQL syntax:CREATE OR REPLACE VIEW<view_name>AS SELECT<anyvalid select query>;The view query itself is saved in the database, but it is not actually run until it is called withanother SELECT statement. For this reason, the view does not take up any disk space for datastorage, and it does not create any redundant copies of data that is already stored in the tablesthat it references (which are sometimes called thebase tablesof the view).

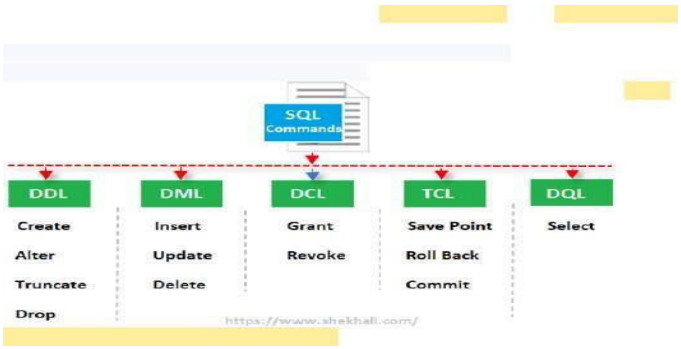
17. Materialized views

The execution speed of a query is so important that a developer is willing to trade increased diskspace use for faster response, by creating amaterialized view. Unlike the view discussed above,a materialized viewdoescreate and store the result table in advance, filled. The schemeof this table is given by the SELECT clause of the view definition. with data

18. Indexes

An index, as you would expect, is a data structure that the database uses to find records within atable more quickly. Indexes are built on one or more columns of a table; each index maintains alist of values within that field that are sorted in ascending or descending order. Rather thansorting records on the field or fields during query execution, the system can simply access therows in order of the index

19. What are the 5 basic SQL commands



There are five types of SQL commands:
DDL, DML, DCL, TCL, and DQL.
Data Definition Language (DDL) DDL changes the structure of the table like creating atable, deleting a table, altering a table, etc. ...
Data Manipulation Language. ... Data Control Language. ...
Transaction Control Language. ...
Data Query Language.

20. What is SQL example?

Structured Query Language (SQL) isa specialized language for updating, deleting, andrequesting information from databases. SQL is an ANSI and ISO standard, and is the de factostandard database query language.

PART - B

1.What is relational data model explain with example?
2.What are the components of relational model explain each?
3.What do you mean by integrity constraints explain the two types of constraints?
4.What are the types of integrity constraints?
5.What are data manipulation statements explain in detail?
6.What is data definition language and data manipulation language give one example ofeach?
7.What is Data Definition Language DDL and Data Manipulation Language DML)? Giveone example of each?
8.What is the difference between Data Definition Language and Data ManipulationLanguage?
9.What is SQL explain with an example? 10. What is SQL explain DDL DML and DCL?

www.BrainKart.com

Anna University

for Affilated Engineering College - 2021 Regulation

AID (Artificial Intelligence & Data Science Engineering)

1st Semester | 2nd Semester | 3rd Semester | 4th Semester

5th Semester | 6th Semester | 7th Semester | 8th Semester

Click on Subject/Paper under Semester to enter.

1st Semester

- Professional English - I - HS3152
- Matrices and Calculus - MA3151
- Engineering Physics - PH3151
- Engineering Chemistry - CY3151
- Problem Solving and Python Programming - GE3151

2nd Semester

- Professional English - II - HS3252
- Statistics and Numerical Methods - MA3251
- Engineering Graphics - GE3251
- Physics for Information Science - PH3256
- Basic Electrical and Electronics Engineering - BE3251
- Data Structures Design - AD3251

3rd Semester

- Discrete Mathematics - MA3354
- Digital Principles and Computer Organization - CS3351
- Database Design and Management - AD3391
- Design and Analysis of Algorithms - AD3351
- Data Exploration and Visualization - AD3301
- Artificial Intelligence - AL3391

4th Semester

- Environmental Sciences and Sustainability - GE3451
- Probability and Statistics - MA3391
- Operating Systems - AL3452
- Machine Learning - AL3451
- Fundamentals of Data Science and Analytics - AD3491
- Computer Networks - CS3591

5th Semester

- Deep Learning - AD3501
- Data and Information Security - CW3551
- Distributed Computing - CS3551
- Big Data Analytics - CCS334
- Elective 1
- Elective 2

6th Semester

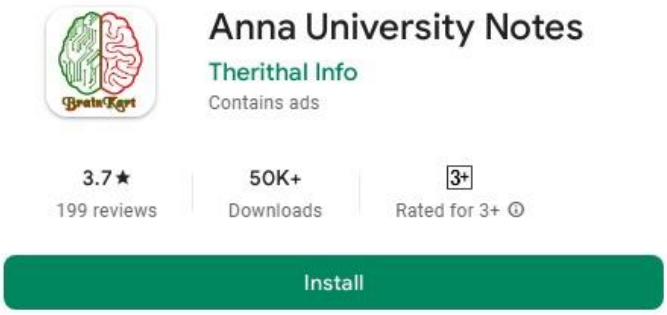
- Embedded Systems and IoT - CS3691
- Open Elective-1
- Elective-3
- Elective-4
- Elective-5
- Elective-6

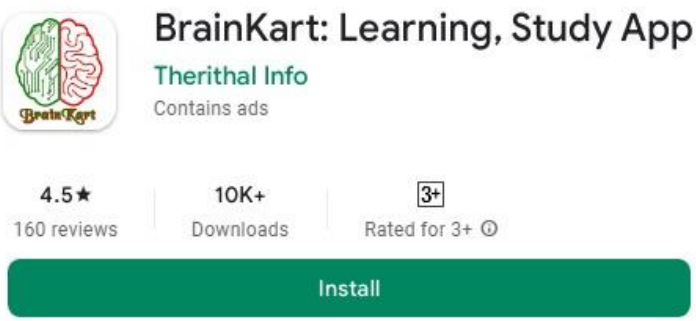
7th Semester

- Human Values and Ethics - GE3791
- Open Elective 2
- Open Elective 3
- Open Elective 4
- Management Elective

8th Semester

- Project Work / Intership





All Computer Engg Subjects - [B.E., M.E.,]		(Click on Subjects to enter)
Programming in C	Computer Networks	Operating Systems
Programming and Data Structures I	Programming and Data Structure II	Problem Solving and Python Programming
Database Management Systems	Computer Architecture	Analog and Digital Communication
Design and Analysis of Algorithms	Microprocessors and Microcontrollers	Object Oriented Analysis and Design
Software Engineering	Discrete Mathematics	Internet Programming
Theory of Computation	Computer Graphics	Distributed Systems
Mobile Computing	Compiler Design	Digital Signal Processing
Artificial Intelligence	Software Testing	Grid and Cloud Computing
Data Ware Housing and Data Mining	Cryptography and Network Security	Resource Management Techniques
Service Oriented Architecture	Embedded and Real Time Systems	Multi - Core Architectures and Programming
Probability and Queueing Theory	Physics for Information Science	Transforms and Partial Differential Equations
Technical English	Engineering Physics	Engineering Chemistry
Engineering Graphics	Total Quality Management	Professional Ethics in Engineering
Basic Electrical and Electronics and Measurement Engineering	Problem Solving and Python Programming	Environmental Science and Engineering