**LAB MANUAL**

**Lab Name :**  Database Management System Lab

**Lab Code :** 4CS4-22

**Branch :** Computer Science & Engineering

**Year :** 2nd Year



**Jaipur Engineering College and Research Center, Jaipur**

Department of Electrical Engineering

(Rajasthan Technical University, KOTA)

**INDEX**

|  |  |
| --- | --- |
| **S.NO** | **CONTENTS** |
| **1** | VISION/MISION |
| **2.** | PEO |
| **3.** | POS |
| **4.** | COS |
| **5.** | MAPPING OF CO & PO |
| **6.** | SYLLABUS |
| **7.** | BOOKS |
| **8.** | INSTRUCTIONAL METHODS |
| **9.** | LEARNING MATERIALS |
| **10.** | ASSESSMENT OF OUTCOMES |
|  | LIST OF EXPERIMENTS (RTU SYLLABUS) |
| **Exp: - 1** | **Objective: -1.** Stating a database design & application problem. |
| **Exp: - 2** | **Objective: -2.** Prepare ER diagram |
| **Exp: -3** | **Objective: -3.** Finding the data fields to be used in the database |
| **Exp: -4** | **Objective: - 4.** Selecting fields for keys. |
| **Exp: -5** | **Objective: - 5.** Normalizing the database including analysis of functional dependencies. |
| **Exp: -6** | **Objective: - 6.** Installing and configuring the database server and the front end tools. |
| **Exp: -7** | **Objective: - 7.** Designing database and writing applications for manipulation of data for a standalone and shared data base including concepts like concurrency control, transaction roll back, logging, report generation etc. |
| **Exp: -8** | **Objective: -8.** Get acquainted with Structured Query Language. |

1. **VISION & MISSION**

**VISION:** To become renowned Centre of excellence in computer science and engineering and make competent engineers & professionals with high ethical values prepared for lifelong learning.

**MISSION:**

**M1:** To impart outcome-based education for emerging technologies in the field of computer science and engineering.

**M2:** To provide opportunities for interaction between academia and industry.

**M3:** To provide platform for lifelong learning by accepting the change in technologies

**M4:** To develop aptitude of fulfilling social responsibilities

**2. PEO**

**1.** To provide students with the fundamentals of Engineering Sciences with more emphasis in **Computer Science &Engineering** by way of analyzing and exploiting engineering challenges.

**2.** To train students with good scientific and engineering knowledge so as to comprehend, analyze, design, and create novel products and solutions for the real life problems.

**3.** To inculcate professional and ethical attitude, effective communication skills, teamwork skills, multidisciplinary approach, entrepreneurial thinking and an ability to relate engineering issues with social issues.

**4.** To provide students with an academic environment aware of excellence, leadership, written ethical codes and guidelines, and the self-motivated life-long learning needed for a successful professional career.

**5**. To prepare students to excel in Industry and Higher education by Educating Students along with High moral values and Knowledge

1. **PROGRAM OUTCOMES**
2. **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems in IT.
3. **Problem analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences in IT.
4. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations using IT.
5. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions using IT.
6. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations in IT.
7. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice using IT.
8. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development in IT.
9. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice using IT.
10. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings in IT.
11. **Communication:**  Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
12. **Project Management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage IT projects and in multidisciplinary environments.
13. **Life –long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological changes needed in IT.

**MAPPING OF PEOs & POs**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PROGRAM OBJECTIVES** | **PROGRAM OUTCOMES** | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **I** | **3** | **1** |  |  |  |  |  |  |  |  |  | **3** |
| **II** | **2** | **3** | **2** | **3** |  | **3** |  |  |  |  | **1** | **3** |
| **III** | **1** | **3** | **2** | **3** | **1** |  |  |  |  |  | **2** |  |
| **IV** |  |  |  | **1** | **2** |  | **3** | **2** | **3** |  | **2** |  |
| **V** |  |  |  |  |  |  |  |  | **2** | **2** |  |  |

[L=1, M=2, H=3]

1. **COURSE OUTCOMES**

CO1: Design a database for a system and understand the requirement and design methodology of system.

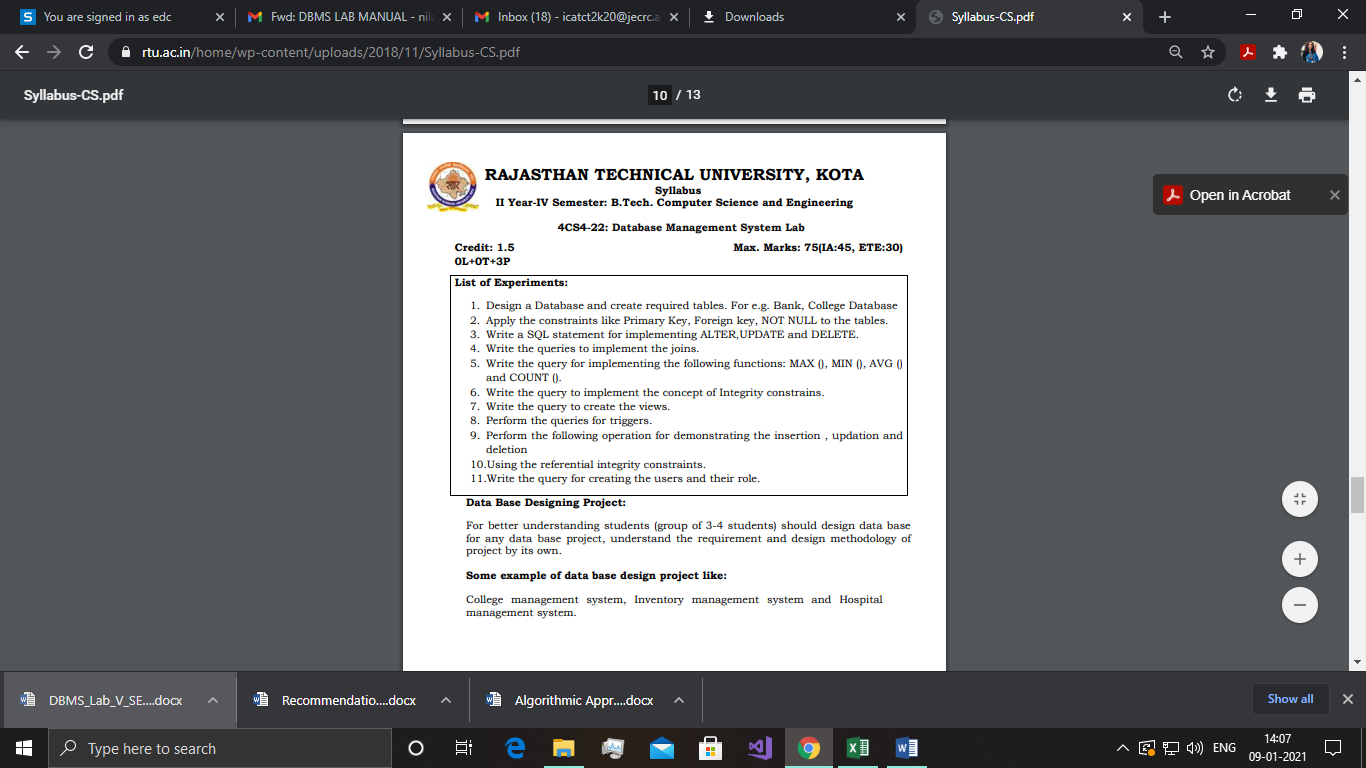
CO2: Perform queries, operations and retrieve data from the database using SQL.

1. **MAPPING OF CO & PO**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Subject** | **CO’s** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| Database Management System Lab(4CS4-22) | CO1 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 3 | 2 | 3 |
| CO2 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 2 | 2 | 3 | 2 | 3 |

[L=1, M=2, H=3]

1. **Syllabus**



1. **BOOKS**
   * **Text books:-**

1. H. F. Korth and Silberschatz, “Database Systems Concepts”, Tata McGraw Hill (5th Edition)

* **Reference Books:-**

1. Almasri and S.B. Navathe, “Fundamentals of Database Systems”, Pearson (4th Edition)

**8. INSTRUCTIONAL METHODS**

* **Direct Instructions:**

1. Black board presentation

* **Interactive Instruction:**

1. Case Study

* **Indirect Instructions:**

1. Problem solving

* **Independent Instructions:**

1. Assignments

**9. LEARNING MATERIALS**

Text/Lab Manual

Web Resources**:-**

i) http://www.w3schools.com/sql/default.asp

ii) http://msdn.microsoft.com/en-us/library/bb264565%28v=sql.90%29.aspx

iii) http://www.sqlcourse.com/index.html

**10. ASSESSMENT OF OUTCOMES**

1. End term Practical exam (Conducted by RTU, KOTA)
2. Daily Lab interaction.
3. Lab Assignment.

* **OUTCOMES WILL BE ACHIEVED THROUGH FOLLOWING:-**

1. Lab Teaching (through chalk and board /PPT).

2. Discussion on case- studies.

**INSTRUCTIONS OF LAB**

#### DO’s

1. Please switch off the Mobile/Cell phone before entering Lab.
2. Enter the Lab with complete source code and data.
3. Check whether all peripheral are available at your desktop before proceeding for program.
4. Intimate the lab in charge whenever you are incompatible in using the system or in case software get corrupted/ infected by virus.
5. Arrange all the peripheral and seats before leaving the lab.
6. Properly shutdown the system before leaving the lab.
7. Keep the bag outside in the racks.
8. Enter the lab on time and leave at proper time.
9. Maintain the decorum of the lab.
10. Utilize lab hours in the corresponding experiment.
11. Get your CD / Pen drive checked by lab in charge before using it in the lab.

###### DON’TS

1. No one is allowed to bring storage devices like Pan Drive /Floppy etc. in the lab.
2. Don’t mishandle the system.
3. Don’t leave the system on standing for long
4. Don’t bring any external material in the lab.
5. Don’t make noise in the lab.
6. Don’t bring the mobile in the lab. If extremely necessary then keep ringers off.
7. Don’t enter in the lab without permission of lab in charge.
8. Don’t litter in the lab.
9. Don’t delete or make any modification in system files.
10. Don’t carry any lab equipment outside the lab.

We need your full support and cooperation for smooth functioning of the

**INSTRUCTIONS FOR STUDENT**

BEFORE ENTERING IN THE LAB

* All the students are supposed to prepare the theory regarding the next program.
* Students are supposed to bring the practical file and the lab copy.
* Previous programs should be written in the practical file.
* Algorithm of the current program should be written in the lab copy.
* Any student not following these instructions will be denied entry in the lab.

# WHILE WORKING IN THE LAB

* Adhere to experimental schedule as instructed by the lab in charge.
* Get the previously executed program signed by the instructor.
* Get the output of the current program checked by the instructor in the lab copy.
* Each student should work on his/her assigned computer at each turn of the lab.
* Take responsibility of valuable accessories.
* Concentrate on the assigned practical and do not play games.

**Software and Hardware Requirements**

**Software Required:**

1. VB, ORACLE and/or DB2
2. VB, MSACCESS
3. ORACLE, D2K
4. VB, MS SQL SERVER 2000

**Hardware Required**:

Processor: Pentium III

RAM: 128 MB

Hard Disk: 40 GB

**Theory and Concept**

**Practical #1**

Objective: Create tables and specify the Questionnaires in SQL.

**Theory & Concepts:**

**Introduction about SQL-**

SQL (Structured Query Language) is a nonprocedural language, you specify what you want, not how to get it. A block structured format of English key words is used in this Query language. It has the following components.

**DDL (Data Definition Language)-**

The SQL DDL provides command for defining relation schemas, deleting relations and modifying relation schema.

**DML (DATA Manipulation Language)-**

It includes commands to insert tuples into, delete tuples from and modify tuples in the database.

**View definition-**

The SQL DDL includes commands for defining views.

Transaction Control- SQL includes for specifying the beginning and ending of transactions.

**Embedded SQL and Dynamic SQL-**

Embedded and Dynamic SQL define how SQL statements can be embedded with in general purpose programming languages, such as C, C++, JAVA, COBOL, Pascal and Fortran.

**Integrity-**

The SQL DDL includes commands for specifying integrity constraints that the data stored in the database must specify. Updates that violate integrity constraints are allowed.

**Authorization-**

The SQL DDL includes commands for specifying access rights to relations and views.

**Data Definition Language-**

The SQL DDL allows specification of not only a set of relations but also information about each relation, including-

* Schema for each relation
* The domain of values associated with each attribute.
* The integrity constraints.
* The security and authorization information for each relation.
* The physical storage structure of each relation on disk.

**Domain types in SQL-**

**The SQL standard supports a variety of built in domain types, including-**

* Char (n) - A fixed length character length string with user specified length.
* Varchar (n) - A variable character length string with user specified maximum length n.
* Int- An integer.
* Small integer- A small integer.
* Numeric (p, d)-A Fixed point number with user defined precision.
* Real, double precision- Floating point and double precision floating point numbers with machine dependent precision.
* Float (n) - A floating point number, with precision of at least n digits.
* Date- A calendar date containing a (four digit) year, month and day of the month.
* Time- The time of day, in hours, minutes and seconds E.g. Time ’09:30:00’.
* Number- Number is used to store numbers (fixed or floating point).

**DDL statement for creating a table-**

**Syntax-**

Create table tablename

(columnname datatype(size), columnname datatype(size));

Creating a table from a table-

**Syntax-**

CREATE TABLE TABLENAME

[(columnname, columnname, ………)]

AS SELECT columnname, columnname……..FROM tablename;

**Insertion of data into tables-**

**Syntax-**

INSERT INTO tablename

[(columnname, columnname, ………)]

Values(expression, expression);

**Inserting data into a table from another table:**

**Syntax-**

INSERT INTO tablename

SELECT columnname, columnname, …….

FROM tablename;

**Insertion of selected data into a table from another table:**

**Syntax-**

INSERT INTO tablename

SELECT columnname, columnname……..

FROM tablename

WHERE columnname= expression;

**Retrieving of data from the tables-**

**Syntax-**

SELECT \* FROM tablename;

**The retrieving of specific columns from a table-**

**Syntax-**

SELECT columnname, columnname, ….

FROM tablename;

**Elimination of duplicates from the select statement-**

**Syntax-**

SELECT DISTINCT columnname, columnname FROM tablename;

**Selecting a data set from table data-**

**Syntax-**

SELECT columnname, columnname

FROM tablename

WHERE searchcondition;

**Assignment No.1**

**Q1. Create the following tables:**

1. **client\_master**

|  |  |  |
| --- | --- | --- |
| columnname | datatype | size |
| client\_no | varchar2 | 6 |
| name | varchar2 | 20 |
| address1 | varchar2 | 30 |
| address2 | varchar2 | 30 |
| city | varchar2 | 15 |
| state | varchar2 | 15 |
| pincode | number | 6 |
| bal\_due | number | 10,2 |

1. **Product\_master**

Columnnamedatatypesize

Product\_novarchar2

Description varchar2 Profit\_percent number Unit\_measure varchar2 Qty\_on\_hand number Reoder\_lvlnumber

Sell\_price number Cost\_price number

**Q2- Insert the following data into their respective tables:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Clientno | | | Name | | city | pincode | | state | bal.due |
|  |  |  |  | |  |  |  | |  |
| 0001 | | Ivan |  |  | Bombay | 400054 | Maharashtra | | 15000 |
| 0002 | | Vandana | |  | Madras | 780001 | Tamil Nadu | | 0 |
| 0003 | | Pramada | |  | Bombay | 400057 | Maharashtra | | 5000 |
| 0004 | | Basu |  |  | Bombay | 400056 | Maharashtra | | 0 |
| 0005 | | Ravi |  |  | Delhi | 100001 |  |  | 2000 |
| 0006 | | Rukmini | |  | Bombay | 400050 | Maharashtra | | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Data for Product Master:** | |  |  |  |  |  |  |  |
| Product No. | Descriptions | Profit % | Unit | Qty | Reorder | Sell | Cost |  |
| price |  | Percent | | measured | on hand lvl |  | price |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| P00001 | 1.44floppies | 5 | piece | 100 | 20 | 525 | 500 |  |
| P03453 | Monitors | 6 | piece | 10 | 3 | 12000 | 11200 |  |
| P06734 | Mouse | 5 | piece | 20 | 5 | 1050 | 500 |  |
| P07865 | 1.22 floppies | 5 | piece | 100 | 20 | 525 | 500 |  |
| P07868 | Keyboards | 2 | piece | 10 | 3 | 3150 | 3050 |  |
| P07885 | CD Drive | 2.5 | piece | 10 | 3 | 5250 | 5100 |  |
| P07965 | 540 HDD | 4 | piece | 10 | 3 | 8400 | 8000 |  |
| P07975 | 1.44 Drive | 5 | piece | 10 | 3 | 1050 | 1000 |  |
| P08865 | 1.22 Drive | 5 | piece | 2 | 3 | 1050 | 1000 |  |

**Q3: - On the basis of above two tables answer the following Questionaries:**

1. Find out the names of all the clients.
2. Retrieve the list of names and cities of all the clients.
3. List the various products available from the product master table.
4. List all the clients who are located in Bombay.
5. Display the information for client no 0001 and 0002.
6. Find the products with description as ‘1.44 drive’ and ‘1.22 Drive’.
7. Find all the products whose sell price is greater then 5000.
8. Find the list of all clients who stay in in city ‘Bombay’ or city ‘Delhi’ or ‘Madras’.
9. Find the product whose selling price is greater than 2000 and less than or equal to 5000.
10. List the name, city and state of clients not in the state of ‘Maharashtra’.

**Theory and Concept**

**Practical #2**

**Objective:- To Manipulate the Operations on the table.**

DML ( Data Manipulation Language) Data manipulation is

* The retrieval of information stored in the database.
* The insertion of new information into the database.
* The deletion of information from the database.
* The modification of information stored by the appropriate data model. There are basically two types.

1. **Procedural DML**:- require a user to specify what data are needed and howto get those data.
2. **Non Procedural DML** : require a user to specify what data are neededwithout specifying how to get those data.

**Updating the content of a table:**

In creation situation we may wish to change a value in table without changing all values in the tuple . For this purpose the update statement can be used.

Update table name

Set columnname = expression, columnname =expression……

Where columnname = expression;

**Deletion Operation:-**

A delete query is expressed in much the same way as Query. We can delete whole tuple ( rows) we can delete values on only particulars attributes.

**Deletion of all rows**

**Syntax:**

Delete from tablename :

**Deletion of specified number of rows**

**Syntax:**

Delete from table name

Where search condition ;

**Computation in expression lists used to select data**

|  |  |  |
| --- | --- | --- |
| + | Addition | - Subtraction |
| \* | multiplication | \*\* exponentiation |
| / Division | | () Enclosed operation |

Renaming columns used with Expression Lists: - The default output column names can be renamed by the user if required

**Syntax:**

Select column name

Columnname

From table name;

result\_columnname,

result\_columnname,

**Logical Operators:**

The logical operators that can be used in SQL sentenced are

AND

OR

NOT

all of must be included

any of may be included

none of could be included

**Range Searching:** Between operation is used for range searching.

**Pattern Searching:**

The most commonly used operation on string is pattern matching using the operation ‘like’ we describe patterns by using two special characters.

* Percent (%) ; the % character matches any substring we consider the following examples.
* ‘Perry %’ matches any string beginning with perry
* ‘% idge % matches any string containing’ idge as substring.
* ‘ - - - ‘ matches any string exactly three characters.
* ‘ - - - % matches any string of at least of three characters.

**Oracle functions:**

Functions are used to manipulate data items and return result. function follow the format of function \_name (argument1, argument2 ..) .An arrangement is user defined variable or constant. The structure of function is such that it accepts zero or more arguments.

Examples:

Avg

return average value of n

**Syntax:**

Avg ([distinct/all]n)

Min return minimum value of expr.

**Syntax:**

MIN((distinct/all )expr)

Count Returns the no of rows where expr is not null

**Syntax:**

Count ([distinct/all)expr]

Count (\*) Returns the no rows in the table, including duplicates and those with nulls. Max Return max value of expr

**Syntax:**

Max ([distinct/all]expr)

Sum Returns sum of values of n

**Syntax:**

Sum ([distinct/all]n)

**Sorting of data in table**

**Syntax:**

Select columnname, columnname

From table

Order by columnname;

**Assignment No. # 2**

**Question.1 Using the table client master and product master answer the following Questionnaires.**

1. Change the selling price of ‘1.44 floppy drive to Rs.1150.00
2. Delete the record with client 0001 from the client master table.
3. Change the city of client\_no’0005’ to Bombay.
4. Change the bal\_due of client\_no ‘0001, to 1000.
5. Find the products whose selling price is more than 1500 and also find the new selling price as original selling price \*15.
6. Find out the clients who stay in a city whose second letter is a.
7. Find out the name of all clients having ‘a’ as the second letter in their names.
8. List the products in sorted order of their description.
9. Count the total number of orders
10. Calculate the average price of all the products.
11. Calculate the minimum price of products.
12. Determine the maximum and minimum prices . Rename the tittle as ‘max\_price’ and min\_price respectively.
13. Count the number of products having price greater than or equal to 1500.

**Theory and Concept**

**Practical #3**

Objective:- To Implement the restrictions on the table.

**Data constraints:** Besides the cell name, cell length and cell data type there are otherparameters i.e. other data constrains that can be passed to the DBA at check creation time. The constraints can either be placed at column level or at the table level.

1. **Column Level Constraints:** If the constraints are defined along with the columndefinition, it is called a column level constraint.
2. **Table Level Constraints:** If the data constraint attached to a specify cell in atable reference the contents of another cell in the table then the user will have to use table level constraints.

**Null Value Concepts:-**while creating tables if a row locks a data value forparticular column that value is said to be null . Column of any data types may contain null values unless the column was defined as not null when the table was created

**Syntax:**

**Create table tablename**

**(**columnname data type (size) not null ……)

**Primary Key:** primary key is one or more columns is a table used to uniquelyidentity each row in the table. Primary key values must not be null and must be unique across the column. A multicolumn primary key is called composite primary key.

**Syntax: primary key as a column constraint**Create table tablename

(columnname datatype (size) primary key,….)

**Primary key as a table constraint** Create table tablename

(columnname datatype (size), columnname datatype( size)… Primary key (columnname,columnname));

**Default value concept: At** the line of cell creation a default value can be assignedto it. When the user is loading a record with values and leaves this cell empty, the DBA will automatically load this cell with the default value specified. The data type of the default value should match the data type of the column

**Syntax:**

Create table tablename

(columnname datatype (size) default value,….);

**Foreign Key Concept :**Foreign key represents relationship between tables. A

foreign key is column whose values are derived from the primary key of the same

of some other table . the existence of foreign key implies that the table with

foreign key is related to the primary key table from which the foreign key is

derived .A foreign key must have corresponding primary key value in the primary

key table to have meaning.

Foreign key as a column constraint

**Syntax :**

Create table table name

(columnname datatype (size) references another table name);

**Foreign key as a table constraint:**

**Syntax :**

Create table name

(columnname datatype (size)….

primary key (columnname);

foreign key (columnname)references table name);

**Check Integrity Constraints:** Use the check constraints when you need toenforce integrity rules that can be evaluated based on a logical expression following are a few examples of appropriate check constraints.

* A check constraints name column of the client\_master so that the name is entered in upper case.
* A check constraint on the client\_no column of the client \_master so that no client\_no value starts with ‘c’

**Syntax:**

Create table tablename

(columnname datatype (size) CONSTRAINT constraintname)

Check (expression));

**Question.2 Create the following tables:**

1. **Sales\_master**

|  |  |  |  |
| --- | --- | --- | --- |
| **Columnname** | **Datatype** | **Size** | **Attributes** |
| Salesman\_no | varchar2 | 6 | Primary key/first letter |
|  |  |  | must start with ‘s’ |
| Sal\_name | varchar2 | 20 | Not null |
| Address | varchar2 |  | Not null |
| City | varchar2 | 20 |  |
| State | varchar2 | 20 |  |
| Pincode | Number | 6 |  |
| Sal\_amt | Number | 8,2 | Not null, cannot be 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Tgt\_to\_get | Number | 6,2 | Not null, cannot be 0 |
| Ytd\_sales | Number | 6,2 | Not null, cannot be 0 |
| Remarks | Varchar2 | 30 |  |

1. **Sales\_order**

|  |  |  |  |
| --- | --- | --- | --- |
| **Columnname** | **Datatype** | **Size** | **Attributes** |
| S\_order\_no | varchar2 | 6 | Primary/first letter must be 0 |
|  |  |  |  |
| S\_order\_date | Date | 6 | Primary key reference clientno |
|  |  |  | of client\_master table |
|  |  |  |  |
| Client\_no | Varchar2 | 25 |  |
| Dely\_add | Varchar2 | 6 |  |
| Salesman\_no | Varchar2 | 6 | Foreign key references |
|  |  |  | salesman\_no of |
|  |  |  | salesman\_master table |
| Dely\_type | Char | 1 | Delivery part(p)/full(f),default f |
| Billed\_yn | Char | 1 |  |
| Dely\_date | Date |  | Can not be lessthan |
|  |  |  | s\_order\_date |
| Order\_status | Varchar2 | 10 | Values (‘in |
|  |  |  | process’;’fulfilled’;back |
|  |  |  | order’;’canceled |

**I. Sales\_order\_details**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Datatype** | **Size** | **Attributes** |
|  |  |  |  |
| S\_order\_no | Varchar2 | 6 | Primary key/foreign |
|  |  |  | key references |
|  |  |  | s\_order\_no of |
|  |  |  | sales\_order |
| Product\_no | Varchar2 | 6 | Primary key/foreign |
|  |  |  | key references |
|  |  |  | product\_no of |
|  |  |  | product\_master |
| Qty\_order | Number | 8 |  |
| Qty\_disp | Number | 8 |  |
| Product\_rate | Number | 10,2 |  |

Insert the following data into their respective tables using insert statement:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Salesman\_** | **Salesman** | **Address** | **City** | **Pin** | **State** | **Salamt** | **Tgt\_to\_get** | **Ytd** | **Rem** |
| **no** | **name** |  |  | **code** |  |  |  | **Sales** |  |
| 500001 | Kiran | A/14 | Bom | 400002 | Mah | 3000 | 100 | 50 | Goo |
|  |  | worli | bay |  |  |  |  |  |  |
| 500002 | Manish | 65,narim | Bom | 400001 | Mah | 3000 | 200 | 100 | Goo |
|  |  | an | bay |  |  |  |  |  |  |
| 500003 | Ravi | P-7 | Bom | 400032 | Mah | 3000 | 200 | 100 | Goo |
|  |  | Bandra | bay |  |  |  |  |  |  |
| 500004 | Ashish | A/5 | Bom | 400044 | Mah | 3500 | 200 | 150 | Goo |
|  |  | Juhu | bay |  |  |  |  |  |  |

**Data for sales\_man master table**

(ii)

**Data for salesorder table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S\_orderno | S\_orderdate | Client no | Dely | Bill | Salesman no | Delay | Orderstatus |
|  |  |  | type | yn |  | date |  |
| 019001 | 12-jan-96 | 0001 | F | N | 50001 | 20-jan- | Ip |
|  |  |  |  |  |  | 96 |  |
| 019002 | 25-jan-96 | 0002 | P | N | 50002 | 27-jan- | C |
|  |  |  |  |  |  | 96 |  |
| 016865 | 18-feb-96 | 0003 | F | Y | 500003 | 20-feb- | F |
|  |  |  |  |  |  | 96 |  |
| 019003 | 03-apr-96 | 0001 | F | Y | 500001 | 07-apr- | F |
|  |  |  |  |  |  | 96 |  |
| 046866 | 20-may-96 | 0004 | P | N | 500002 | 22- | C |
|  |  |  |  |  |  | may-96 |  |
| 010008 | 24-may-96 | 0005 | F | N | 500004 | 26- | Ip |
|  |  |  |  |  |  | may-96 |  |

(iii)

**Data for sales\_order\_details table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S\_order no | Product no | Qty ordered | Qtydisp | Product\_rate |
| 019001 | P00001 | 4 | 4 | 525 |
| 019001 | P07965 | 2 | 1 | 8400 |
| 019001 | P07885 | 2 | 1 | 5250 |
| 019002 | P00001 | 10 | 0 | 525 |
| 046865 | P07868 | 3 | 3 | 3150 |
| 046865 | P07885 | 10 | 10 | 5250 |
| 019003 | P00001 | 4 | 4 | 1050 |
| 019003 | P03453 | 2 | 2 | 1050 |
| 046866 | P06734 | 1 | 1 | 12000 |
| 046866 | P07965 | 1 | 0 | 8400 |
| 010008 | P07975 | 1 | 0 | 1050 |
| 010008 | P00001 | 10 | 5 | 525 |

**Theory and Concept**

**Practical .4**

**Objective:- To Implement the structure of the table**

**Modifying the Structure of Tables**- Alter table command is used to changing thestructure of a table. Using the alter table clause you cannot perform the following tasks:

1. change the name of table
2. change the name of column
3. drop a column
4. decrease the size of a table if table data exists.

The following tasks you can perform through alter table command.

1. **Adding new columns**:Syntax

ALTER TABLE tablename

ADD (newcolumnnamenewdatatype (size));

1. **Modifying existing table** Syntax:

ALTER TABLE tablename

MODIFY (newcolumnnamenewdatatype (size));

***NOTE*:**Oracle not allow constraints defined using the alter table, if the data in the table,violates such constraints.

**Removing/Deleting Tables**- Following command is used for removing or deleting atable.

Syntax:

DROP TABLE tablename:

Defining Integrity constraints in the ALTER TABLE command-

You can also define integrity constraints using the constraint clause in the ALTER TABLE command. The following examples show the definitions of several integrity constraints.

1. **Add PRIMARY KEY**-Syntax:

ALTER TABLE tablename

ADD PRIMARY KEY (columnname);

1. **Add FOREIGN KEY**-Syntax:

ALTER TABLE tablename

ADD CONSTRAINT constraintname

FOREIGN KEY(columnname) REFERENCES tablename;

Dropping integrity constraints in the ALTER TABLE command:

You can drop an integrity constraint if the rule that if enforces is no longer true or if the constraint is no longer needed. Drop the constraint using the ALTER TABLE command with the DROP clause. The following examples illustrate the dropping of integrity constraints.

1. **DROP the PRIMARY KEY**-Syntax:

ALTER TABLE tablename

DROP PRIMARY KEY

1. **DROP FOREIGN KEY**-Syntax:

ALTER TABLE tablename

DROP CONSTRAINT constraintname;

**Assignment No.4**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Question 1. Create the following tables: | | | |  |  |
| **Challan\_Header** | | **data type** | **size** | **Attributes** |  |
| **Column name** |  |  |
| Challan\_no | | varchar2 | 6 | Primary key |  |
| s\_order\_no | | varchar2 | 6 | Foreign key references s\_order\_no of |  |
|  |  |  |  | sales\_order table |  |
| challan\_date | | date |  | not null |  |
| billed\_yn | | char | 1 | values (‘Y’,’N’). Default ‘N’ |  |
| **Table Name : Challan\_Details** | | |  |  |  |
| **Column name** | | **data type** | **size** | **Attributes** |  |
| Challan\_no | | varchar2 | 6 | Primary key/Foreign key references |  |
|  |  |  |  | Product\_no of product\_master |  |
| Qty\_disp | | number | 4,2 | not null |  |

Q2. Insert the following values into the challan header and challan\_details tables:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| (i) | **Challan No** | **S\_order No** | **Challan Date Billed** | |
|  | CH9001 | 019001 | 12-DEC-95 | Y |
|  | CH865 | 046865 | 12-NOV-95 Y | |
|  | CH3965 | 010008 | 12-OCT-95 | Y |
| Data for challan\_details table | | |  |  |

|  |  |  |
| --- | --- | --- |
| **Challan No** | **Product No** | **QtyDisp** |
| CH9001 | P00001 | 4 |
| CH9001 | P07965 | 1 |
| CH9001 | P07885 | 1 |
| CH6865 | P07868 | 3 |
| CH6865 | P03453 | 4 |
| CH6865 | P00001 | 10 |
| CH3965 | P00001 | 5 |
| CH3965 | P07975 | 2 |

**Objective –**Answer the following Questionries

Q1. Make the primary key to client\_no in client\_master.

Q2. Add a new column phone\_no in the client\_master table.

Q3. Add the not null constraint in the product\_master table with the columns description, profit percent , sell price and cost price.

Q4. Change the size of client\_no field in the client\_master table.

Q5. Select product\_no, description where profit percent is between 20 and 30 both inclusive.

**Theory & Concept**

**Practical #5**

**Objective:- To implement the concept of Joins**

**Joint Multiple Table (Equi Join):**Some times we require to treat more than one table asthough manipulate data from all the tables as though the tables were not separate object but one single entity. To achieve this we have to join tables.Tables are joined on column that have dame data type and data with in tables.

The tables that have to be joined are specified in the FROM clause and the joining attributes in the WHERE clause.

**Algorithm for JOIN in SQL:**

* 1. Cartesian product of tables (specified in the FROM clause)
  2. Selection of rows that match (predicate in the WHERE clause)
  3. Project column specified in the SELECT clause.

1. **Cartesian product:-**

Consider two table student and course

Select B.\*,P.\*

FROM student B, course P;

**2. INNER JOIN:**

Cartesian product followed by selection

Select B.\*,P.\*

FROM student B, Course P

WHERE B.course # P.course# ;

**3. LEFT OUTER JOIN:**

LEFT OUTER JOIN = Cartesian product + selection but include rows from the left table which are unmatched pat nulls in the values of attributes belonging to th e second table

Exam:

Select B.\*,P\*

FROM student B left join course p

ON B.course # P.course #;

**4. RIGHT OUTER JOIN:**

RIGHT OUTER JOIN = Cartesian product + selection but include rows from right table which are unmatched

Exam:

Select B.\*,P.\*

From student B RIGHT JOIN course P

B.course# = P course # ;

1. **FULL OUTER JOIN** Exam Select B.\*,P.\*

From student B FULL JOIN course P On B.course # = P course # ;

**ASSIGNMENT NO. 5**

**OBJECTIVE: A**nswer the following Queries:

1. Find out the product which has been sold to ‘Ivan Sayross.’
2. Find out the product and their quantities that will have do delivered.
3. Find the product\_no and description of moving products.
4. Find out the names of clients who have purchased ‘CD DRIVE’
5. List the product\_no and s\_order\_no of customershaavingqty ordered less than 5 from the order details table for the product “1.44 floppies”.
6. Find the products and their quantities for the orders placed by ‘VandanSaitwal ’ and “Ivan Bayross”.
7. Find the products and their quantities for the orders placed by client\_no “ C00001” and “C00002”
8. Find the order No,, Client No and salesman No. where a client has been received by more than one salesman.
9. Display the s\_order\_date in the format “dd-mm-yy” e.g. “12- feb-96”
10. Find the date , 15 days after date.

**Theory & Concept**

**Practical # 6**

**Objective:- To implement the concept of grouping of Data.**

**Grouping Data From Tables:**

There are circumstances where we would like to apply the aggregate function not only to a single set of tuples, but also to a group of sets of tuples, we specify this wish in SQL using the group by clause. The attribute or attributes given in the group by clause are used to form group. Tuples with the same value on all attributes in the group by clause are placed in one group.

**Syntax:**

SELECT columnname, columnname

FROM tablename

GROUP BY columnname;

At times it is useful to state a condition that applies to groups rather than to tuples. For example we might be interested in only those branches where the average account balance is more than 1200. This condition does not apply to a single tuple, rather it applies to each group constructed by the GROUP BY clause. To express such Questionry, we use the having clause of SQL. SQL applies predicates in the having may be used.

**Syntax:**

SELECT columnname, columnname

FROM tablename

GROUP BY columnname;

HAVING searchcondition;

**Assignment No.6**

**Objective-Answer the following Queries:**

Q1.- Print the description and total quantity sold for each product.

Q2.- Find the value of each product sold.

Q3.- Calculate the average quantity sold for each client that has a maximum order value of

15000.

Q4.- Find out the products which has been sold to Ivan.

Q5.- Find the names of clients who have ‘CD Drive’.

Q6.- Find the products and their quantities for the orders placed by ‘Vandana’ and ‘Ivan’.

Q7.- Select product\_no, total qty\_ordered for each product.

Q8.- Select product\_no, product description and qty ordered for each product.

Q9.- Display the order number and day on which clients placed their order.

Q10.- Display the month and Date when the order must be delivered.

**Theory & Concept**

**Practical #7**

**Objective:- To implement the concept of SubQueries. SubQueries:- A** subQuery is a form of an SQL statement that appears inside anotherSQL statement. It also termed as nested Query. The statement containing a subQuery called a parent statement. The rows returned by the subQuery are used by the following statement.

It can be used by the following commands:

1. To insert records in the target table.
2. To create tables and insert records in this table.
3. To update records in the target table.
4. To create view.
5. To provide values for the condition in the WHERE , HAVING IN , SELECT,UPDATE, and DELETE statements.

Exam:-

Creating clientmaster table from oldclient\_master, table

Create table client\_master

AS SELECT \* FROM oldclient\_master;

**Using the Union, Intersect and Minus Clause:**

**Union Clause:**

The user can put together multiple Queries and combine their output using the union clause . The union clause merges the output of two or more Queries into a single set of rows and column. The final output of union clause will be

Output: = Records only in Query one + records only in Query two + A single set of records with is common in the both Queries.

Syntax:

SELECT columnname, columname

FROM tablename 1

UNION

SELECT columnname, columnname

From tablename2;

**Intersect Clause:** The use can put together multiple Queries and their output using theinterest clause. The final output of the interest clause will be :

Output =A single set of records which are common in both Queries

Syntax:

SELECT columnname, columnname

FROM tablename 1

INTERSECT

SELECT columnname, columnname

FROM tablename 2;

MINUS CLAUSE:- The user can put together multiple Queries and combine their output

* records only in Query one Syntax:

SELECT columnname, columnname FROM tablename ;

MINUS

SELECT columnname, columnname FROM tablename ;

**Assignment NO.7**

**Objective:** Answer the following Queries:

Question.

1. Find the product\_no and description of non- moving products.
2. Find the customer name, address, city and pincode for the client who has placed order no “019001”
3. Find the client names who have placed order before the month of may 96.
4. Find out if product “1.44 Drive” is ordered by only client and print the client\_no name to whom it was sold.
5. find the names of client who have placed orders worth Rs.10000 or more.
6. Select the orders placed by ‘Rahul Desai”
7. Select the names of persons who are in Mr. Pradeep’s department and who have also worked on an inventory control system.
8. Select all the clients and the salesman in the city of Bombay.
9. Select salesman name in “Bombay” who has atleast one client located at “Bombay”
10. Select the product\_no, description, qty\_on-hand,cost\_price of non\_moving items in the product\_master table.

**Theory and Concept**

**Practical # 8**

**Objective:- To implement the concept of Indexes and views.**

**Indexes-** An index is an ordered list of content of a column or group of columns in a table.An index created on the single column of the table is called simple index. When multiple table columns are included in the index it is called composite index.

Creating an Index for a table:-

**Syntax (Simple)**

CREATE INDEX index\_name

ON tablename(column name);

Composite Index:-

CREATE INDEX index\_name

ON tablename(columnname,columnname);

Creating anUniQuestion Index:-

CREATE UNIQUESTION INDEX indexfilename ON tablename(columnname);

Dropping Indexes:-

An index can be dropped by using DROP INDEX

**SYNTAX:-**

DROP INDEX indexfilename;

Views:-

Logical data is how we want to see the current data in our database. Physical data is how this data is actually placed in our database.

Views are masks placed upon tables. This allows the programmer to develop a method via which we can display predetermined data to users according to our desire.

Views may be created fore the following reasons:

1. The DBA stores the views as a definition only. Hence there is no duplication of data.
2. Simplifies Queries.
3. Can be Queried as a base table itself.
4. Provides data security.
5. Avoids data redundancy.

**Creation of Views:-**

**Syntax:-**

CREATE VIEW viewname AS

SELECT columnname,columnname

FROM tablename

WHERE columnname=expression\_list;

**Renaming the columns of a view:-**

**Syntax:-**

CREATE VIEW viewname AS

SELECT newcolumnname….

FROM tablename

WHERE columnname=expression\_list;

**Selecting a data set from a view-**

**Syntax:-**

SELECT columnname, columnname

FROM viewname

WHERE search condition;

**Destroying a view-**

**Syntax:-**

DROP VIEW viewname;

**Assignment No # 8**

**Objective: Answer the following Questions**

Q1. Create an index on the table client\_master, field client\_no.

Q2. Create an index on the sales\_order, field s\_order\_no.

Q3. Create an composite index on the sales\_order\_details table for the columns s\_order\_no and product\_no.

Q4. Create an composite index ch\_index on challan\_header table for the columns challan no and s\_order\_no.

Q5. Create anuniQuestion index on the table salesman\_master, field salesman\_no.

Q6. Drop index ch\_index on table challan\_header.

Q7. Create view on salesman\_master whose sal\_amt is less than 3500.

Q8. Create a view client\_view on client\_master and rename the columns as name, add1, add2, city, pcode, state respectively.

Q9. Select the client names from client\_view who lives in city ‘Bombay’.

Q10. Drop the view client\_view.

**References**

* Database System Concepts by A. Silberschatz, H.F. Korth and S. Sudarshan, 3rd edition, 1997, McGraw-Hill, International Edition.
* PL/SQL By Ivan Bayross
* Introduction to Database Management system by Bipin Desai, 1991, Galgotia Pub.
* Fundamentals of Database Systems by R. Elmasri and S.B. Navathe, 3rd edition, 2000, Addision-Wesley, Low Priced Edition.
* An Introduction to Database Systems by C.J. Date, 7th edition, Addison-Wesley, Low Priced Edition, 2000.
* Database Management and Design by G.W. Hansen and J.V. Hansen, 2nd edition, 1999, Prentice-Hall of India, Eastern Economy Edition.
* Database Management Systems by A.K. Majumdar and P. Bhattacharyya, 5th edition, 1999, Tata McGraw-Hill Publishing.
* A Guide to the SQL Standard, Date, C. and Darwen,H. 3rd edition, Reading, MA: 1994, Addison-Wesl

**Frequently Asked Questions**

* What is database?
* What is DBMS?
* What is a Database system?
* What are the Advantages and disadvantages of DBMS?
* What is a view? How it is related to data independence?
* What is E-R model?
* What is an Entity?
* What is an Entity set?
* What is DDL (Data Definition Language)?
* What is DML (Data Manipulation Language)?
* What is a query?
* What do you mean by Correlated subquery?
* What are the unary operations in Relational Algebra?
* Are the resulting relations of PRODUCT and JOIN operation the same?
* Define SQL and state the differences between SQL and other conventional programming Languages
* What is database Trigger?
* What are cursors give different types of cursors?
* What operator performs pattern matching?
* What operator tests column for the absence of data?
* Which command executes the contents of a specified file?
* What are the wildcards used for pattern matching?
* What are the privileges that can be granted on a table by a user to others?
* What command is used to get back the privileges offered by the GRANT command?
* Which system table contains information on constraints on all the tables created?
* What the difference is between TRUNCATE and DELETE commands?
* What will be the output of the following query? SELECT DECODE(TRANSLATE('A','1234567890','1111111111'), '1','YES', 'NO' );
* What does the following query do? SELECT SAL + NVL(COMM,0) FROM EMP;
* Which date function is used to find the difference between two dates?
* Which function is used to find the largest integer less than or equal to a specific value?
* What is the use of CASCADE CONSTRAINTS?
* What is the use of DESC in SQL?