

**Gujarat University**  
**Choice Based Credit System (CBCS) Syllabus for**  
**B. Sc. Semester IV (Computer Science) COM 204:**  
**System Development Tools – 2 (Theory)**

**Hours: 4 /week**

**Credits: 4**

**Unit: 1**

**1. Introduction to .NET Framework:**

- 1.1. .NET framework Architecture
- 1.2. .NET Framework Components (CLR, CLS, CTS, MSIL, NameSpace, JIT, MetaData, FCL, Assembly, GC, Memory Management)

**2. Basics of ASP.NET**

- 2.1. Features of ASP.NET
- 2.2. Difference between ASP.NET and Classic ASP
- 2.3. Web Applications and Webpage
- 2.4. Client Server Architecture

**Unit: 2**

**3. Creating simple Web Application in ASP.NET**

- 3.1. Introduction to Visual Studio
- 3.2. Creating a New Web Project
- 3.3. Opening an Existing Web Site
- 3.4. Building Web Sites
- 3.5. Set up of work environment, start page, the menu system, toolbars, the new project dialog box, graphical designer, code designer.

**4. Develop simple web page using built in Objects**

- 4.1. Adding Controls to the Web Page
- 4.2. Types of ASP.NET Files
- 4.3. Page Life Cycle
- 4.4. Web Form Processing Stages (Roundtrip)
- 4.5. ASP.Net In-Built Objects (Response, Request, Server, Trace Objects)

**Unit: 3**

**5. Web Server Controls**

- 5.1. Button, Check Box, Check Box List, Drop Down List, HyperLink, Image, Image Button, Label, Link Button, List Box, List Item, Panel, Place Holder, Radio Button, Radio Button List, Text Box
- 5.2. Working with Control Properties and Events
- 5.3. Validation Controls (Required Field Validator, RangeValidator Control, Compare Validator, RegularExpression Validator, CustomValidator, Validation Summary)

## **6. Apply Styles, themes and Master pages in ASP.NET Web applications.**

### 6.1. Styles

#### 6.1.1. Creating Style Sheets

#### 6.1.2. Applying Style Sheet Rules

### 6.2. Themes

### 6.3. Master Page

#### 6.3.1 Basics of Master page

#### 6.3.2 How Master page and Content pages are connected

## **Unit: 4**

## **7. Develop programs using session management and user's preference in ASP.NET**

### 7.1. State Management

#### 7.1.1 View State

#### 7.1.2 The Query String

#### 7.1.3 Cross-Page Posting and Validation

#### 7.1.4 Cookies (create, set, add and expire cookie)

#### 7.1.5 Session State

#### 7.1.6 Application State

### 7.2. ASP.NET Configuration

#### 7.2.1 The Web.config File

#### 7.2.2 Storing Custom Settings in the web.config file

## **8. Connecting Database with ADO.NET**

### 8.1. ADO.NET Architecture

#### 8.1.1 DataProvider

#### 8.1.2 Connection Object

#### 8.1.3 Command Object

#### 8.1.4 DataReader Object

#### 8.1.5 DataAdapter Object

#### 8.1.6 DataSet

#### 8.1.7 DataView

### 8.2. Data Binding

### 8.3. SQL Data Source

#### 8.3.1 Selecting, Updating and Deleting Records

## **Reference Books**

1. ASP.NET Bible By MridulaParihar, et al. – IDG Books India.
2. ASP.NET: The Complete Reference Books Matthew Macdonald McGraw Hill education.
3. ASP.NET Black Book By Kogent Solutions Inc. – Dreamtech Press
4. Building XML Web Services for ASP.NET By Bill Evjen – Wiley Dreamtech.

**Gujarat University**  
**Choice Based Credit System (CBCS) Syllabus for**  
**B. Sc. Semester IV (Computer Science)**  
**COM 205: Relational Database Management System (Theory)**

**Hours: 4 /week**

**Credits: 4**

**Unit – 1**

**1. Introduction to DBMS**

- 1.1 What is database?
- 1.2 Requirement of database system.
- 1.3 Data models and data independence.
- 1.4 DDL, DML
- 1.5 Database Manager, Database Administrator.

**2. Entity Relationship Models**

- 2.1 Entities and Entity sets
- 2.2 Relationship and relationship sets
- 2.3 Mapping constrains
- 2.4 Primary keys
- 2.5 Entity Relationship diagram and reducing it to tables.
- 2.6 Generation and Specialization.
- 2.7 Aggregation

**3. Relational Model**

- 3.1 Structure of relational database.
- 3.2 Relation algebra.

**Unit – 2**

**4. Introduction to other models**

- 4.1 Network Model
- 4.2 Hierarchical Model

**5. Relational Database Design**

- 5.1 Functional Dependencies
- 5.2 Referential Integrity
- 5.3 Need for Normalization
- 5.4 Normal forms
- 5.5 Data Dictionary
- 5.6 Tables, Table spaces & Data files, Views.

**6. Crash Recovery**

- 6.1 Failure Classification
- 6.2 Transactions

- 6.2.1 Transaction Concept, State.
- 6.2.2 Implementation of ACID rules.
- 6.2.3 Concurrent Executions.
- 6.2.4 Serializability, Recoverability.
- 6.3 Incremental log with differed and immediate updates.
- 6.4 Checkpoints
- 6.5 Buffer Management
- 6.6 Shadow Paging.
- 6.7 Concurrency Control.
  - 6.7.1 Lock Based Protocols.
  - 6.7.2 Timestamp-based Protocols.
  - 6.7.3 Validation based protocols.
  - 6.7.4 Deadlock Handling.

### **Unit – 3**

#### **7. PL/SQL**

##### 7.1 Overview of SQL

##### 7.2 Various types of data, conventions and terminology

##### 7.3 Retrieval of information from tables. :

Making a query, SELECT command, column reordering, Use of relational operators, use of Boolean operators, operations like IN, BETWEEN, LIKE, NULL, Not etc., Aggregate functions, COUNT, GROUP By clause, HAVING clause.

##### 7.4 Formatting Query output:

String and expressions, ordering output by fields, multiple columns, Aggregate Group, Column number, ORDER BY, with NULL.

##### 7.5 Querying multiple tables :

Joining tables through Referential Integrity, Equijoins and other kinds of joins of more than two tables, joining a table to itself.

### **Unit – 4**

##### 7.6 Subqueries :

DISTINCT with subqueries, Predicates with subqueries, Aggregate Functions in subqueries, Correlated subqueries, Correlating tables to itself, Correlated subqueries in HAVING, Correlated subqueries and joins EXISTS operator, using EXISTS with Correlated sub queries, combining EXISTS and joins, special operator ANY or SOME, ALL, UNION classes.

##### 7.7 Entering Deleting and Changing Field Values :

DML Update command, UPDATE with multiple columns, Updating to NULL values, INSERT command using sub queries with UPDATE commands.

#### 7.8 CREATE TABLE Command:

Indexing Altering a table, Dropping a table, Constraining a Table, Declaring Constraints, PRIMARY KEY constraint, Foreign and Parent keys, Multicolumn Foreign keys, FOREIGN KEY constraint, Foreign key restriction.

#### 7.9 CREATE VIEW Command:

Updating views, Group views and Joins, Views and subqueries, Changing values through views, Grant command, using ALL and PUBLIC arguments, GRANT OPTION.

#### **Recommended Reference Books:**

1. An introduction to Database Systems, C J Date, Addition-Wesley.
2. Database System Concepts, Abraham Silberschatz, Henry F. Korth & S. Sudarshan, McGraw Hill
3. SQL- PL/SQL by Ivan bayross
4. Understanding SQL by Martin Gruber, BPB
5. Oracle – The complete reference – TMH /oracle press

**Gujarat University**  
**Choice Based Credit System (CBCS) Syllabus for**  
**B. Sc. Semester IV (Computer Science)**  
**COM 206: Practicals (Based on ASP.NET, SQL and PL/SQL)**

**Hours: 6 /week**

**Credits: 2.5**

**ASP.NET Practical List**

1. Getting acquainted with Visual Studio environment. (Create new web project, open existing web project, building website, and study of toolbars, menu etc.)
2. Create a Simple calculator with validations and details.
3. Create a web page using the concept of cascading style sheets in ASP.NET.
4. Create a web page using the concept of Theme in ASP.NET.
5. Create Home page of your website using master page concept.
6. Develop a web page to implement the concept of state management using Session.
7. Create a login page in your web application. Login page must have user name and password fields. If user enters correct ID, Password, he must be redirected to the homepage of your website.
8. Create a web page to insert user biodata information with all validations into the database.
9. ADO.NET code to show records in Gridview control.
10. ADO.NET code to insert, update and delete records from database.

**PL/SQL**

- Using Variables, Constants & Data types
- User-Defined RECORD and TABLE data types.
- Assigning Database Values to variables  
SELECT.....INTO.....CURSORS.
- Using Flow Control Statements. The IF....THEN Statement, The LOOP statement WHILE loops, the GOTO statement.
- Error handling built in PL/SQL Exceptions, User – Defined Exceptions, Unhandled Exceptions.
- PL/SQL Programs  
Anonymous PL/SQL Blocks, procedures, Functions, Packages, Triggers.

## **SQL Practical List**

**Create following Three Tables.**

### **1. Salesman**

SNUM	SNAME	CITY	COMMITION
1001	PIYUSH	LONDON	12%
1002	NIRAJ	SURAT	13%
1003	MITI	LONDON	11%
1004	RAJESH	BARODA	15%
1005	ANAND	NEW DELHI	10%
1006	RAM	PATAN	10%
1007	LAXMAN	BOMBAY	09%

SNUM : A Unique numer assign to each salesman  
SNAME : The name of salesman CITY  
: The location of salesman COMMITION :The  
salesman commition on order

### **2. Customer**

CNUM	CNAME	CITY	RATING	SNUM
2001	HARDIK	LONDON	100	1001
2002	GITA	ROME	200	1003
2003	LAXIT	SURAT	200	1002
2004	GOVIND	BOMBAY	300	1002
2005	CHANDRESH	LONDON	100	1001
2006	CHAMPAK	SURAT	300	1007
2007	PRATIK	ROME	100	1004

CNUM : A Unique number assign to each customer.  
CNAME : The name of customer.  
CITY : The location of customer.  
RATING : A level of preference indicator given to this customer.  
SNUM : A salesman number assign to this customer.

### 3. Order

ONUM	AMOUNT	ODATE	CNUM	SNUM
3001	18.69	10/03/99	2007	1007
3002	767.19	10/03/99	2001	1001
3003	1900.10	10/03/99	2007	1004
3004	5160.45	10/03/99	2003	1002
3005	1098.25	10/04/99	2007	1007
3006	1713.12	10/04/99	2002	1003
3007	75.75	10/05/99	2004	1002
3008	4723.00	10/05/99	2006	1001
3009	1309.95	10/05/99	2004	1002
3010	9898.87	10/06/99	2006	1001

ONUM : A Unique number assign to each Order.  
AMOUNT : Amount of order in Rs.  
ODATE : The date of order.  
CNUM : The number of customer making the order.  
SNUM : The number of salesman credited with the sale.

#### Solve following request with the help of SQL query.

1. Produce the order no, amount and date of all orders.
2. Give all the information about all the customers with salesman number 1001
3. List out names and cities of all salesmen in London with commission above 10%
4. List all customers excluding those with rating  $\leq 100$  or they are located in Rome.
5. List all order for more than Rs.1000 except the orders of snum, 1006 of 10/03/99.
6. List all orders taken on 10<sup>th</sup> march or april 1999.
7. List all customers whose names begins with a letter 'C'.
8. List all customers whose names begins with letter 'A' to 'G'.
9. Find out the largest orders of salesman 1002 and 1007.
10. Count all orders of 10-Mar-99.
11. Count the no. of salesmen currently having order.
12. Find the largest order taken by each salesman on each date.
13. Find the largest order taken by each salasman on 10/03/1999.
14. Find out each customer's smallest order.

15. List all salesmen with their % of commission.
16. Display the no. of order for each day in the following format. FOR dd-mon-yy,there are orders.
17. Assume each salesperson has a 12% commission. Write a query on the order table that will produce the order number, salesman no and amount of commission for that order.
18. Find the highest rating in each city.
19. List all customers in descending order of rating.
20. Show the name of all customers with their salesman's name.
21. List all customers and salesmen who shared a same city.
22. List all orders with the names of their customer and salesman.
23. List all orders by the customers not located in the same city as their salesman.
24. List all customers serviced by salesman with commission above 12%.
25. List all customers located in cities where salesman Niraj has customers.
26. List all salesmen who are living in the same rating with our duplication.
27. Produce the name and city of all the customers with the same city as Hardik.
28. Extract all orders of Miti.
29. Extract all orders of Baroda's salesmen.
30. Find all orders of the salesmen who services 'Hardik'.
31. List all orders that are greater than the average of April 10, 1999.
32. Find all orders attributed to salesmen in 'London'.
33. List the commission of all salesmen serving customers in 'London'.
34. Find all customers whose cnum is 1000 above than the snum of Niraj.
35. List all orders of the customer 'Chandresh'.
36. Find all customers with orders on 10<sup>th</sup> March, 1999 using correlate sub query.
37. List the name and number of all salesmen who has more than zero customers.
38. Calculate the total amount ordered on each day eliminating the days where the total amount was not at least Rs. 2000 above the maximum amount of that day.
39. Using correlated sub query find the name and number of all customer with rating equal to maximum for their city.
40. Find all salesmen who have customers with rating > 300.
41. List all salesmen with customers located in their cities.
42. Find all salesmen for whom there are customers that follow them alphabetical order.
43. Find all customers having rating greater than any customer in 'Rome'.
44. List all orders that has amount greater than at least one of the orders from 10<sup>th</sup> June, 1999.
45. Find all orders with amounts smaller than any amount for a customer in 'Rome'.
46. Find all the customers who have greater rating than every customer in 'Rome'.

47. List all customers whose ratings are equal to or greater than ANY 'Niraj'.
48. Find out which salesmen produce largest and smallest orders on each date.
49. Create a union of two queries that shows the names, cities and ratings of all customers. Those with rating of  $\geq 200$  should display 'HIGH RATING' and those with  $< 200$  should display 'LOW RATING'
50. Insert a row into salesmen table with the values snum is 100 salesmen name is Rakesh, city is unknown and commission is 14%.
51. Insert a row in to customer table with values London, Pratik a 2005 for the columns city, name and number.
52. Create another table London staff having same structure as salesmen table.
53. Insert all the rows of salesmen table with city London in the London staff table.
54. Create another table Day totals with two attributes date and total and insert rows into this table from order table.
55. Create a duplicate of the salesmen table with a name Multicust. Now delet all the rows from the salesmen table.
56. Get back all the rows of salesmen table from its duplicate table.
57. Remove all orders from customer chandresh from the orders table.
58. Set the ratings of all the customers of Piyush to 400
59. Increase the rating of all customers in Rome by 100
60. Salesmen Miti has resigned. Reassign her number to a new salesmen Gopal whose city is Bombay and commission is 10%
61. Double the commission of all salesmen of London.
62. Set ratings for all customers in London to NULL.
63. Suppose we have a table called sales Manager with the same definition as Salesmen table. Company decides to promote salesmen having total order more than 5000 to sales Manager. Fill up the sales Manager table.
64. Assume that we have a table called smcity. Store the information of all salesmen with the customers in their home cities into smcity.
65. Create a table Bonus that contains date wise maximum amount of order for all salesmen.
66. Create a table Multicust containing the salesmen with more than one customer.
67. New Delhi office has closed. Remove all customers assigned to salesmen in New Delhi.
68. Delete all salesmen who have at least one customer with a rating of 100 from salesmen table.
69. Delete the salesmen who produce the lowest order for each day.
70. Find the smallest order for each day. Reduce the commission of all salesmen by 2% who produce this order.
71. Delete all customers with no current orders.
72. Write a command to find out the orders by date.
73. Write a command to add the item-name column to the order table.
74. Create a copy of your order table. Drop the original order table.
75. Write a command to create the order table so that all onum values as well as all combinations of cnum and snum are different from one another and so that NULL values are excluded from the date field.
76. Write a command to create the salesmen table so that default commission is 10% with no NULL permitted, snum is the primary key and all names contain alphabets only.
77. Give the commands to create our sample tables (salesmen, customer, orders) with all the necessary constraints like PRIMARY KEY , NOT NULL UNIQUE, FOREIGN KEY.

78. Create a view called Big orders which stores all orders larger than Rs. 4000.
79. Create a view Rate count that gives the count of no. of customers at each rating.
80. Create a view that shows all the customers who have the highest ratings.
81. Create a view that shows all the number of salesmen in each city.
82. Create a view that shows all the number of salesmen in each city.
83. Create a view that shows the average and total orders for each
84. Salesmen after his name and number.
85. Create a view that shows all the salesmen with multiple customers
86. Create a view to keep track of the total no of customers ordering, no of salesmen taking orders, the no of orders, the average amount ordered, and the total amount ordered for each day.
87. Create a view show name that shows for each order the order no, amount, salesmen name and the customer name.
88. List all orders of salesmen 'Rajesh' using show name view along with his commission.
89. Create a view Max sales to store the name and number of salesmen, along with the date, who have the highest order on any given date.
90. Using above view, find out the name and number of salesmen who have the highest order at least two times. Store the result in another view.
91. Create a view same city that shows the no and name and city of the customers along with the city of the salesmen serving them.
92. Create a view commission of salesmen table to include only snum and commission field so that through. This view someone can enter or change the commission but only to values between 10% and 20%.
93. Assume that the CURDATE is a constant representing current date. Give a command to create orders table with CURDATE as a default date.
94. List all salesmen in London who had at least one customer located there as well.
95. List all salesmen in London who didn't have any customer there.

## PL/SQL Programs

1. Write a PL/SQL block to find the given numbers is ODD or EVEN.
2. Write a PL/SQL block to find the factorial of given number.
3. Write a PL/SQL block to find the maximum number from given three numbers.
4. Write a PL/SQL block to find the sum of first 100 natural Nos.
5. Write a PL/SQL block to display the pyramid using any character.

## Procedure Programs

1. Write a procedure to check the given year is leap or not.
2. Write a procedure to display the following type of Multiplication Table as per given number.

```
5 * 1 = 5
5 * 2 = 10
“ “ = “
“ “ = “
5 * 10 = 50
```

3. Write a procedure to display this kind of output on screen.

```
1
2 3
3 4 5
4 5 6 7
5 6 7 8 9
..... 90 91
```

4. Write a procedure to convert given octal number to decimal number.

## Function Programs

1. Write a function to check whether the given number is prime or not.
2. Write a function to find the sum of digits of accepted no.
3. Write a function to display first 25 Fibonacci nos.
4. Write a function to display the reverse string of a given string.

## **Package Programs**

1. Write a package that has a function that check the given string is palindrome or not.
2. Write a package that has a procedure to find  $1 + 1/2 + 1/3 + \dots + 1/n$ .
3. Write a package that has a function to check given number is not negative and a procedure to convert the given number into word. For example 25 = Twenty Five.

## **Trigger Programs**

1. Write a trigger on insert to convert the name into capital letters.
2. Write a trigger to check the pin code is exactly six digits or not.
3. Write a trigger to check the mark is not zero or negative.



