

PART A: Introduction			
Program: Diploma	Class: B.C.A.	Year: II Year	Session:
Subject: Computer Applications			
1.	Course Code	UDIGTCM401	
2.	Course Title	Digital Marketing	
3.	Course Type (Core Course/Elective/Generic Elective/Vocational)	SEC	
4.	Pre-Requisite (if any)	To study this course, a student must have the basic knowledge of Computers.	
5.	Course Learning Outcomes(CLO)	1. Understanding the challenging needs of the society in the field of E- Commerce. 2. Understanding various activities and operations in the context of online transactions. 3. Applying the steps in surfing of e-commerce websites. 4. Understanding the working of arithmetic & logic unit 5. Applying the various e-payment systems. 6. Analyzing security issues in E-Commerce.	
6.	Credit Value	4 Credits	
7.	Total Marks	Max.Marks : 40+60	Min. Passing Marks: 16+24

PART B: Content of the Course		
Total No. of Lectures (in hours per week): 2 Hours per week		
Total Lectures: 30 Hours		
Unit	Topics	No. of Lectures
I	Introduction to E-Commerce Concepts and significance of E-commerce; Driving forces of E-commerce; E-commerce business models Key elements of a business model and categories; Design and launch of E-commerce website - Decisions regarding Selection of hardware and software; Outsourcing Vs in-house development of a website; Functions of E-Commerce; Types of E-Commerce; E-Commerce Systems and Prerequisites, Scope of E-Commerce.	8
II	E-Commerce Activities and Operations Various E-Commerce activities; Various manpower associated with e-commerce activities; Types of E-Commerce Providers and Vendors: Modes of operations associated with E-Commerce; E-commerce applications in various industries (banking, insurance, payment of utility bills and others), e-marketing, e-tailing, online services, e-auctions, online portal, online learning, e-publishing and e-entertainment, online shopping.	7
III	E-payment System E-payment Methods- Debit card, Credit card, Smart cards, E-Money, E-Wallets; Digital signatures- procedures and legal position; Payment gateways; Online banking- concepts, importance; Electronic fund transfer; Automated Clearing House. Automated Ledger Posting, Emerging modes and systems of E-payment (M-Paisa, PayPal and other digital currency), UPI Apps, Aadhar Enabled Payment Systems, BHIM App E-payments risks.	8
IV	Security and Legal Aspects of E-commerce E-commerce security meaning and issues. Security threats in the E-commerce environment- security intrusions and breaches, attacking methods like hacking, sniffing, cyber-vandalism etc.; Technology solutions- encryption, security channels of communication, protecting networks, servers and clients. Overview of Information Technology Act, 2000-provisions related to secure electronic records.	7

PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings:

Text Books:

1. E-commerce- Tulsiram Kundala, K.maheshwari- Himalya publication Nagpur
2. E-commerce- Dr. Sandeep Srivastava, Er. Meera Goyal - SBPD-New Delhi
3. E-commerce-Avriti Tangri- VK Global Publications Pvt
4. E-commerce-B. Bhardwaj Dr. RS, Garg-Galgotia Publishing Company
5. Essential of E-Commerce -M.K.Mallick - Sanjay Sahity Bhavan Agra
- 6 Business on the Net: An Introduction to the whats and hows of E-commerce. Noida, Uttar Pradesh: Agarwala, K. N., Lal, A., &Agarwala, D Macmillan Publishers India Limited.
7. Electronic commerce- A Manager's Guide to E- Business.- Diwan, P., & Sharma, S- Vanity Books International Delhi.

Suggested equivalent online courses: e-reading:

1. https://www.iare.ac.in/sites/default/files/lecture_notes/IARE_ECommerce_Lecture_Notes.pdf
2. https://oms.bdu.ac.in/ec/admin/contents/387_P16MCE4A_2020051801071611.pdf
3. <https://www.gaskovilpatti.com/studymaterial/commerce/I1%20MCOM%20E%20COMMERCE%20pKCM33.pdf>
4. <https://irp-cdn.multiscreensite.com/1c74f035/files/uploaded/introduction-to-e-commerce.pdf>
5. <https://backup.pondiuni.edu.in/storage/dde/dde ug pg books/E-%20Commerce.pdf>
6. <https://www.tutorialspoint.com/e-commerce/e-commerce-tutorial.pdf>

List of Practical

1. Help others to learn the use of e-wallet, e-payment, digital signatures. Prepare a report on the skills used by them to help others learn.
2. Use the internet banking facility to buy a product from any online website.
3. Open internet banking account and operate it.
4. Create their own YouTube channel and post one video on awareness of cyber security and crime
5. Clicking various E-Commerce websites and how to register and login on it.
6. Use the internet banking facility to buy a product from any online website.
7. Register the complaint regarding various issues i.e. refund, return, defective product and delay in delivery.

PART D: Assessment and Evaluation			
Internal Assessment : Continuous Comprehensive Evaluation (CCE) : 40 Marks Shall be based on allotted assignments and Class Tests. The marks shall be as follows:		External Assessment: University Exam (UE) : 60 Marks Time : 03.00 Hours	
Assessment and presentation of assignment	10 Marks	Section (A) : Five Very Short Questions (50 Words Each) OR MCQ Questions	05 x 02 = 10 Marks OR 10 x 01 = 10 Marks
Class Test I (Objective Questions)	10 Marks		
Class Test II (Descriptive Questions)	10 Marks	Section (B) : Five Short Questions (200 Words Each)	05 x 06 = 30 Marks
Class Test III (Based on Practicals)	10 Marks	Section (C): Two Long Questions (500 Words Each)	02 x 10 = 20 Marks
Total	40 Marks	Total	60 Marks
Any remarks/suggestions:			

PART A: INTRODUCTION			
Program: Diploma	Class: BCA	Year: II Year	Session:.....
Subject: Computer Applications			
1.	Course Code	UDATAACA402	
2.	Course Title	Database Management Systems Using PL/SQL	
3.	Course Type (Core Course/ Discipline Specific Elective/ Generic Elective/ Vocational)	Core Course (Theory)	
4.	Pre-Requisite (if any)	To study this course, a student must have the basic knowledge of Computers.	
5.	Course Learning Outcomes(CLO)	After completing this course student will be able to: • explain the features of database management systems and relational database. • design conceptual models of a database using ER modelling for real life applications and construct queries in relational algebra. • create and populate a RDBMS for a real-life application, with constraints and keys, using SQL. • retrieve any type of information from a database by formulating complex queries in SQL. • analyse the existing design of a database schema and apply concepts of normalization to design an optimal database.	
6.	Credit Value	4 credits (4-TH)	
7.	Total Marks	Max.Marks: 40+60	Min. Passing Marks:16+24
PART-B:CONTENT OF THE COURSE			
Total No. of Lectures-Tutorials-Practical (in hours per week): 4			
Total No. of Lectures: 60			
I	Introduction to DBMS: Why database? Characteristics of data in database, DBMS. What are database advantages of DBMS? Database Architecture and Modeling: Conceptual, physical and logical database models, Role of DBA, Database design. Entity Relationship (ER) Model: Components of ER-model, ER modeling symbols, Relationships. Enhanced Entity Relationship (EER) Model: An introduction, Superclass and subclass entity types, Specialization, Generalization, Attribute inheritance, Categorization & Aggregation. Keywords: DBMS, DBA, Entity Relationship (ER), EER, Superclass, Subclass, Specialization, Generalization, Categorization & Aggregation.		12

II	The Relational Data Model: Fundamental Concepts: Relations, Null Values, Keys, Foreign Keys, Integrity Constraints - Entity Integrity & Relational Integrity. Normalization Process: First Normal Form, Functional Dependencies, Second Normal Form, Third Normal Form, Boyce-Codd Normal Form (BCNF), Fourth Normal Form; Other Normal Forms - Fifth Normal Form & Domain/Key Normal Form. Transforming a Conceptual Model to a Relational Model: Transforming Objects Sets and Attributes, Transforming Models without External Keys, Transforming Specialization and Generalization Object Sets, Transforming Relationships: One-One Relationships, One-Many Relationships, Many-Many Relationships; Transforming Aggregated Object Sets, Transforming Recursive Relationships. Keywords: Keys, Normalization, BCNF, Aggregated Object Sets, Recursive Relationships.	12
III	Relational database implementation: (a) Relational Algebra and Calculus Relational Algebra: Union, Intersection, Difference, Product, Select, Project, Join - Natural, Theta & Outer Join, Divide, Assignment. Relational Calculus: Target list & Qualifying Statement, The Existential Quantifier, The Universal Quantifier. Keywords: JOIN, Target list, Existential Quantifier, Universal Quantifier.	12
IV	Relational database implementation (continued): (b) Relational Implementation with SQL <i>Relational Implementations:</i> An Overview. <i>Schema and Table Definition:</i> Schema definition, Data types & domains, Defining Tables, Column Definition. <i>Data Manipulation:</i> Simple Queries (SELECT, FROM, WHERE), Multiple-Table Queries, Subqueries, Correlated Subqueries, EXISTS and NOT EXISTS operators, Built-In Functions (SUM, AVG, COUNT, MAX, and MIN), GROUP BY and HAVING clause, Built-In Functions with Subqueries. <i>Relational Algebra Operations:</i> UNION, INTERSECT, EXCEPT, JOIN. <i>Database Change Operations:</i> INSERT, UPDATE, DELETE. Using SQL with Data Processing Languages; View Definition, Restrictions on View Queries and Updates. Keywords: Schema, SELECT, Data Manipulation, Database Change Operation, View.	12

V	Physical Database Systems Introduction, Physical Access of the Database. <i>Physical Storage Media:</i> Secondary Storage, Physical Storage Blocks. <i>Disk Performance Factors:</i> Access Motion Time, Head Activation Time, Rotational Delay, Data Transfer Rate, Data Transfer Time. <i>Data Storage Formats on Disk:</i> Track Format, Record Format—Fixed-Length Records	12
---	--	----

& Variable-Length Records, Input/output Management. <i>File Organizing and Addressing Methods:</i> Sequential File Organization, Indexed-Sequential File Organization, Direct File Organization, Hashing: Static Hash Functions and Dynamic Hash Functions. <i>Keywords:</i> Disk Performance Factors, Sequential File Organization, Indexed-Sequential File Organization, Direct File Organization, Hashing
--

PART C: LEARNING RESOURCES

Textbooks, Reference Books, Other Resources

Suggested Readings:

Textbooks:

1. Gary W. Hansen & James V. Hansen, "Database Management and Design", 2nd Ed., 2007, Prentice Hall of India Pvt Ltd.
2. Instructional Software Research & Development (ISRD) Group, Lucknow "Introduction to Database Management Systems", 2006, Ace Series, Tata McGraw Hill Publishing Company Limited, New Delhi
3. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", 7th Edition, 2016, Pearson

Reference Book:

1. Raghu Ramakrishnan & Johannes Gehrke, "Database Management Systems", 3rd Edition, 2014, McGraw Hill Education
2. C.J. Date, "An Introduction to Database System", 8th Edition, 2003, Pearson
3. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", 6th Edition, 2010, Tata McGraw Hill
4. Books published by M.P. Hindi Granth Academy, Bhopal

Suggestive digital platform web links

[http://en.wikipedia.org/wiki/Relationalmodel](http://en.wikipedia.org/wiki/Relational_model)

[http://en.wikipedia.org/wiki/Relational algebra](http://en.wikipedia.org/wiki/Relational_algebra)
cs.nyu.edu

<https://www.w3school/dbms/database-normalization/>

<https://ecomputernotes.com/>

<http://www.mphindigranthacademy.org/>

Suggested equivalent online courses

NPTEL Course: *INTRODUCTION TO DATABASE SYSTEMS* or *DATABASE DESIGN*

PART D: Assessment and Evaluation

Internal Assessment : Continuous Comprehensive Evaluation (CCE) : 40 Marks Shall be based on allotted assignments and Class Tests. The marks shall be as follows:		External Assessment: University Exam (UE) : 60 Marks Time : 03.00 Hours	
Assessment and presentation of assignment	10 Marks	Section (A) : Five Very Short Questions (50 Words Each) OR MCQ Questions	05 x 02 = 10 Marks OR 10 x 01 = 10 Marks
Class Test I (Objective Questions)	10 Marks		
Class Test II (Descriptive Questions)	10 Marks	Section (B) : Five Short Questions (200 Words Each)	05 x 06 = 30 Marks
Class Test III	10 Marks	Section (C): Two Long Questions (500 Words Each)	02 x 10 = 20 Marks
Total	40 Marks	Total	60 Marks
Any remarks/suggestions:			

PART A: Introduction			
Program: Diploma	Class: BCA	Year: II Year	Session:
Subject: Computer Application (BCA)			
1.	Course Code	UINTECA403	
2.	Course Title	Internet of Things (IOTs)	
3.	Course Type (Core Course/ Elective/ Generic Elective/ Vocational)	Generic Elective	
4.	Pre-Requisite (if any)	Student must have basic Computer Knowledge	
5.	Course Learning Outcomes (CLO)	After completing this course student will be able to: 1.To understand the basics of Internet of Things 2. To get an idea of some of the application areas where Internet of Things can be applied 3. To understand the middleware for Internet of Things and the concepts of Web of Things 4. To understand the concepts of Cloud of Things with emphasis on Mobile cloud computing 5 To understand the IOT protocols	
6.	Credit Value	Theory —4 Credits Practical — 2 Credits	
7.	Total Marks	Max. Marks: 40+60	Min. Passing Marks: 16+24

PART B: Content of the Course		
No. of Lectures (in hours per week): 2 Hrs. per week		
Total No. of Lectures (in hours): 60 Hrs.		
Module	Topics	No. of Lectures
I	Introduction Introduction: Definition, Characteristics of IOT, IOT Conceptual framework, IOT Architectural view, Physical design of IOT, Logical design of IOT, Application of IOT.	8
II	Machine-to-machine (M2M), SDN (software defined networking) and NFV (network function virtualization) for IOT, data storage in IOT, IOT Cloud Based Services.	14
III	Design Principles for Web Connectivity: Web Communication Protocols for connected devices, Message Communication Protocols for connected devices, SOAP, REST, HTTP Restful and Web Sockets. Internet Connectivity Principles: Internet Connectivity, Internet based communication, IP addressing in IOT, Media Access control.	14
IV	Sensor Technology , Participatory Sensing, Industrial IOT and Automotive IOT, Actuator, Sensor data Communication Protocols, Radio Frequency Identification Technology, Wireless Sensor Network Technology.	12
V	IOT Design methodology: Specification - Requirement, process, model, service, functional & operational view. IOT Privacy and security solutions, Raspberry Pi & arduino devices. IOT Case studies: smart city streetlights control & monitoring.	12

PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings

Textbooks:

- Rajkamal, "Internet of Things", Tata McGraw Hill publication.
- HakimaChaouchi "The Internet of Things: Connecting Objects", Wiley publication.
- Francis dacosta "Rethinking the Internet of things: A scalable Approach to connecting everything", 1st edition, Apress publications2013.
- Donald Norris "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black", McGraw Hillpublication.

Reference books:

1. Philip Levis, " TinyOS Programming"
2. D. Norris, "The Internet of Things: Do-It-Yourself Projects with Arduino, Raspberry Pi, and Beagle Bone Black", McGraw-Hill Education, New Delhi.
3. Raj Kamal, "Internet of Things: Architecture and Design", Tata McGraw Hill publication.
4. A. Pajankar and A. Kakkar, "Raspberry Pi by Example", Packt Publishing Ltd, Birmingham,UK.
5. Books published by M.P. Hindi Granth Academy, Bhopal

Suggestive digital platform web links

- <https://www.iotforall.com/introduction-iot-applications-in-education>
- https://onlinecourses.swayam2.ac.in/arp19_ap52/preview
- <http://www.mphindigranthacademy.org/>

PART D: Assessment and Evaluation			
Internal Assessment : Continuous Comprehensive Evaluation (CCE) : 40 Marks Shall be based on allotted assignments and Class Tests. The marks shall be as follows:		External Assessment: University Exam (UE) : 60 Marks Time : 03.00 Hours	
Assessment and presentation of assignment	10 Marks	Section (A) : Five Very Short Questions (50 Words Each) OR MCQ Questions	05 x 02 = 10 Marks OR 10 x 01 = 10 Marks
Class Test I (Objective Questions)	10 Marks		
Class Test II (Descriptive Questions)	10 Marks	Section (B) : Five Short Questions (200 Words Each)	05 x 06 = 30 Marks
Class Test III	10 Marks	Section (C): Two Long Questions (500 Words Each)	02 x 10 = 20 Marks
Total	40 Marks	Total	60 Marks
Any remarks/suggestions:			

PART A: Introduction			
Program: Diploma	Class: BCA	Year: II Year	Session:
Subject: Computer Application (BCA)			
1.	Course Code	URURALCM404	
5.	Course Title	Rural Development in India	
6.	Course Type (Core Course/ Elective/ Generic Elective/ Vocational)	Generic Elective	
7.	Pre-Requisite (if any)	Student must have basic Computer Knowledge	
6.	Course Learning Outcomes (CLO)	C01: The Student will be able to know the problems of rural development along with the strategy of rural development. CO2. Students will be able to get Information about various dimensions of rural development. CO3.The student will be familiar with the role of agriculture and industries in rural development. C04.Students will be able to get information about various schemes and policies of the Government regarding rural development. CO5. Students will develop an understanding of the contribution of Panchayati raj to rural development.	
8.	Credit Value	Theory—4 Credits	
9.	Total Marks	Max. Marks: 40+60	Min. Passing Marks: 16+24

PART B: Content of the Course		
No. of Lectures (in hours per week): 2 Hrs. per week		
Total No. of Lectures (in hours): 60 Hrs.		
Unit	Topics	No. of
Unit-I	Basics of Rural Development: I Concept and Objectives of Rural Development, Significance of Rural Development, Indicators of Rural Development, Rural Development Strategies: Aims & Objectives, Problems of the rural development in India -(Poverty, indebtedness & Unemployment), Major Initiatives for inclusive growth in rural India. Keywords : Rural Development , Indicators of Rural Development, Strategy, Poverty, indebtedness, Unemployment, inclusive growth.	12
Unit-II	Infrastructure Development in Rural India: II Social Infrastructure - Education, Health, Women and Child Welfare, livelihood mission, sanitation: physical Infrastructure - Irrigation, Energy, Road, Transportation. Role of infrastructure in Rural Development. Keywords: Social Infrastructure, livelihood, sanitation, physical Infrastructure, Irrigation, Energy, Road, Transport.	12
Unit-III	Agriculture Development: III Role of Agriculture in Rural Development, Characteristics and nature of Indian Agriculture, Green Revolution in Agriculture, Agricultural Production and Productivity, Agriculture Growth in India, Land Utilization, Current Agricultural Policy, Land reforms, Role of MSMEs & Self-Help Groups in Rural Development. Keywords: Green Revolution, Agriculture Productivity, Land Utilization, Agricultural Policy, Micro Small Medium Enterprise, Self Help Groups.	12
Unit-IV	Programme and Policies of Rural Development: IV An overview of Current Rural Development Programmes: Pradhan Mantri Kaushal Vikas Yojana (PMKVY), Swachhh Bharat Mission (SBM), Sansad Adarsh Gram Yojna (SAGY), Mahatma Gandhi National Rural Employment Guarantee Scheme (MGNREGS), Pradhan Mantri Gramin Awaas Yojana (PMGAY), Pradhan Mantri Gram Sadak Yojana (PMGSY), Shyama Prasad Mukherji Rurban Mission (SPMRM), Deendayal Antyodaya Yojana — National Rural Livelihood Mission (DAYNRLM), Deendayal Upadhyaya Grameen Kaushalya Yojana (DDU-GKY), Pradhan Mantri Ujjawala Yojna (PMUY), Pradhan Mantri Fasal Bima Yojna (PMFBY), Atal Pension Yojana (APY) Keywords : Programme, Policies, Overview of Present Policies, Rural Development.	12

Unit-V	Role of Panchayati Raj in Rural Development: Concept, structure and functions of Panchayati Raj Institutions, Concept and importance of Decentralization in rural development, Constitutional Provisions regarding Panchayati Raj (Including 73rd constitutional amendment), Role of Panchayats in Resources Generation, Poverty Alleviation and Women Empowerment, Linkage between Panchayat and NGOs. Challenges of Panchayati Raj institutions. Keywords: Panchayati Raj, Constitutional Provisions, Women Empowerment, Poverty Alleviation, Resources Generation, NGOs.	12
--------	---	----



PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings:

1. Abdul Azeed, N.P.S Akhtar Jawed S.M. - Rural Development in India" (Policies and programmes) kapaz publication Ashok vihas, New Delhi 2016.
2. Gupta K.R. - Rural Development in India" Atlantic publishers and Distributions, New Delhi 2010.
3. Dr. Nath V. - "Rural Development and Planning in India" Concept Publishing Co. 2010.
4. Pareek,Udai - Education and Rural Development D.G.M. Infai Book Publication 2005.
5. Sinha, Kumar Rajesh - "Sustainable development goals and panchayat publication - Indigo Books 2019.
6. Bilgrami S.A. "An Introduction to agriculture Economics" Himalaya Publishing House Mumbai.
7. Gupta, U.C.S. Verna Pratik " Gramin Vikas pariyojana (Hindi) Raj publications, New Delhi 2013.
8. Dewan, M.L. - "Agriculture and Rural Development in India" concept Publishing Co. (English).
9. Singh, Ram Sakal - " Rural Development Administration" - Anmol Publications Pvt. Ltd. 2006.
10. Dhingra I.C. Dec 2019
11. Prasad B.K. - Sasup & Sons New Delhi 2003.
12. Agrawal A.N. 37th Edition 2011 New Age International Publishers.

Hindi

1. सिंह कटार, सिसोदिया, यतीन्द्र सिंह - " ग्रामीण विकास सिद्धान्त , नीतियों एवं प्रबन्ध (हिन्दी भाषा) Sage Publication Pvt. Ltd. फरवरी 208,
2. सिंह , धर्मेन्द्र - " पंचायती राज एवं ग्रामीण विकास " रावत पब्लिशन, नई दिल्ली 2077.
3. सिंह , इकबाल - " भारत में ग्रामीण विकास " Olympia Publications , राष्ट्रीय शैक्षिक अनुसंधान और प्रशिक्षण परिषद्
- 4 . बाबेल , डॉ. बसंती लाल - "पंचायती राज और ग्रामीण विकास की योजना, राजस्थान हिन्दी ग्रन्थ अकादमी
- 5 . गुप्ता, यू.सी एण्ड वर्मा, प्रतीक - " ग्रामीण विकास परियोजना " प्रगुन पब्लिकेशन 2012,
6. अग्रवाल अनुपम एवं रंजन सौम्या " अर्थशास्त्र " एस.वी.पी.डी. पब्लिकेशन आगरा 2020 बुक कोड -4945
7. महेश्वरी पी.डी. एवं गुप्ता शील चन्द्र " भारतीय आर्थिक नीति" कैलाश पुस्तक सदन भोपाल 2014
8. मिश्रा एस.के. एवं पुरी वही. के. " विकास का अर्थशास्त्र और नियोजन" हिमालय पब्लिकेशन हाउस मुम्बई 204
- 9 . दत्त , सन्दरम एवं महाजन "भारतीय अर्थव्यवस्था" एस.चन्द्र. पब्लिकेशन नई दिल्ली 2018

Suggestive digital platform web links

1. <https://rural.nic.in/>
2. [https://www.ilo.org/global/lang -- en/index.htm](https://www.ilo.org/global/lang--en/index.htm)
3. <https://www.india.gov.in/official-website-ministry-rural-development-0>
4. www.researchgate.net
5. www.oecd.org/regional
6. <http://niti.gov.in/>

PART D: Assessment and Evaluation			
Internal Assessment : Continuous Comprehensive Evaluation (CCE) : 40 Marks Shall be based on allotted assignments and Class Tests. The marks shall be as follows:		External Assessment: University Exam (UE) : 60 Marks Time : 03.00 Hours	
Assessment and presentation of assignment	10 Marks	Section (A) : Five Very Short Questions (50 Words Each) OR MCQ Questions	05 x 02 = 10 Marks OR 10 x 01 = 10 Marks
Class Test I (Objective Questions)	10 Marks		
Class Test II (Descriptive Questions)	10 Marks	Section (B) : Five Short Questions (200 Words Each)	05 x 06 = 30 Marks
Class Test III	10 Marks	Section (C): Two Long Questions (500 Words Each)	02 x 10 = 20 Marks
Total	40 Marks	Total	60 Marks
Any remarks/suggestions:			

PART A: INTRODUCTION			
Program: Diploma		Class: BCA	Year: II Yr
Subject: Computer Applications			
1.	Course Code	UDATACA405	
2.	Course Title	DBMS Using PL/SQL Lab	
3.	Course Type (Core Course / Discipline Specific Elective / Generic Elective / Vocational)	Core Course (Practical)	
4.	PreRequisite (if any)	To study this course, a student must have the basic knowledge of Computers.	
5.	Course Learning Outcomes (CLO)	This lab is based on the theory course of DBMS involves the development of the practical skills in DBMS using MS-Access/Visual-FoxPro/SQL-Server/etc. this course is an attempt to upgrade and enhance students' theoretical skills and provide the hands-on experience. After completing this lab course sessions, students will be able • to create Databases & Views, • execute simple & advance SQL queries, • use DBMS tools in the areas of database application Topics to be covered in the lab syllabus • Introduction to MS-Access/Visual-FoxPro/SQL-Server/etc • Hands on practice on the application package (i.e. on MS-AccessVisual-FoxPro/SQL-Servedetc) • Database creation using MS-Access/Visual-FoxPro/SQL-Server/etc • Simple SQL queries (Single table retrieval) • Use of Advanced SQL queries • Implementation of Views	
6.	Credit Value	2 credits (2-PR)	
7.	Total Marks	Max. Marks: 40 Int + 60 Ext	Min. Passing Marks: 16+24
PART B: CONTENT OF THE COURSE			
Total No. of Lectures-Tutorials-Practical (in hours per week): P - 2			

Practical details:

Students are required to practice the concepts learnt in the theory by designing and querying a database for a chosen organization (Like: College, Library, Transport, etc). The teacher may devise appropriate weekly lab assignments to help students practice the designing, querying a database in the context of example database. Some indicative list of experiments with their aim, problem definition, theory is given below:

Experiment-1

Aim: To draw ER Model and Relational Model for a given database. Show ER to Relational Model reduction.

Resources used: MS-Access/Visual-FoxPro/SQL-Server/etc

Problem Definition: List the data requirements for the database of the company which keeps track of the company employee, department and projects. The database designers provide the following description:

1. The company is organized into departments. Each department has unique name, unique number, and particular employee to manage the department. We keep track of the start date and the employee begins managing the department. The department has several locations.

The department controls a number of projects each of which has a unique name, unique number and a single location.

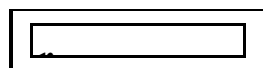
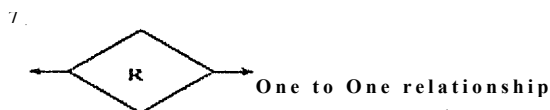
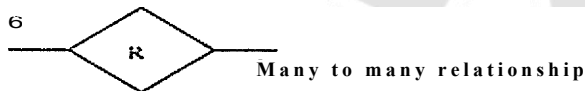
3. We store each employee names social security number, address, salary, sex and dob. An employee is assigned one department but may work on several projects which are not necessarily controlled by the same department. We keep track of the department of each employee works on each project and for insurance purpose. We keep each dependent's first name, sex, dob and relation.

Theory: The ER data model was developed to facilitate the database design by allowing specification of an enterprise schema that represents the overall logical structure of the database. The ER model data model is one of the several data models. The semantic aspect of the model lies in its representation of the meaning of the data. The ER model is very useful many database design tools drawn on concepts from the ER model. The ER model employs 3 basic notations: entity set, relationship set and attributes.

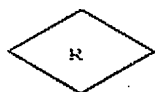
Symbols Used in ER Notation

1

Entity set: An entity is a set of objects of the same type that share the properties or attributes.

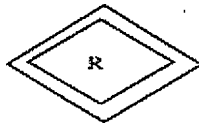


Weak entity set: An entity set may not have sufficient attributes to form a primary key. Such an entity set is termed as weak entity set.



Relationship Set A relationship is an association between several entities. A relationship set is a set of relationships of the same type.

4

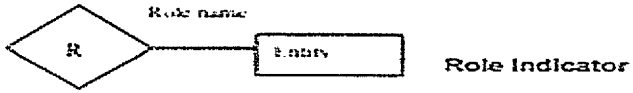


Identification relationship set for weak entity set :
The relationship associating the weak entity set '....'th the identifying entity set is called the identifying relationship.

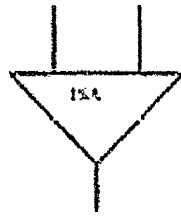
5



Primary key : The primary key is used to denote a candidate key that is chosen by the database designers as the primary means of identifying entities within an entity set.



9.



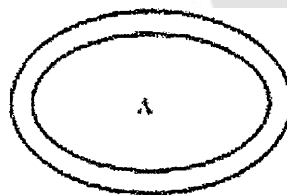
Total Generalization

10.



Attribute

11.



Multi valued Attribute

12.



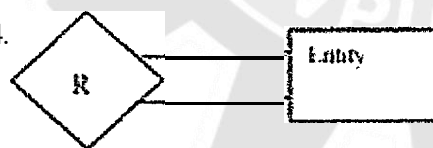
Derived Attribute

13.



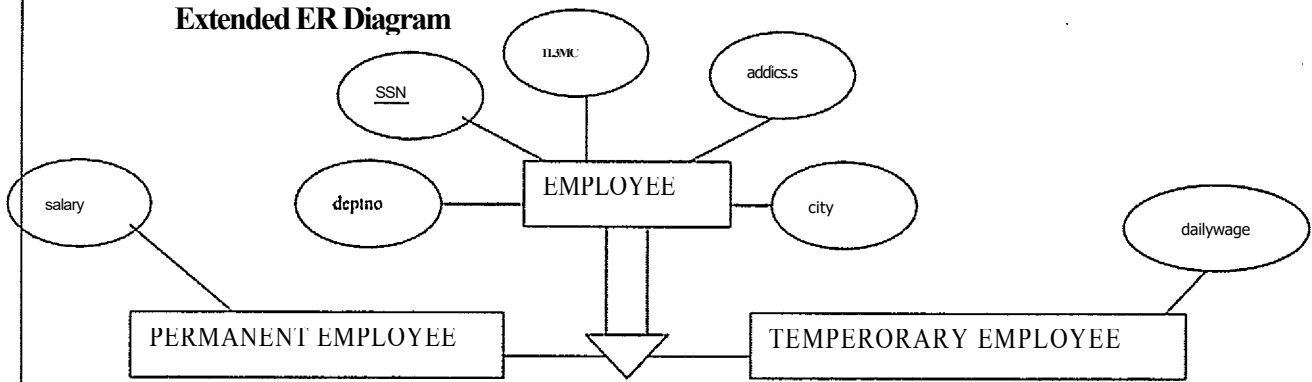
Discriminating Attribute of weak entity set : The dischntrotiori of weak entity set is a set of atUibutes that allows the distindion to be made.

14.

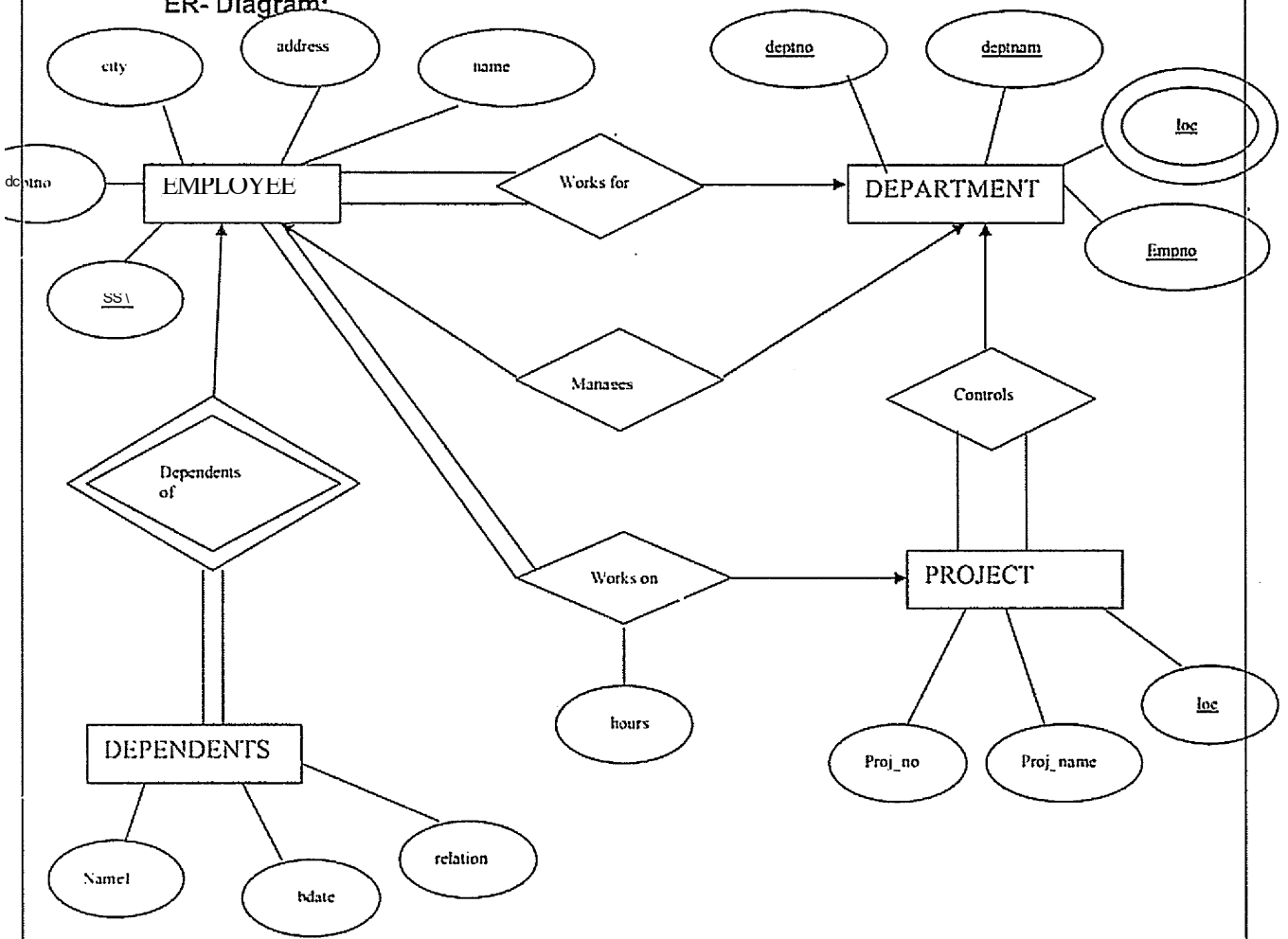


Total Participation of entity set In relationship: The participation of an entity set E in a rela6onship set R is said to be total if every entity in E participates at least one relationship in R

Extended ER Diagram



ER- Diagram:



Relational Model:

Employee

<u>fname</u>	<u>SSN</u>	<u>address</u>	<u>salary</u>	<u>city</u>	<u>deptno</u>

Department

<u>deptno</u>	<u>deptname</u>	<u>mgr SSN</u>

Department Location

<u>deptno</u>	<u>deptloc</u>

Project

<u>projno</u>	<u>projname</u>	<u>location</u>	<u>deptno</u>

Works on

<u>SSN</u>	<u>hours</u>	<u>projname</u>

Dependents

<u>namel</u>	<u>relation</u>	<u>bdate</u>	<u>SSN</u>

Employee

<u>SSN</u>
fname
address
salary
city
deptno

Department

<u>deptno</u>
deptname
mgr SSN

Department Location

<u>deptno</u>
deptloc

Project

<u>projno</u>
projname
<u>location</u>
<u>deptno</u>

Works on

<u>SSN</u>
hours
projno

Dependents

<u>namel</u>
<u>relation</u>
<u>bdate</u>
<u>SSN</u>

Conclusion: We have drawn ER model and Relational Model for the same.

Experiment-2

Aim: Implementation Database

1. Creation of Database with proper constraints (Pk, Fk,etc)
2. Insert into database using different types of insert statements
3. Display

Resources used: MS-Access/Visual-FoxPro/SQL-Server/etc

Theory: The set of relations in 'a database must be specifies to the system by means of a data definition language (DDL). The SQL DDL allows specification of not only a set of relations but also specific information about the relation including:

1. The schema for each relation
2. The domain of values associated with each attribute
3. The integrity constraints
4. The set of indices to be maintained for each relation
5. The security and authorization information for each relation
6. The physical storage structure of each relation on disk

Create Table

create table tab (A1D1, A2D2, ..., Ana, <integrity constraint-I>,<integrity constraint-k>)

where tab is the name of the relation each Ai is the name of the attribute in the schema of relation tab and Di is the domain type of the values in the domain of attribute A. There are a number of different allowable integrity constraints. We specify here only the primary key for the relation.

Insert

A newly created relation is empty initially. We can use the insert command to load data into the relation.

insert into <table na/ne> values (A1, 242, ..., Aid)

The values are specified in the order in which the corresponding attributes are listed in the relation schema.

Display

To display the table after creation and insertion we use the following syntax:

*select * from <table name>*

Select clause is used to list the attributes desired in the result of a query. It corresponds to the projection operation of the relational algebra. From clause lists the relations to be scanned in the evaluation of the expression. The asterisk symbol ("*") is used to denote "all attributes".

Conclusion

Thus, we have successfully created the database of company and inserted values in the database.

Experiment-3

Aim: Data Definition (schema) Modification

1. Alter table: add column, remove column, add constraint, remove constraint
2. Drop table
3. Show schema of any table
4. Applying different constraints check, not null, etc.

Resources used: MS-Access/Visual-FoxPro/SQL-Server/etc

Theory: The various command, clauses, functions used for the modification of database are as follows:

- (1) **Alter table:** Alter table command is used to add attribute to an existing relation. All the tuples are assigned to null as the values for the new attribute. The form of the alter table command is

Alter table r add A D

Where, *r* is the name of an existing relation. *A* is the name of the attribute to be added and *D* is the domain of the added attribute. We can drop attribute from a relation by the command:

Alter table r drop A

- (2) **Update:** In certain situation we may wish to change a value in a tuple without changing all values in the tuple. For this purpose, the update statement can be used, as we could for insert and delete. We can choose the tuple be updated by using a query.

*eg, update EMPLOYEE
set age=20
where SSN=514065*

The preceding update statement is applied only to tuple where SSN=514065. If we want same changes in all tuples, then we write

*Update EMPLOYEE
set age=20*

- (3) **Drop Table:** To remove a relation from an SQL database we use the drop table command. The drop table command deletes all information about the dropped relation from the database

drop table r

The relation *r* and to delete all tuples from *r*, the following command is used.

delete from r

- (4) **Adding and Removing Columns:** To add a column to an existing relation, we use

*alter table r
add A D*

*eg. alter table EMPLOYEE
add age int*

To remove a column from an existing relation we use

*Alter table r
drop column A*

*Eg. alter table EMPLOYEE
drop column age*

(5) *Not Null*: The not null specification prohibits the insertion of a null value. For a attribute any database modification that would cause null to be inserted in an attribute declared to be not null generates an error diagnostic. If an attribute is declared as the primary key then it cannot take a null value.

Eg, *alter table EMPLOYEE*
alter column salary int NOT NULL

(6) *Check*: The check clause in SQL can be applied to relation declarations as well to domain declarations when applied to a relation declaration, the clause *check(p)* specified a predicate p that must be specified by every tuple in a relation. A common use of the check clause is to ensure that the attribute value satisfy specified condition.

Eg, *alter table EMPLOYEE*
add constraint em_age
check (age > 19)

Conclusion: Thus, we have executed all the queries required for the modification of database.

Experiment4

Aim: Simple SQL queries (Single table retrieval)

1. Make use of different operators (relational, logical etc.)
2. Selection of rows and columns, renaming columns, use of distinct keyword
3. String handling (% , etc.)
4. Update statement, case update
5. Delete, cascade delete (if possible)

Resources used: MS-Access/Visual-FoxPro/SQL-Server/etc

Theory:

I. Select clause: Select clause is used to list the attributes desired in the result of a query. It corresponds to the projection operation of the relational algebra:

Eg. *select *from EMPLOYEE*
all attributes
select fname, SSN from EMPLOYEE
-only fname and SSN

2. from clause: From clause lists the relations to be scanned in the evaluation of the expansion.

3. where clause: The where clause corresponds to the selection predicate of the relational algebra. It consists of a predicate involving attribute of the relations that appear in the from clause.

(i) *and*: *and* clause is used when we want a result and all the conditions are satisfied in the where clause.

True and unknown = true

False and unknown = unknown

Unknown Or unknown = unknown

(ii) *as (Rename operator)*: SQL provides a mechanism for renaming both relations and attributes. It uses the *as* clause taking the form

old name as new name

(iii) *distinct*: If we want to eliminate duplicates, we use the keyword *distinct* in the aggregation expression.

eg. *select distinct salary*
from EMPLOYEE

(iv) *String operations*: The most commonly used operations on strings are pattern matching using the operation like we describe the patterns by using the two special characters % and

%: The % character matches any substring

_ : The character matches any character

eg, 'Perry%' matches any string beginning with "Perry".

`%idge%' matches any string containing "idge" as substring

___ " matches any string of exactly three characters

——— %" matches any string of at least three characters

(v) *Update and Case Update*: In certain situations, we may wish to change a value in a tuple without changing all the values in the tuple. For this purpose, the update statement can be used.

eg. *update EMPLOYEE*

set age=20

where SSN=514065

SQL provides a case construct which we can use to perform both the update with a single update statement avoiding the problem with the order of updates.

eg. *update account*

set balance =case

when balance<=1000

*then balance *1.05*

*else balance*1.06 end*

(vi) *delete*: To delete a tuple from relation r, we use the following command

delete from r

where, r is the name of the relation

Conclusion: Thus, we have executed simple queries in SQL.

Experiment-5

Aim: Advanced SQL Queries-1

1. Group by, having clause, aggregate function
2. Set operations like union, union all and use of order by clause
3. Nested queries: in, not_in, exists, not exists and any, all

Resources used: MS-Access/Visual-FoxPro/SQL-Server/etc

Theory:

1. *Group by clause*: These are circumstances where we would like to apply the aggregate functions to a single set of tuples but also to a group of sets of tuples, we would like to specify this wish in SQL using the group by clause. The attributes or attributes given by the group by clause are used to form groups. Tuples with the same value on all attributes in the group by clause placed in one group:

eg.

select dept_no, avg(sal) as avg_sal

from EMPLOYEE

group by dept_no

2. *Having clause*: A having clause is like a where clause but only applies only to groups as a whole whereas the where clause applies to the individual rows. A query can contain both where clause and a having clause. In that case

- a. The where clause is applied first to the individual rows in the tables or table structures objects in the diagram pane. Only the rows that meet the conditions in the where clause are grouped.
- b. The having clause is then applied to the rows in the result set that are produced by grouping. Only the groups that meet the having conditions appear in the query output.

eg.

```
select dept_no from EMPLOYEE
group by dept_no
having avg (salary) >=all
(select avg (salary)
from EMPLOYEE
group by dept_no)
```

3. *Aggregate functions*: Aggregate functions such as SUM, AVG, count, count (*), MAX and MIN generate summary values in query result sets. An aggregate functions (with the exception of count (*)) processes all the selected values in a single column to produce a single result value:

eg.

```
select dept_no, count (*)
from EMPLOYEE
group by dept_no
```

eg.

```
select max(salary) as maxi1171117
from EMPLOYEE
```

eg.

```
select sum(salary) as total salaiy
from EMPLOYEE
```

eg.

```
select min(salary) as mninsal
from EMPLOYEE
```

4. *Union and Union Operators*: Combines the result of two or more queries into a single result set consisting of all the rows belonging to all queries in the union. This is different from using joins that combine columns from two tables. Two basic rules for combining the result sets of two queries with union are:

- A. The number and the order of the columns must be identical in all queries.
- B. The data types must be compatible:

```
select max(salary) as maximum
from EMPLOYEE
11171017
select min(salary)
from EMPLOYEE
union
```

Specifies that multiple result two or more queries into a single result set consisting of all the rows belonging to all queries into single result set consisting of all the rows belonging to all queries in the union. This is different from using joins that combine columns from two tables. Two basic rules are followed.

5. *Order by clause*: SQL allows the user to order the tuples in the result set of the query of a query by the values of one or more attributes using the order by clause. The default order is in the increasing order of values. We call specify the keyword DES if we want values in descending order.
6. *Exists and not exists*: Subqueries introduced with exists and not queries can be used for two set theory operations: Intersection and Difference. The intersection of two sets contains all elements that belong to both of the original sets. The difference contains elements that belong to only first of the two sets.

eg.

```
select 'from DEPARTMENT
where exists (select * from PROJECT
where DEPARTMENT.deptno=PROJECTdept_no)
```

7. *IN and NOT IN*: SQL allows testing tuples for membership in a relation. The "IN" connective tests for set membership where the set is a collection of values produced by select clause. The "NOT IN" connective tests for the absence of set membership. The IN and NOT IN connectives can also be used on enumerated sets.

eg.

```
select prof_name from PROJECT
where dept no not in (select dept no from DEPARTMENT
where dept_name='chemistry')
```

eg.

```
select fname from EMPLOYEE
where SSN in (select mgr_SSN from DEPARTMENT)
```

Conclusion: Thus, we have studied and executed all the queries mentioned using various clauses.

Experiment-6

Aim: Advanced SQL Queries -2.

- (1) Join (Inner & Outer)
- (2) Exists & Union

Resources used: MS-Access, Visual FoxPro, SQL-Server/etc

Theory:

JOINS: SQL joins are used to query data from two or more tables, based on a relationship between certain columns in these tables.

Type of JOIN:

Equi Joins:

This operation allows to connect, with a relation of equality, the tables which have at least a common attribute. One must have $n-1$ conditions of join, n being the number of tables which intervene in the query.

If no condition of join is specified, the corresponding query will realize the Cartesian product of the implied tables.

Syntax:

```
SELECT TABLE1.col1, TABLE1.col2... TABLE2.col1, TABLE2.col2...
FROM table_name1, table_name2
```

```
WHERE table_name1.col1 = table_name2.col2
```

TYPE OF Equi-Joins:

An equi-join is further classified into two categories:

- (a) *Inner Join*
- (b) *Outer Join*

(a) *Inner Join:*

The INNER JOIN keyword return rows when there is at least one match in both tables.

Syntax:

```
SELECT column _name(s)
FROM table name 1
INNER JOIN table name2
ON tablename1 .column n name=tablename2.column name
```

(b) *Outer Joins:*

The outer join is returning all the rows returned by simple join or equijoin as well as those rows from one table that do not match any row from the other table, the symbol (+) represents outer join. the outer table operator can appear only on side of the expression.

Type of Outer Joins:

- **Left OUTER JOIN:** Return all rows from the left table, even if there are no matches in the right table.

Syntax:

```
SELECT TABLE1.column
TABLE1.column
FROM table _name 1 , table _name2
WHERE table _name1.column(+) = table _name2.column;
```

- **Right OUTER JOIN:** Return all rows from the right table, even if there are no matches in the left table.

Syntax:

```
SELECT TABLE1.column
TABLE2.column
FROM table _name1,table _name2
WHERE table _name1.column = table _name2.column(+);
```

EXISTS

EXISTS uses a subquery as a condition, where the condition is True if the subquery returns any rows, and False if the subquery does not return any rows.

Syntax:

```
SELECT columns
FROM tables
WHERE EXISTS (subquery);
```

UNION

There are occasions where you might want to see the results of multiple queries together, combining their output; use UNION.

The SQL UNION operator combines two or more SELECT statements.

Syntax:

```
SELECT column name(s) FROM table _name 1
UNION
SELECT column name(s) FROM table name2
```

Notice that SQL requires that the Select list (of columns) must match, column-by-column, in data type This concept is useful in situations where a primary key is related to a foreign key, but the foreign key value for some primary keys is NULL. For example, in one table, the primary key is a salesperson, and in another

table is customers, with their salesperson listed in the same row. However, if a salesperson has no customers, that person's name won't appear in the customer table.

Conclusion: Thus, we have studied and executed all the queries mentioned using various clauses.

Experiment-7

. Aim: Implementation of views.

1. *Creation of views*
2. *Usage of views*
3. *Creation of views using views*
4. *Drop view*

Resources used: MS-Access/Visual-FoxPro/SQL-Server/etc

Theory:

Views: Any relation that is not part of any logical model but is made visible to the user as a virtual relation is called as a view. It is possible to support a large number of views on the top of any given set of actual database relation. Views help in 2 ways:

- I. For security purpose
2. Create a personalized collection of relation that is better user's intuition than is logical model

Creation of Views:

1. Views is defined using 'create view' command
2. To define a view we must give the view a better name and must state the query that computes the view.
3. Syntax:
create view <view name> as <query expression>
Where query expression is any legal query expression.
4. Once we have defined a view, we can use the view name to refer to the virtual relation that the view generation.
5. Attribute name of the view can be specified explicitly as:
Create view V(VA1, VA2,... VA,,) as selectA,,) from RI where(p)
where, p: predicate
RI: relation
At-A_n: attribute of view
V: view name

Creation of views using VIEW:

Since, view relations may appear in any place that a relation name may appear, except for restrictions on the use of views in update expressions. Thus, one view may be used in the expression defining another view. For eg. Let Emp_work_info is a view with attribute F_name, SSN, Project_no, Work_hrs. Then creation of other view can be done as:

```
create view new_view
select f11c7177e, work_hrs
from emp_work_info
```

Updating of views

Although views are useful for the queries, they present a serious problem. If we express updates insertion or deletion on view as the modification done to the database in terms of the views must be translated to a modification to actual relations in the logical methods of database,

Drop view A view creates earlier can be dropped using 'Drop View' command Syntax:

```
Drop view 'r'
where, r: View Name.
```

It deletes all the information about view from the database.

PART C: LEARNING RESOURCES	
Textbooks, Reference Books, Other Resources	
Suggested Readings:	
1. Dr Rajeev Chopra, "Database Management System (DBMS) A Practical Approach", 2010, S Chand 2. Jitendra Patel, "DBMS Lab Manual" Kindle Edition, 2012 3. Books published by M.P. Hindi Granth Academy, Bhopal	
Suggestive digital platform web links	
http://www.mphindigranthacademy.org/	
Suggested equivalent online courses	
Nil	

Part D-Assessment and Evaluation

Suggested Continuous Evaluation Methods:

PART D: Assessment and Evaluation			
Internal Assessment : Continuous Comprehensive Evaluation (CCE) : 40 Marks		External Assessment: University Exam (UE) : 60 Marks Time 03.00 Hours	
Internal Assessment	Marks	External Assessment	Marks
Hands-on Lab Practice	10 Marks	Practical record file	10 Marks
Viva	10 Marks	Viva voce practical	10 Marks
Lab Test from practical list	10 Marks	Table works/ Exercise Assigned (02) in practical exam	30 Marks
Assignments (Charts/ Model)/ Technology Dissemination/ Excursion/ Lab visit/ Industrial Training	10 Marks	Reports of excursion/ Lab visits/ Industrial training/ Survey/ Collection/ Models	10 Marks
Total	40 Marks	Total	60 Marks
Excursion/ Lab visits/ Industrial Training is compulsory			

Any remarks/suggestions: Students should also prepare a small Audio-video clip to present the details of

- Assignments submitted
- Imparting training of common online citizen services or software tools

PART A: Introduction			
Program: Diploma	Class: BCA	Year: II Year	Session:
1.	Course Code	UINTECA406	
2.	Course Title	Internet of Things (IOTs) Lab	
3.	Course Type (Core Course/ Elective/ Generic Elective/ Vocational)	Elective	
4.	Pre-Requisite (if any)	Open for all	
5.	Course Learning Outcomes (CLO)	After completing this lab course, students will be able to: 1. Arduino/Raspberry Concept. 2. Knowledge of Digital Sensor. 3. Uses of DHT11 Sensors. 4. Knowledge of Bluetooth interface.	
6.	Credit Value	Practical — 2 Credits	
7.	Total Marks	Max. Marks: 40+60	Min. Passing Marks: 16+24

PART B: Content of the Course		
No. of Lab. Practicals (in hours per week): 1 Hr. per week		
Total No. of Labs: 30 Hrs.		
	Suggestive List of Practicals	No. of Labs.
	1. To interface LED/Buzzer with Arduino/Raspberry Pi and write a program to turn ON LED for 1 sec after every 2 seconds. 2. To interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection. 3. To interface DHT11 sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings. 4. To interface motor using relay with Arduino/Raspberry Pi and write a program to turn ON motor when push button is pressed. 5. To interface OLED with Arduino/Raspberry Pi and write a program to print temperature and humidity readings on it. 6. To interface Bluetooth with Arduino/Raspberry Pi and write a program to send sensor data to smartphone using Bluetooth. 7. To interface Bluetooth with Arduino/Raspberry Pi and write a program to turn LED ON/OFF when '17'0' is received from smartphone using Bluetooth. 8. Write a program on Arduino/Raspberry Pi to upload temperature and humidity data to things peak cloud. 9. Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from things peak cloud. 10. To install MySQL database on Raspberry Pi and perform basic SQL queries.	30 Hrs.

PART D: Assessment and Evaluation			
Internal Assessment : Continuous Comprehensive Evaluation (CCE) : 40 Marks		External Assessment: University Exam (UE) : 60 Marks Time 03.00 Hours	
Internal Assessment	Marks	External Assessment	Marks
Hands-on Lab Practice	10 Marks	Practical record file	10 Marks
Viva	10 Marks	Viva voce practical	10 Marks
Lab Test from practical list	10 Marks	Table works/ Exercise Assigned (02) in practical exam	30 Marks
Assignments (Charts/ Model)/ Technology Dissemination/ Excursion/ Lab visit/ Industrial Training	10 Marks	Reports of excursion/ Lab visits/ Industrial training/ Survey/ Collection/ Models	10 Marks

Total	40 Marks	Total	60 Marks
<i>Excursion/ Lab visits/ Industrial</i>			
<i>Training is compulsory</i>			

