

MCA 3rd Semester

Course: 301: Client Server Architecture

Course Code	301
Course Title	Client Server Architecture
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2016
Purpose of Course	This course is an introduction to students to understand fundamentals of DBMS. The course also gives students an idea about various concept of Database and its working.
Course Objective	1. To acquaint the students with Client Server Architecture and PL/SQL programming
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand how dbmsworks and the importance of various concepts of dbms. This coursewill also help students to appreciate the role of various dbms tools. Aftersuccessful completion, students will be able to use various tools of dbmsnecessary for the application.
Course Content	<p>Unit 1: Database Computing Model</p> <p>1.1 Client Server Computing: Functions of client, server, middleware components</p> <p>1.2 Advantages and limitations of client server computing</p> <p>1.3 Three Tier Architecture: Overview of thin client, application server, web server</p> <p>1.4 Overview of Distributed Database</p> <p>1.5 Overview of Real Application Clusters</p> <p>1.6 Overview of High Performance Database Computing</p> <p>1.7 Overview of Data Warehousing and Data Mining</p> <p>Unit 2: Overview of Oracle Database Server Architecture</p> <p>2.1 Architecture of Oracle Database and Oracle Instance</p> <p>2.2 Overview of Physical and Logical Structures</p> <p>2.3 Dedicated and Shared Server Configuration</p> <p>2.4 Oracle Server Startup and Shutdown</p> <p>2.5 Creating Database</p> <p>Unit 3: Oracle Tools and Utilities</p> <p>3.1 SQL</p> <p>3.2 PL/SQL Procedural Extension.</p> <p> 3.2.1 Overview, PL/SQL data types & Control Structures</p> <p> 3.2.2 Cursors</p> <p> 3.2.3 Stored Procedures & Functions</p> <p> 3.2.4 Database Triggers</p> <p> 3.2.5 Package Creation</p> <p> 3.2.6 Dynamic SQL</p> <p> 3.2.7 Collections & Objects</p> <p>Unit 4. Database Administration</p> <p>4.1 Managing Users</p> <p> 4.1.1 User Authentication Methods</p> <p> 4.1.1.1 Password Authentication</p>

	<p>4.1.1.2 O.S Authentication</p> <p>4.1.2 User Configuration Setup</p> <p>4.1.2.1 Profiles</p> <p>4.1.2.2 Default Table space</p> <p>4.1.2.3 Temporary Table space</p> <p>4.1.3 Resource Management</p> <p>4.1.3.1 Quotas</p> <p>4.1.4 Working with user database account</p> <p>4.1.4.1 Creating, Modifying and deleting user account</p> <p>4.1.4.2 Changing password</p> <p>4.2 Backup & Recovery</p> <p>4.3 Database Security</p> <p>4.3.1 Authentication</p> <p>4.3.2 Privileged Accounts & Privileges</p> <p>4.3.3 Object Security</p> <p>4.3.4 System security</p> <p>4.3.5 Database Roles</p> <p>4.3.6 Database Auditing</p> <p>[Self Study]</p> <p>Export & Import Tools, Overview of Grid Based Database</p> <p>**Computing, Calling External Routines from PL/SQL</p>
Reference Books	<ol style="list-style-type: none"> 1. Oracle 9i PL/SQL Programming -Scott Urman- Oracle Press 2. Oracle DBA Fundamentals-I - Oracle Press 3. Effective PL/SQL:- Apress 4. Expert Oracle Database Architecture 9i and 10g- Tom Kyte- Apress 5. Effective Oracle By Design - Peter Norton - Tom Kyte- Oracle Press 6. Oracle 24 X 7 Tips and Techniques - VenkatDevraj– Oracle Press 7. Expert Oracle Database 11g Administration – Alpati- Wiley Student Edition 8. Fundamentals of Database Management System- Gilleneon- Wiley Student Edition 9. SQL & PL/SQL for Oracle 11g Black Book - Deshpande- McGraw Hill 10. Beginning Oracle Database 11g Administration from novice to professional-Iggy Fernandez - Apress/Springer 11. Oracle PL/SQL-Benjamin Rosenweig& Elena Silvestrova- 4/e,Pearson 12. Database Systems Using Oracle: A simplified guide to SQL & PL/SQL- Shah Nilesh- PHI 13. Learning Oracle SQL & PL/SQL: A Simplified Guide- Chatterjee, Rajeeb C- PHI
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 302: Operating System

Course Code	302
Course Title	Operating System
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2016
Purpose of Course	This course is an introduction to students to understand fundamentals of Operating system. The course also explains the concept of various operating systems, their use, architecture and its working.
Course Objective	<ol style="list-style-type: none"> 1. To make students understand operating system fundamentals 2. To make students understand various parts of operating system 3. To make students understand the importance and use of operating system 4. To explain various types of operating system architecture
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand how operating system works and the importance of various parts of operating system. This course will also help students to appreciate the role of various operating systems. After successful completion, students will be able to select particular configuration of computer and operating system necessary for the application.
Course Content	<p>Unit 1: Operating System Concepts</p> <p>1.1 History of Operating Systems 1.2 Operating System Structure: Layered System, Microkernel and Virtual Machine 1.3 System Calls</p> <p>Unit 2: Process & Multi-processing</p> <p>2.1 Process Concept 2.2 Process State Transition Model 2.3 Process Scheduling 2.3.1 Scheduling Criteria 2.3.2 Scheduling Algorithms 2.3.3 Context Switching 2.4 Thread & Multithreading</p> <p>Unit 3: Process Coordination</p> <p>3.1 Inter-process Communication 3.2 Process Synchronization 3.2.1 The Critical-Section Problem 3.2.2 Peterson's Algorithm 3.2.3 Semaphore 3.2.4 Classic Problems of Synchronization 3.3 Deadlock 3.3.1 Deadlock Characteristics 3.3.2 Deadlock Avoidance 3.3.3 Deadlock Prevention 3.3.4 Deadlock Detection & Recovery</p> <p>Unit 4: Memory & File Management</p> <p>4.1 The notion of Physical and Logical Address Space 4.2 Binding and Binding Times</p>

	<p>4.3 Contiguous Allocation & Non-Contiguous Allocation</p> <p> 4.3.1 Paging</p> <p> 4.3.2 Memory Mapping with Paging</p> <p>4.4 Structure of Page Table</p> <p>4.5 Segmentation</p> <p>4.6 Other Memory Management Schemes: Swapping and Overlays</p> <p>4.7 Demand Paging & Demand Segmentation</p> <p> 4.7.1 Allocation of Frames</p> <p> 4.7.2 Page Replacement policies</p> <p> 4.7.3 Thrashing and other issues</p> <p>4.8 File Concept: File Types and File Operation</p> <p>4.9 Directory Structure</p> <p>4.10 Directory Implementation</p> <p>4.11 File-System Implementation</p> <p>4.12 Allocation Methods</p> <p>4.13 Free-Space Management</p> <p>4.14 File-System Mounting</p> <p>4.15 File Sharing and Protection</p> <p>4.16 Log-Structured File-System</p> <p>Unit 5: Device Management & Security</p> <p>5.1 I/O Hardware</p> <p>5.2 Application I/O Interface</p> <p>5.3 Kernel I/O Subsystem</p> <p>5.4 Mass Storage Structure</p> <p> 5.4.1 Disk Structure</p> <p> 5.4.2 Disk scheduling</p> <p> 5.4.3 Disk Management</p> <p>5.5 Protection</p> <p> 5.5.1 Goals of Protection</p> <p> 5.5.2 Domain of Protection</p> <p>5.6 Security Problem</p> <p> 5.6.1 User Authentication</p> <p> 5.6.2 Cryptography as Security Tool</p> <p> 5.6.3 Program Threats</p> <p> 5.6.4 System and Network threats</p> <p> 5.6.5 Implementing Security Defences</p> <p>5.7 Introduction to Network and Distributed O.S</p> <p>[Self Study]</p> <p>Introduction to FreeBSD and Windows Server 2008</p>
Reference Books	<ol style="list-style-type: none"> 1. Operating System Principles by Abraham Silberschatz - Peter Baer Galvin, and Greg Gagne Wiley India 2. Operating Systems: A Concept-based Approach – Dhamdhare-TMH 3. Modern Operating Systems - Andrew S. Tanenbaum- Pearson Edu./PHI 4. Operating Systems-AchyutGodbole – TMH 5. Distributed Operating Systems - Tanenbaum, Pearson 6. Unix Concepts and Application - Das-McGraw-Hill
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 303-Software Engineering

Course Code	303
Course Title	Software Engineering
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2016
Purpose of Course	This course is an introduction to students to understand life cycle of Software engineering. The course also gives students an idea about various process models of software engineering
Course Objective	<ol style="list-style-type: none"> 1. To make students understand software engineering 2. To make students understand various components of software engineering 3. To make students understand the importance of software models
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand the life cycle of Software engineering. This course will also help students to appreciate the Software engineering life cycle. After successful completion, students will be able to select particular model of software engineering
Course Content	<p>Unit 1: Introduction</p> <ol style="list-style-type: none"> 1.1 Role of Software 1.2 What is software? 1.3 Software characteristics 1.4 Software Engineering: definition 1.5 Process models <ol style="list-style-type: none"> 1.5.1 Waterfall Model 1.5.2 Prototyping 1.5.3 Incremental models 1.5.4 Spiral model 1.6 Agile Processes <ol style="list-style-type: none"> 1.6.1 Agility Principles 1.6.2 Agile Modelling 1.7 Agile Process Models <ol style="list-style-type: none"> 1.7.1 Extreme Programming Model 1.7.2 Adaptive Software Development Model 1.7.3 Scrum Model 1.7.4 Feature Driven Development Model 1.7.5 Lean Software Development Model <p>Unit 2: Requirement Engineering and Requirement analysis modelling, Design</p> <ol style="list-style-type: none"> 2.1 Requirement engineering tasks 2.2 Introduction to requirement Inception, Elicitation, Elaboration, Negotiation, Specification, Validation, Management 2.3 Requirement analysis 2.4 Elements of analysis model and modelling approaches 2.5 Data Flow oriented modelling 2.6 Software requirement specification 2.7 Translation of analysis model into the design model 2.8 Design process and design quality 2.9 Design concepts 2.10 Design model

- 2.11 Concept of Pattern based software design
- 2.12 Introduction: Architecture design
- 2.13 Mapping data flow into software architecture
- 2.14 User interface design: golden rules

Unit 3: Software Project management, metrics, Estimation & Scheduling

- 3.1 People
 - 3.1.1 Stake holders
 - 3.1.2 Team leaders
 - 3.1.3 Software team
 - 3.1.4 Agile teams
- 3.2 Product
 - 3.2.1 Software scope
 - 3.2.2 Problem decomposition
- 3.3 Process
 - 3.3.1 Process decomposition
- 3.4 Project
 - 3.4.1 Issues, approaches
- 3.5 W5HH principle
- 3.6 Measures, metrics and indicators
- 3.7 Process metrics and process improvement
- 3.8 Project metrics
- 3.9 Size oriented metrics
- 3.10 Function oriented metrics
- 3.11 Metrics for software quality
- 3.12 Integrating metrics within the software process
- 3.13 Software project estimation
- 3.13 Decomposition techniques
- 3.14 Problem based estimation
- 3.15 LOC based estimation
- 3.16 FP based estimation
- 3.17 Process based estimation
- 3.18 Empirical estimation models
- 3.19 Basic concepts
- 3.20 Basic principles
- 3.21 Relationship between people and effort
- 3.22 Effort distribution
- 3.23 Task network
- 3.24 Scheduling and tracking
- 3.25 Earned value analysis

Unit 4: Software Testing & Quality Management

- 4.1 Testing fundamentals
 - 4.1.1 Principles
 - 4.1.2 Test characteristics
- 4.2 White box testing
 - 4.2.1 Basis path testing
 - 4.2.2 Control structure testing
- 4.3 Black box testing
 - 4.3.1 Equivalence partitioning
 - 4.3.2 Boundary value analysis
- 4.4 Software Testing Strategies
 - 4.4.1 Unit testing
 - 4.4.2 Integration testing
 - 4.4.3 Validation testing
 - 4.4.4 System testing

	<p>4.5 Quality concepts 4.6 Software quality assurance 4.7 SQA activities 4.8 Formal technical reviews 4.9 Statistical software quality assurance 4.10 Software reliability</p> <p>Unit 5: Risk Management 5.1 Strategies 5.2 Software risks 5.3 Risk identification, assessment, components, and drivers 5.4 Risk projection 5.5 Risk mitigation, monitoring and management</p> <p>[Self Study] Test Case Generation and Testing using Testing Tool SRS and case studies should be carried out.</p>
Reference Books	<ol style="list-style-type: none"> 1. Software Engineering – A Practitioner’s approach - R.S. Pressman 2. Software engineering principles and practice- Javadekar- McGraw Hill 3. Software engineering -Ian Somerville- Tata McGraw Hill 4. Fundamentals of software engineering,3rd edition-Rajib Mall - Addition Wesley / pearson education PHI 5. An integrated approach to software engineering,3rd Edition- Pankaj Jalote-Narosa 6. Software engineering with java- Stephan R. Schach –TMH
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 304: Programming in .NET

Course Code	304
Course Title	Programming in .NET
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2016
Purpose of Course	This course is an introduction to students to understand fundamentals of .NET technology. The course also gives students an idea about VB.NET Programming. The course also explains the concept of ASP.NET
Course Objective	<ol style="list-style-type: none"> 1. To make students understand .NET Technology 2. To make students understand VB.NET Programming 3. To make students understand the importance of ASP.NET
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand how .NET Technology works and the importance of object oriented programming. This course will also help students to appreciate the VB.NET programming.
Course Content	<p>Unit1: Overview of Microsoft .NET Framework</p> <ol style="list-style-type: none"> 1.1 The .NET Framework 1.2 The Common Language Runtime (CLR) 1.3 The .NET Framework class Library 1.4 .NET Web Services <p>Unit2: Visual Basic .NET programming</p> <ol style="list-style-type: none"> 2.1 Working with Tool box controls 2.2 Working with Menus and Dialog Boxes 2.3 Tapping Errors Using Structured Error Handling 2.4 Using Modules and Procedures 2.5 Using Arrays and Collections <p>Unit3: Object Oriented Programming</p> <ol style="list-style-type: none"> 3.1 Creating Classes, Object Construction & Destruction 3.2 Class Libraries : DLL's & Static Classes 3.3 Abstraction, Encapsulation & Polymorphism 3.4 Interfaces & Inheritance 3.5 Object Serialization <p>Unit4: Multithreading, Files & Streams</p> <p>Unit5: Database access using ADO.NET & ASP.NET Introduction</p> <ol style="list-style-type: none"> 5.1 Visual Database Tools 5.2 ADO.NET Object Model 5.3 ADO.NET Programming 5.4 Working with DataSets 5.5 Integration with XML 5.6 The Middle Tier <p>[Self Study] Creating user controls in VB.Net</p>
Reference Books	<ol style="list-style-type: none"> 1. Moving to VB.NET: Strategies, Concepts, and Code - Dan Appleman 2. Microsoft Visual Basic .NET Step By Step- Michael Halvorson –

	<p>PHI</p> <p>3. Database Programming with Visual Basic .NET and ADO.NET- F. Scott Barker - Sams Publishing</p> <p>4. Beginning .NET Web Services Using Visual Basic .NET- Joe Bustos and Karlli Watson – Wrox Publication</p> <p>5. .NET – Complete Development Cycle- G. Lenz, T. Moeller – Pearson Education</p> <p>6. Professional VB.NET, 2nd Edition- Fred Barwell – et al - Wrox Publication</p> <p>7. Mastering Visual Basic .NET Database Programming – Bilgin– BPB Publication</p> <p>8. Visual Basic 2008 Programming Black Book Beginners Ed.- Kogent - Wiley</p> <p>9. Pro VB 2008 and the .Net 3.5 Platform, Microsoft Visual Basic 2008 Step By Step - Michael Halvorson- Microsoft Press</p>
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 305: Java Programming

Course Code	305
Course Title	Java Programming
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2016
Purpose of Course	This course is an introduction to students to understand java programming. The course also gives students an idea about object oriented programming methodology.
Course Objective	<ol style="list-style-type: none"> 1. To make students understand java programming. 2. To make students understand object oriented programming methodology. 3. To make students understand the importance of desktop and web application
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand how to program java and the importance of various object oriented programming methodology.
Course Content	<p>Unit1: Fundamentals of Java</p> <ol style="list-style-type: none"> 1.1 Features of Java 1.2 Java Developer's Kit: Java Compiler, Java Interpreter, Java Debugger, Applet Viewer 1.3 Working with IDE 1.4 JVM & Bytecode 1.4 Primitive and Non-Primitive Data types 1.5 Variable and Reference variable 1.6 Operators & Expressions 1.7 Flow control statements 1.8 Working with Arrays & String <p>Unit2: Object Oriented Programming</p> <ol style="list-style-type: none"> 2.1 Classes and Objects <ol style="list-style-type: none"> 2.1.1 Fields & Methods 2.1.2 Object Creation, Construction and Initialization 2.1.3 Method and Constructor Overloading 2.2 Inheritance <ol style="list-style-type: none"> 2.2.1 Extending Class 2.2.2 Role of Constructors in inheritance 2.2.3 Inheriting and redefining members 2.2.4 Type computability and conversion 2.2.5 'This' & 'Super' Keyword 2.3 Interfaces <ol style="list-style-type: none"> 2.3.1 Abstract class & Interfaces 2.3.2 Defining & Extending Interfaces 2.3.3 Implementing Interfaces 2.4 Packages <ol style="list-style-type: none"> 2.4.1 Package Naming 2.4.2 Package Access 2.4.3 Packages and CLASSPATH 2.4.4 Package Example 2.5 Overview of Streams 2.6 Bytes vs. Characters Streams 2.7 Converting Byte Streams to Character Streams

	<p>2.8 Basic and Filtered Streams 2.9 File Object 2.10 Reading and Writing to Files 2.11 Object Serialization 2.12 Garbage Collection 2.13 Finalization 2.14 Interacting with Garbage Collector 2.15 Reachability States and Reference Objects</p> <p>Unit3: Exception Handling & Threads 3.1 Exception & Types of Exception 3.2 Use of try, catch, finally, throw, throws in Exception Handling 3.3 Control Flow in Exceptions 3.4 Creating User Defined Exceptions 3.5 Creating Threads 3.6 Extending Thread class 3.7 Implementing Runnable interface 3.8 Thread scheduling 3.9 Thread Synchronizing 3.10 Use of 'synchronized' keyword 3.11 wait, notify and notifyAll methods 3.12 'volatile' modifier</p> <p>4. Applets & GUI Programming 4.1 Abstract Window Toolkit (AWT) Class 4.1.1 Components and Containers 4.1.2 Layout Managers 4.1.3 Classes for various controls: Labels, Buttons, Text Fields, Text Areas, Check Boxes and Choice List, etc... 4.2 Event Handling 4.2.1 Event Types & Classes 4.2.2 The Delegation Model of Event Handling: Event Classes, Event Sources & Event Listeners 4.2.3 Adapter Classes 4.3 Introduction to Applets 4.4 Applet Life Cycle Methods</p> <p>Unit 5: Networking & JDBC 5.1 URL Manipulation 5.2 InetAddress Class 5.3 Socket Overview 5.3.1 TCP and UDP Socket 5.3.2 Introduction of Java Classes: ServerSocket, Socket, DatagramSocket & DatagramPacket 5.3.3 Client Server Interaction with Stream Socket Connection 5.3.4 Datagrams: connectionless Client/Server Interaction 5.4 JDBC Drivers 5.5 Making a Connection with Database 5.6 Basis for Processing SQL Statements 5.7 Creating Statements: PreparedStatement 5.8 Executing Queries 5.9 Processing ResultSet Objects</p> <p>[Self Study] EJB – Overview</p>
Reference Books	1. Java Programming Language– James Gosling, David Holmes Arnold - Addition Wesley

	<p>2. Java – The Complete Reference- Patrick Naughton - Tata McGraw Hill</p> <p>3. Experiments with Java: An Introductory Lab Manuals -S.A. Robelsky - Addison Wesley</p> <p>4. Java 2 From Scratch-Stevens Halmes – PHI / pearson education PHI</p> <p>5. Java: how to program, 9th Edition - Paul Deitel,HarveyDeitel-Tata McGraw Hill</p> <p>6. Complete Reference- Peter Naughten</p>
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Veer Narmad South Gujarat University, Surat

M.C.A (3rd Semester)

Syllabus Effective from: June 2016

Paper: 306-Programming Skills V

Practical based on paper no 301.

Separate journals to be prepared for this subject 306.

Veer Narmad South Gujarat University, Surat

M.C.A (3rd Semester)

Syllabus Effective from: June 2016

Paper: 307- Programming Skills VI

Practical based on paper no 304.

Separate journals to be prepared for this subject 307.

Veer Narmad South Gujarat University, Surat

M.C.A (3rd Semester)

Syllabus Effective from: June 2016

Paper: 308- Programming Skills VII

Practical based on paper no 305.

Separate journals to be prepared for this subject 308.