JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	L	T	P	C
	4	0	0	4

(17F00301) DATABASE MANAGEMENT SYSTEMS

Objectives:

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To become familiar with the basic issues of transaction processing and concurrency control.
- To become familiar with database storage structures and access techniques.

Outcomes:

- Demonstrate the basic elements of a relational database management system,
- Ability to identify the data models for relevant problems.
- Ability to design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data.
- Apply normalization for the development of application software.

UNIT-I

Introduction-Database System Applications, Purpose of Database Systems, View of Data - Data Abstraction, Instances and Schemas, Data Models, Database Languages - DDL, DML, Database Architecture, Database Users and Administrators, History of Data base Systems.

Introduction to Data base design , ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises. Relational Model: Introduction to the Relational Model - Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views Destroying/ altering Tables and Views.

UNIT-II

Relational Algebra and Calculus: Relational Algebra - Selection and Projection, Set operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus - Tuple relational Calculus - Domain relational calculus - Expressive Power of Algebra and calculus.

Form of Basic SQL Query - Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set - Comparison Operators, Aggregate Operators, NULL values - Comparison using Null values - Logical connectives - AND, OR and NOT - Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers and Active Data bases.

UNIT-III

Introduction to Schema Refinement - Problems Caused by redundancy, Decompositions - Problem related to decomposition, Functional Dependencies - Reasoning about FDS, Normal Forms - FIRST, SECOND, THIRD Normal forms - BCNF - Properties of Decompositions - Loss less join Decomposition, Dependency preserving Decomposition, Schema Refinement in Data base Design - Multi valued Dependencies - FOURTH Normal Form, Join Dependencies, FIFTH Normal form, Inclusion Dependencies.

UNIT-IV

Transaction Management - Transaction Concept - Transaction State - Implementation of Atomicity and Durability - Concurrent - Executions - Serializability - Recoverability - Implementation of Isolation - Testing for serializability.

Concurrency Control - Lock - Based Protocols - Timestamp Based Protocols - Validation - Based Protocols - Multiple Granularity.

Recovery System-Failure Classification-Storage Structure-Recovery and Atomicity - Log - Based Recovery - Recovery with Concurrent Transactions - Buffer Management - Failure with loss of nonvolatile storage - Advance Recovery systems - Remote Backup systems.

UNIT-V

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing - Clustered Indexes, Primary and Secondary Indexes, Index data Structures - Hash Based Indexing, Tree based Indexing, Comparison of File Organizations.

Tree Structured Indexing: Intuitions for tree indexes, Indexed Sequential Access Methods(ISAM) B+ Trees: A Dynamic Index Structure, Search, Insert, Delete.

Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendible vs. Linear Hashing.

TEXT BOOKS:

1. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, McGrawHill Education, 3rd Edition, 2003.
2. Data base System Concepts, A.Silberschatz, H.F. Korth, S.Sudarshan, McGraw Hill, VI edition, 2006.

REFERENCES:

1. Database Systems, 6th edition, Ramez Elmasri, Shamkat B. Navathe, Pearson Education, 2013.
2. Database Systems Concepts, Peter Rob & Carlos Coronel, Cengage Learning, 2008.
3. Introduction to Database Systems, C.J. Date, Pearson Education.
4. Database Management Systems, G.K. Gupta, McGrawHill Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	**L**	**T**	**P**	**C**
	4	**0**	**0**	**4**

(17F00302) COMPUTER NETWORKS

Course Objectives:

- Study the evolution of computer networks and future directions.
- Study the concepts of computer networks from layered perspective.
- Study the issues open for research in computer networks.

Course Outcomes:

- Ability to choose the transmission media depending on the requirements.
- Ability to design new protocols for computer network.
- Ability to configure a computer network logically.

Unit I

Introduction: Networks, Network Types, Internet History, Standards and Administration, Network Models: Protocol Layering, TCP/IP Protocol Suite, The ISO Model.

The Physical layer: Data and Signals, Transmission impairment, Data rate limits, Performance, Transmission media: Introduction, Guided Media, Unguided Media, Switching: Introduction, Circuit Switched Networks, Packet switching.

Unit II

The Data Link Layer: Introduction, Link layer addressing, Error detection and Correction: Cyclic codes, Checksum, Forward error correction, Data link control: DLC Services, Data link layer protocols, HDLC, Point to Point Protocol, Media Access control: Random Access, Controlled Access, Channelization, Connecting devices and virtual LANs: Connecting Devices.

Unit III

The Network Layer: Network layer design issues, Routing algorithms, Congestion control algorithms, Quality of service, Internetworking, The network layer in the Internet: IPV4 Addresses, IPV6, Internet Control protocol, OSPF, BGP, IP, ICMPv4, IGMP.

Unit IV

The Transport Layer: The Transport Service, Elements of Transport Protocols, Congestion Control, The internet transport protocols: UDP, TCP, Performance problems in computer networks, Network performance measurement.

Unit V

The Application Layer: Introduction, Client Server Programming, WWW and HTTP, FTP, e-mail, TELNET, Secure Shell, Domain Name System, SNMP.

Text Books:

1. "Data communications and networking", Behrouz A. Forouzan, Mc Graw Hill Education, 5th edition, 2012.
2. "Computer Networks", Andrew S. Tanenbaum, Wetherall, Pearson, 5th edition, 2010.

References:

1. Data Communication and Networks, Bhushan Trivedi, Oxford
2. "Internetworking with TCP/IP – Principles, protocols, and architecture- Volume 1, Douglas E. Comer, 5th edition, PHI
3. "Computer Networks", 5E, Peterson, Davie, Elsevier.
4. "Introduction to Computer Networks and Cyber Security", Chawan- Hwa Wu, Irwin, CRC Publications.

"Computer Networks and Internets with Internet Applications", Comer.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	**L**	**T**	**P**	**C**
	4	**0**	**0**	**4**

(17F00303) WEB TECHNOLOGIES

Course Objectives:

- Learn the fundamentals of HTML and JavaScript
- Learn to communicate over a network using java
- Learn do design server side programs and access them from client side

Course Outcomes:

- Ability to design websites and do client side validations
- Share information over a network
- Ability to write server side programs

Unit I

Fundamentals: Introduction to the Web, Web servers and Clients, Resources, URL and its Anatomy, Message Format, Persistent and Non-persistent connections, Web Caching, Proxy, Java and the Net, Java Network Classes and Interfaces, Looking up Internet Address, Client/Server programs, Socket programming, e-mail client, POP3 programs, Remote method invocation, Example.

Unit II

HTML: HTML and its Flavors, HTML basics, Elements, Attributes and Tags, Basic Tags, Advanced Tags, Frames, Images, Meta tag, Planning of Web page, Model and Structure for a Website, Designing Web pages, Multimedia content.

Cascading style sheets: Advantages, Adding CSS, Browser compatibility, CSS and page layout, Selectors.

Unit III

JavaScript: Introduction, Variables, Literals, Operators, Control structure, Conditional statements, Arrays, Functions, Objects, Predefined objects, Object hierarchy, Accessing objects, Events, Event handlers, Multiple windows and Frames, Form object and Element, Advanced JavaScript and HTML, Data entry and Validation, Tables and Forms, DHTML with javascript.

Unit IV

Server side programming: Internet programming paradigm, Sever-side programming, Languages for CGI, Applications, Server environment, Environment variables, CGI building blocks,CGI scripting using C, Shell script, Writing CGI program, CGI security, Alternatives and Enhancement to CGI, Server-side Java, Advantages over Applets, Servlet alternatives, Servlet strengths, Servlet architecture, Servlet life cycle, Generic and HTTP Servelet, First servlet, Passing parameters to servlets, Retrieving parameters, Server-side include, Cookies, Fileters, Problems with servlet, Security issues, JSP and HTTP, JSP Engines, How JSP works, JSP and Servlet, Anatomy of a JSP page, JSP syntax, JSP components.

Unit – V

Sever side programming: continued: Beans, Session tracking, Users passing control and data between pages, Sharing session and Application data, Database connectivity, JDBC drivers, Basic steps, Loading a driver, Making a connection, Execute and SQL statement, SQL statements, Retrieving the result, Getting database information, Scrollable and updatable resultset, Result set metadata, Introduction to

JavaBeans, Bean builder, Advantages of Java Beans, BDK introspection, Properties, BeanInfo interface, Persistence, Customizer, JavaBeans API, EJB, Introduction to Structs Framework.

Text Books:

1. “Web Technologies”, Uttam K. Roy, , Oxford Higher Education., 1st edition, 10th impression, 2015

References

1. “Java How to program”, Paul deitel, Harvey deital, PHI
2. “Introduction to Java Programming”, Y.Daniel Liang, 6th Edition, Pearson Education, 2007
3. “The J2EE Tutorial”, Stephanie Bodoff et al, 2nd Edition, Pearson Education, 2004.
4. “Web Technologies”, Roy, Oxford University Press
5. “Web Technologies” Srinivasan, Pearson Education, 2012
6. “Java EE 5 for Beginners”, Ivan Bayross, Sharanam Shah, Cynthia Bayrossand Vaishali shai,SPD.
7. “Programming the Worldwide Web”, Robert W.Sebesta, 7th edition, 2009, Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	**L**	**T**	**P**	**C**
	4	**0**	**0**	**4**

(17F00304) SOFTWARE ENGINEERING

Course Objectives

- To understand the software life cycle models.
- To understand the software requirements and SRS document.
- To understand the importance of modeling and modeling languages.
- To design and develop correct and robust software products.
- To understand the quality control and how to ensure good quality software.
- To understand the planning and estimation of software projects.
- To understand the implementation issues, validation and verification procedures.
- To understand the maintenance of software

Course Outcomes

- Define and develop a software project from requirement gathering to implementation.
- Ability to code and test the software
- Ability to plan, Estimate and Maintain software systems

Unit I:

Software and Software Engineering: The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths

Process Models: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process.

Agile Development: Agility, Agility and the Cost of Change, Agile Process, Extreme Programming, Other Agile Process Models

Unit II:

Understanding Requirements: Requirements Engineering, Establishing the groundwork, Eliciting Requirements, Developing Use Cases, Building the requirements model, Negotiating Requirements, Validating Requirements.

Requirements Modeling (Scenarios, Information and Analysis Classes): Requirements Analysis, Scenario-Based Modeling, UML Models that Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling.

Requirements Modeling (Flow, Behavior, Patterns and WEBAPPS): Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements Modeling, Requirements Modeling for WebApps.

Unit III:

Design Concepts: Design with Context of Software Engineering, The Design Process, Design Concepts, The Design Model.

Architectural Design: Software Architecture, Architecture Genres, Architecture Styles, Architectural Design, Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow.

Component-Level Design: Component, Designing Class-Based Components, Conducting Component-level Design, Component Level Design for WebApps, Designing Traditional Components, Component-Based Development.

Unit IV:

User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation.

WebApp Design: WebApp Design Quality, Design Goal, A Desigin Pyramid for WebApps, WebApp Interface Design, Aestheic Design, Content Design, Architecture Design, Navigation Design, Component-Level Design, Object-Oriented Hypermedia Design Method(OOHMD).

Unit V:

Software Testing Strategies: A strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, The Art of Debugging.

Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, basic Path testing, Control Structure Testing, Black-Box Testing, Model-based Testing, Testing for Specialized Environments, Architectures and Applications, Patterns for Software Testing.Testing Object-Oriented Applications: Broadening the View of Testing, Testing with OOA and OOD Models, Object-Oriented Testing Strategies, Object-Oriented Testing Methods, Testing Methods Applicable at the Class level, Interclass Test-Case Design.

Textbook:

1. “Software engineering A practitioner’s Approach”, Roger S. Pressman, McGraw Hill International Education, Seventh Edition, 2016.

Reference Textbooks:

1. Fundamentals of Software Engineering, Fourth Edition, Rajib Mall, PHI,
2. Software Engineering, Ninth Edition, IAN Sommerville, Pearson, Ninth edition.
3. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India,2010.
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.
6. Software Engineering2: Specification of systems and languages, Diner Bjorner, Springer International edition, 2006.
7. Software Engineering Foundations, Yingxu Wang, Auerbach Publications,2008.
8. Software Engineering Principles and Practice, Hans Van Vliet,3rd edition, John Wiley &Sons Ltd.
9. Software Engineering 3: Domains, Requirements, and Software Design, D.Bjorner, Springer International Edition.
10. Introduction to Software Engineering R.J.Leach, CRC Press

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	L	T	P	C
	4	0	0	4

(17F00305) OPERATING SYSTEMS

Course Objectives:

- To make the students understand the basic operating system concepts such as processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection.
- To get acquaintance with the class of abstractions afford by general purpose operating systems that aid the development of user applications.

Course Outcomes:

- Able to use operating systems effectively.
- Write System and application programs to exploit operating system functionality.
- Add functionality to the exiting operating systems
- Design new operating systems

UNIT I

Operating Systems Overview: Operating system functions, Operating system structure, operating systems Operations, protection and security, Computing Environments, Open- Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, operating system debugging, System Boot.

Processes: Process concept, process Scheduling, Operations on processes, Inter process Communication, Examples of IPC systems.

UNIT II

Threads: overview, Multicore Programming, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues.

Process Synchronization: The critical-section problem, Peterson‘s Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic problems of synchronization, Monitors, Synchronization examples, Alternative approaches.

CPU Scheduling: Scheduling-Criteria, Scheduling Algorithms, Thread Scheduling, Multiple-Processor Scheduling, Real-Time CPU Scheduling, Algorithm Evaluation.

UNIT III

Memory Management: Swapping, contiguous memory allocation, segmentation, paging, structure of the page table.

Virtual memory: demand paging, page-replacement, Allocation of frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory

Deadlocks: System Model, deadlock characterization, Methods of handling Deadlocks, Deadlock prevention, Detection and Avoidance, Recovery from deadlock.

UNIT IV

Mass-storage structure: Overview of Mass-storage structure, Disk structure, Disk attachment, Disk scheduling, Swap-space management, RAID structure, Stable-storage implementation.

File system Interface: The concept of a file, Access Methods, Directory and Disk structure, File system mounting, File sharing, Protection.

File system Implementation: File-system structure, File-system Implementation, Directory Implementation, Allocation Methods, Free-Space management.

UNIT V

I/O systems: I/O Hardware, Application I/O interface, Kernel I/O subsystem, Transforming I/O requests to Hardware operations.

Protection: Goals of Protection, Principles of Protection, Domain of protection, Access Matrix, Implementation of Access Matrix, Access control, Revocation of Access Rights, Capability- Based systems, Language – Based Protection

Security: The Security problem, Program threats, System and Network threats, Cryptography as a security tool, User authentication, Implementing security defenses, Firewalling to protect systems and networks, Computer–security classifications.

Text Books:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Wiley , Eight Edition, 2014.

Reference Books:

1. Operating systems by A K Sharma, Universities Press,
2. Operating Systems, S.Haldar, A.A.Aravind, Pearson Education.
3. Modern Operating Systems, Andrew S Tanenbaum, Second Edition, PHI.
4. Operating Systems, A.S.Godbole, Second Edition, TMH.
5. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
6. Operating Systems, G.Nutt, N.Chaki and S.Neogy, Third Edition, Pearson Education.
7. Operating Systems, R.Elmasri, A,G.Carrick and D.Levine, Mc Graw Hill.
8. Principles of Operating Systems, B.L.Stuart, Cengage learning, India Edition.
9. Operating System Desgin, Douglas Comer, CRC Press, 2nd Edition.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	L	T	P	C
	0	0	4	2

(17F00306) WEB TECHNOLOGIES LAB

Course Outcomes:

- To create a fully functional website with MVC architecture. To develop an online Book store using we can sell books (Ex: amazon .com).

Course Outcomes:

- Ability to apply object oriented concepts for programming and its use.
- Practical WEB Development using java by using JDBC and ODBC connectivity.
- Implementation of servlets and PHP connectivity by using MYSQL applications.
- Learning how to use PHP in different operating systems with different editors like eclipse and net beans.
- Acquire skills to develop final project by acquired knowledge during curriculum.

Week-1:

Design the following static web pages required for an online book store web site.
1) **HOME PAGE:**
The static home page must contain three **frames**.

Top frame : Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description ofthese pages will be given below).
Left frame : At least four links for navigation, which will display the catalogue ofrespective links.
For e.g.: When you click the link **"CSE"** the catalogue for **CSE** Books should be displayed in the Right frame.
Right frame: The *pages to the links in the left frame must be loaded here*. Initially this page contains description ofthe web site.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart

CSE ECE EEE CIVIL	Description ofthe Web Site

Fig 1.1

2) LOGIN PAGE:

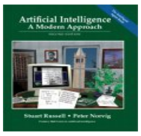
This page looks like below:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Login : □ Passwor□ Submit Reset			

3) CATALOGUE PAGE:

The catalogue page should contain the details ofall the books available in the web site in a table. The details should contain the following:

1. Snap shot ofCover Page.
2. Author Name.
3. Publisher.
4. Price.
5. Add to cart button.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	  	Book : XML Bible Author : Winston Publication : Wiely Book : AI Author : S.Russel Publication : Princeton hall Book : Java 2 Author : Watson Publication : BPB publications Book : HTML in 24 hours Author : Sam Peter Publication : Sam publication	$ 40.5 $ 63 $ 35.5 $ 50	Add to cart Add to cart Add to cart Add to cart

Note: Week 2 contains the remaining pages and their description.

Week-2:

4) CART PAGE:

The cart page contains the details about the books which are added to the cart.

The cart page should look like this:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE	**Book name**	**Price**	**Quantity**	**Amount**
ECE				
EEE	Java 2	$35.5	2	$70
CIVIL	XML bible	$40.5	1	$40.5
			Total amount -	$130.5

5) REGISTRATION PAGE:

Create a “*registration form* “with the following fields
1) Name (Text field)
2) Password (password field)
3) E-mail id (text field)
4) Phone number (text field)
5) Sex (radio button)
6) Date ofbirth (3 select boxes)
7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
8) Address (text area)

WEEK 3:

VALIDATION:
Write *JavaScript* to validate the following fields ofthe above registration page.

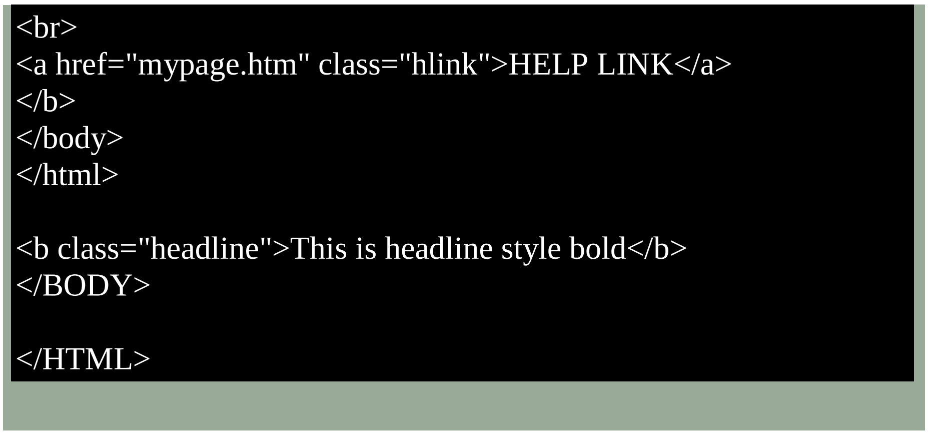
1. Name (Name should contains alphabets and the length should not be less than 6 characters).
2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

Note : You can also validate the login page with these parameters.

Week-4:

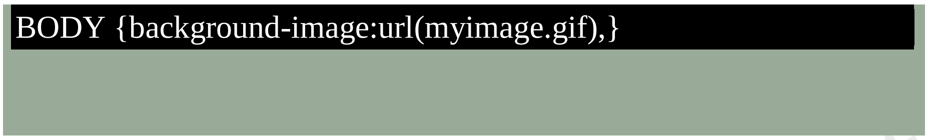
Design a web page using **CSS (C**ascading **S**tyle **S**heets**)** which includes the following:
1) Use different font, styles:
In the style definition you define how each selector should work (font, color etc.). Then, in the body ofyour pages, you refer to these selectors to activate the styles.
For example:

```
<HTML>
<HEAD>
<style type="text/css">
B.headline {color:red, font-size:22px, font-family:arial, text-
decoration:underline}
</style>

</HEAD>

<BODY>
<b>This is normal bold</b><br>
Selector {cursor:value}

For example:

<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink{cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
```



```
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>

<b class="headline">This is headline style bold</b>
</BODY>

</HTML>
```

2) Set a background image for both the page and single elements on the page. You can define the background image for the page like this:



```
BODY {background-image:url(myimage.gif),}
```

3) Control the repetition ofthe image with the background-repeat property. As background-repeat: repeat Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

4) Define styles for links as

```
A:link
A:visited
A:active
A:hover
```

Example:

```
<style type="text/css">
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline, color: red,}
</style>
```

5) Work with layers:

For example:

LAYER 1 ON TOP:

<div style="position:relative, font-size:50px, z-index:2,">LAYER 1</div><div style="position:relative, top:-50, left:5, color:red, font-size:80px, z-index:1">LAYER 2</div>

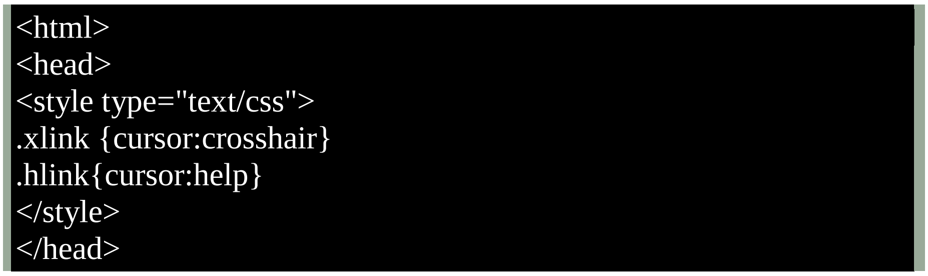
LAYER 2 ON TOP:

<div style="position:relative, font-size:50px, z-index:3,">LAYER 1</div><div style="position:relative, top:-50, left:5, color:red, font-size:80px, z-index:4">LAYER 2</div>

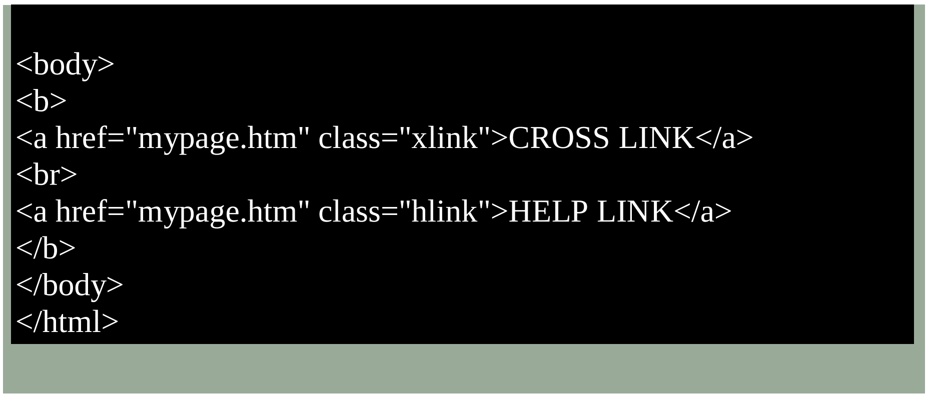
6) Add a customized cursor:

Selector {cursor:value}

For example:



```
<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink{cursor:help}
</style>
</head>
```



```
<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
```

Week-5:

Write an XML file which will display the Book information which includes the following:

1) Title ofthe book
2) Author Name
3) ISBN number
4) Publisher name
5) Edition
6) Price

Write a Document Type Definition (DTD) to validate the above XML file.
Display the XML file as follows.
The contents should be displayed in a table. The header ofthe table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and in bold. Use your own colors for remaining columns.
Use XML schemas XSL and CSS for the above purpose.
Note: Give at least for 4 books. It should be valid syntactically.
Hint: You can use some xml editors like XML-spy

Week-6:
VISUAL BEANS:
Create a simple visual bean with a area filled with a color.
The shape ofthe area depends on the property shape. If it is set to true then the shape ofthe area is Square and it is Circle, if it is false.
The color ofthe area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the “property window “.

Week-7:

1) Install IIS web server and APACHE.

While installation assign port number 4040 to IIS and 8080 to APACHE. Make sure that these ports are available i.e., no other process is using this port.

2) Access the above developed static web pages for books web site, using these servers by putting the web pages developed in week-1 and week-2 in the document root.

Access the pages by using the urls : http://localhost:4040/rama/books.html (for tomcat)
http://localhost:8080/books.html (for Apache)

Week-8:
User Authentication :
Assume four users user1,user2,user3 and user4 having the passwords pwd1,pwd2,pwd3 and pwd4 respectively. Write a PHP for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.

If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display " You are not an authenticated user ''.

Use init-parameters to do this.

Week-9:

Install a database(Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).

Write a PHP program to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.

Insert the details ofthe users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Week-10:

Write a PHP which does the following job:

Insert the details ofthe 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

Week-11:

Create tables in the database which contain the details of items (books in our case like Book name , Price, Quantity, Amount) ofeach category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using PHP

Week-12:

HTTP is a stateless protocol. Session is required to maintain the state.

The user may add some items to cart from the catalog page. He can check the cart page for the selected items. He may visit the catalogue again and select some more items. Here our interest is the selected items should be added to the old cart rather than a new cart. Multiple users can do the same thing at a time(i.e., from different systems in the LAN using the ip-address instead of localhost). This can be achieved through the use of sessions. Every user will have his own session which will be created after his successful login to the website. When the user logs out his session should get invalidated (by using the method session.invalidate()).

Modify your catalogue and cart PHP pages to achieve the above mentioned functionality using sessions.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	L	T	P	C
	0	0	4	2

(17F00307) OPERATING SYSTEMS LAB

Course Objectives:

- To understand the design aspects of operating system
- To solve various synchronization problems

Course out comes:

- Ensure the development of applied skills in operating systems related areas.
- Able to write software routines modules or implementing various concepts of operating system.

1. Simulate the following CPU scheduling algorithms
   a) Round Robin b) SJF c) FCFS d) Priority
2. Simulate all file allocation strategies
   a) Sequential b) Indexed c) Linked
3. Simulate MVT and MFT
4. Simulate all File Organization Techniques
   a) Single level directory b) Two level c) Hierarchical d) DAG
5. Simulate Bankers Algorithm for Dead Lock Avoidance
6. Simulate Bankers Algorithm for Dead Lock Prevention
7. Simulate all page replacement algorithms
   a) FIFO b) LRU c) LFU Etc. …
8. Simulate Paging Technique of memory management
9. Control the number of ports opened by the operating system with
   a) Semaphore b) monitors
10. Simulate how parent and child processes use shared memory and address space
11. Simulate sleeping barber problem
12. Simulate dining philosopher's problem
13. Simulate producer and consumer problem using threads (use java)
14. Simulate little's formula to predict next burst time of a process for SJF scheduling algorithm.
15. Develop a code to detect a cycle in wait-for graph
16. Develop a code to convert virtual address to physical address
17. Simulate how operating system allocates frame to process
18. Simulate the prediction of deadlock in operating system when all the processes announce their resource requirement in advance.

Reference Books :

1. "Operating System Concepts", Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Eighth edition, John Wiley.
2. "Operating Systems: Internals and Design Principles", Stallings, Sixth Edition–2009,Pearson Education
3. "Modern Operating Systems", Andrew S Tanenbaum, Second Edition, PHI.
4. "Operating Systems", S.Haldar, A.A.Aravind, Pearson Education.
5. "Principles of Operating Systems", B.L.Stuart, Cengage learning, India Edition.2013-2014
6. "Operating Systems", A.S.Godbole, Second Edition, TMH.
7. "An Introduction to Operating Systems", P.C.P. Bhatt, PHI.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

MCA II Year I Sem	L	T	P	C
	0	0	4	2

(17F00308) DATABASE MANAGEMENT SYSTEMS LAB

Course Objectives:

- To create a database and query it using SQL, design forms and generate reports.
- Understand the significance of integrity constraints, referential integrity constraints, triggers, assertions.

Course Outcomes:

- Design databases
- Retrieve information from data bases
- Use procedures to program the data access and manipulation
- Create user interfaces and generate reports

List of Experiments:

1. Practice session: Students should be allowed to choose appropriate DBMS software, install it, configure it and start working on it. Create sample tables, execute some queries, use SQLPLUS features, use PL/SQL features like cursors on sample database. Students should be permitted to practice appropriate User interface creation tool and Report generation tool.

2. A college consists of number of employees working in different departments. In this context, create two tables **employee** and **department**. Employee consists of columns empno, empname, basic, hra, da, deductions, gross, net, date-of-birth. The calculation of hra,da are as per the rules of the college. Initially only empno, empname, basic have valid values. Other values are to be computed and updated later. Department contains deptno, deptname, and description columns. Deptno is the primary key in department table and referential integrity constraint exists between employee and department tables. Perform the following operations on the the database:

- Create tables department and employee with required constraints.
- Initially only the few columns (essential) are to be added. Add the remaining columns separately by using appropriate SQL command
- Basic column should not be null
- Add constraint that basic should not be less than 5000.
- Calculate hra,da,gross and net by using PL/SQL program.
- Whenever salary is updated and its value becomes less than 5000 a trigger has to be raised preventing the operation.
- The assertions are: hra should not be less than 10% of basic and da should not be less than 50% of basic.
- The percentage of hra and da are to be stored separately.
- When the da becomes more than 100%, a message has to be generated and with user permission da has to be merged with basic.
- Empno should be unique and has to be generated automatically.
- If the employee is going to retire in a particular month, automatically a message has to be generated.
- The default value for date-of-birth is 1 jan, 1970.
- When the employees called daily-wagers are to be added the constraint that salary should be greater than or equal to 5000 should be dropped.
- Display the information of the employees and departments with description of the fields.
- Display the average salary of all the departments.
- Display the average salary department wise.
- Display the maximum salary of each department and also all departments put together.
- Commit the changes whenever required and rollback if necessary.
- Use substitution variables to insert values repeatedly.
- Assume some of the employees have given wrong information about date-of-birth. Update the corresponding tables to change the value.
- Find the employees whose salary is between 5000 and 10000 but not exactly 7500.
- Find the employees whose name contains ‘en’.
- Try to delete a particular deptno. What happens if there are employees in it and if there are no employees.
- Create alias for columns and use them in queries.
- List the employees according to ascending order of salary.
- List the employees according to ascending order of salary in each department.
- Use ‘&&’ wherever necessary
- Amount 6000 has to be deducted as CM relief fund in a particular month which has to be accepted as input from the user. Whenever the salary becomes negative it has to be maintained as 1000 and the deduction amount for those employees is reduced appropriately.
- The retirement age is 60 years. Display the retirement day of all the employees.
- If salary of all the employees is increased by 10% every year, what is the salary of all the employees at retirement time.
- Find the employees who are born in leap year.
- Find the employees who are born on feb 29.
- Find the departments where the salary of atleast one employee is more than 20000.
- Find the departments where the salary of all the employees is less than 20000.

- On first January of every year a bonus of 10% has to be given to all the employees. The amount has to be deducted equally in the next 5 months. Write procedures for it.
- As a designer identify the views that may have to be supported and create views.
- As a designer identify the PL/SQL procedures necessary and create them using cursors.

Use appropriate Visual programming tools like oracle forms and reports, visual basic etc to create user interface screens and generate reports.

Note: As a designer identify other operations that may be required and add to the above list. The above operations are not in order. Order them appropriately. Use SQL or PL/SQL depending on the requirement.

3. Students may be divided into batches and the following experiments may be given to them to better understand the DBMS concepts. Students should gather the required information, draw ER diagrams, map them to tables, normalize, create tables, triggers, procedures, execute queries, create user interfaces, and generate reports.

- Student information system
- APSRTC reservation system
- Hostel management
- Library management
- Indian Railways reservation
- Super market management
- Postal system
- Banking system
- Courier system
- Publishing house system

References:

1. “Oracle Database 11g PL/SQL Programming”, M.Mc Laughlin,TMH.
2. “Learning Oracle SQL and PL/SQL”, Rajeeb C. Chatterjee, PHI.
3. “Introduction to SQL”, Rick F.Vander Lans, Pearson education.
4. “Oracle PL/SQL”, B.Rosenzweig and E.Silvestrova, Pearson education.