

Database Management System
Prof. Partha Pratim Das
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture – 21
Application Design and Development

Welcome to module 21 of Database Management Systems in this module and the next too we will discuss about application design and development.

(Refer Slide Time: 00:31)



The slide is titled "Week 04 Recap" and lists the following modules and topics:

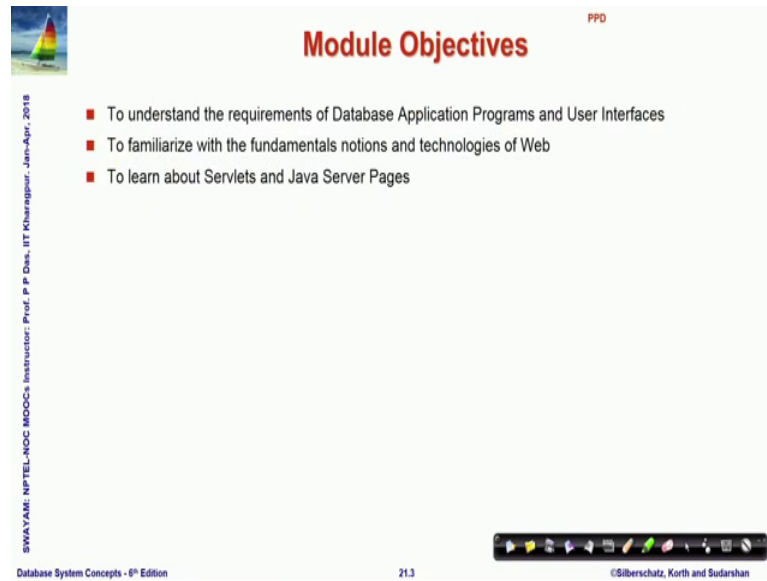
- Module 16: Relational Database Design/1**
 - Features of Good Relational Design
 - Atomic Domains and First Normal Form
 - Functional Dependencies
- Module 17: Relational Database Design/2**
 - Decomposition Using Functional Dependencies
 - Functional Dependency Theory
- Module 18: Relational Database Design/3**
 - Algorithms for Functional Dependencies
 - Lossless Join Decomposition
 - Dependency Preservation
- Module 19: Relational Database Design/4**
 - Normal Forms
 - Decomposition to 3NF
 - Decomposition to BCNF
- Module 20: Relational Database Design/5**
 - Multivalued Dependencies
 - Decomposition to 4NF
 - Database-Design Process
 - Modeling Temporal Data

SWAYAM: NPTEL-NOC MOOC's Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.2 ©Silberschatz, Korth and Sudarshan

In the last week we have for the whole week in the five modules we have discussed about relational database design; in depth we have looked into what are the different aspects of that.

(Refer Slide Time: 00:43)



The slide is titled "Module Objectives" in a large, bold, red font. It features a small image of a sailboat in the top left corner. The objectives are listed in a red square bullet format. At the bottom, there is a navigation bar with various icons and a copyright notice.

PPD

Module Objectives

- To understand the requirements of Database Application Programs and User Interfaces
- To familiarize with the fundamentals notions and technologies of Web
- To learn about Servlets and Java Server Pages

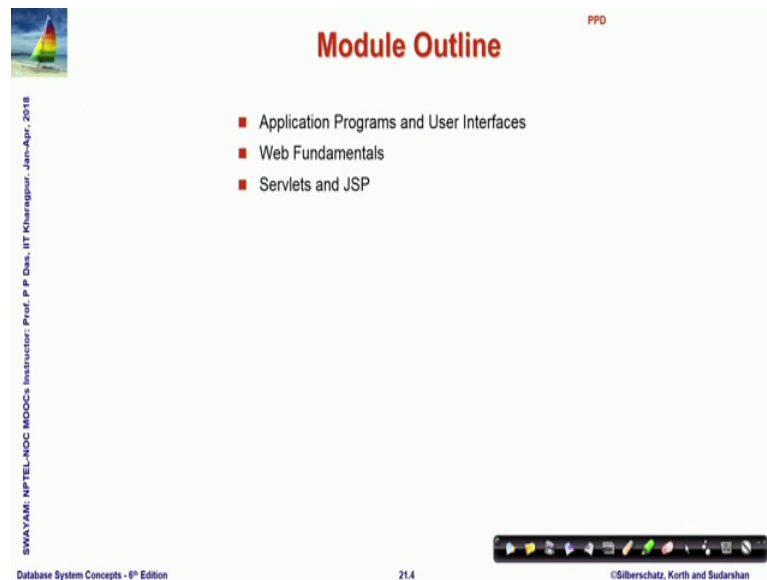
SWAYAM: NPTEL-NOC MDOCS Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.3 ©Silberschatz, Korth and Sudarshan

And now we get into the core issue of if I have a relational database design existing and that is populated with the data then based on that how do we develop, how do we create an application where the user can interact and actually get answers to the questions that the user has or the user can actually update the data create new data, remove old data and so, on.

So, we would in that process like to familiarize with the fundamental notions of notions and technologies of web applications and specifically we would learned about servlets and Java server pages.

(Refer Slide Time: 01:30)



PPD

Module Outline

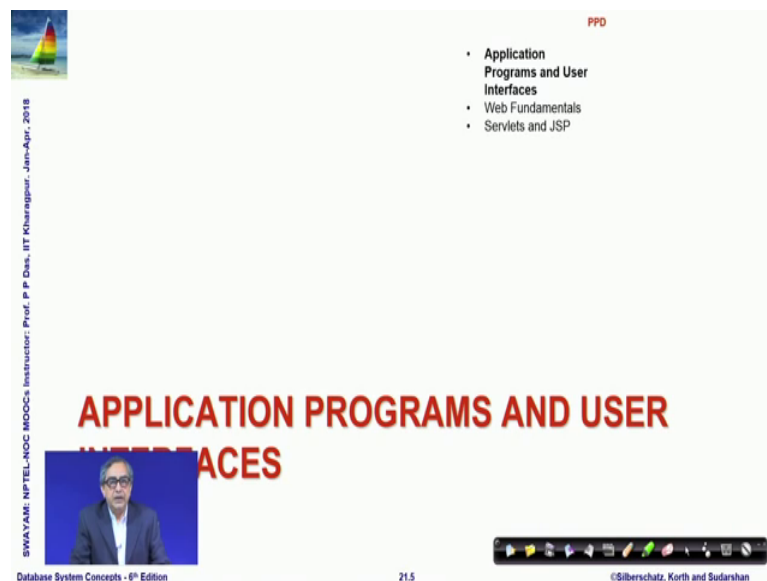
- Application Programs and User Interfaces
- Web Fundamentals
- Servlets and JSP

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P P Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.4 ©Silberschatz, Korth and Sudarshan

So, these are the free topics to be covered.

(Refer Slide Time: 01:34)



PPD

- Application Programs and User Interfaces
- Web Fundamentals
- Servlets and JSP

APPLICATION PROGRAMS AND USER INTERFACES

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P P Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.5 ©Silberschatz, Korth and Sudarshan

So, we first start with application programs and user interfaces.

(Refer Slide Time: 01:41)

Application Programs and User Interfaces

- Most database users do *not* use a query language like SQL
- An application program acts as the intermediary between users and the database
 - Applications split into
 - frontend
 - middle layer
 - backend
- Frontend or Presentation Layer: user interface
 - Forms, Graphical user interfaces
 - Many interfaces are Web-based or Mobile App
- Middle Layer or Application / Business Logic Layer
 - Functionality of the Application – links front and backend
- Backend or Data Access Layer
 - Persistent data, large in volume, needs efficient access

Overview of a 3-tier Architecture

The diagram illustrates a 3-tier architecture. At the top is the **Presentation tier**, which includes a user interface (UI) and a user. Below it is the **Logic tier**, which contains application logic and business rules. At the bottom is the **Data tier**, which includes a database and storage. Arrows indicate the flow of data and control between these tiers. The Logic tier sends queries to the Data tier and receives data back. The Presentation tier sends user input to the Logic tier and receives output from it.

PPD

SWAYAM: NPTEL-NOC MOCs Instructor: Prof. P. P. Das, IIT Khargpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.6 ©Silberschatz, Korth and Sudarshan

So, the situation is the where we do have a relational database design it is populated with the required data, but how about the interaction with the user incidentally most of the you database users do not interact with the database or query the database using language like SQL because as you have seen it is not a very friendly language and it is not presentable in a way which I would we would always expect or we would like.

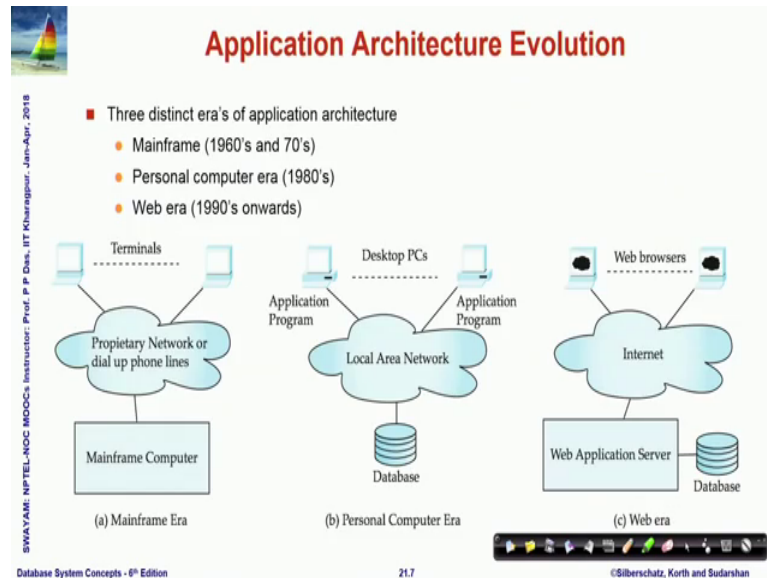
So, usually an application program acts as a as we say as an intermediately between the user and the database. And it is often split into three layers as we will say frontend middle layer and backend. So, on the right I have shown these three layers we will talk more about the this is just a representative diagram. So, the frontend is the user interface it is also called the presentation layer. So, it is the layer where it is the part of the application where there are forms GUIs and different ways to input as well as get output of the data.

Which is directly interacting with the user then we have a middle layer or its also called the business logic layer or the application layer where the functionality of the application the required operations of the or desired behavior of the application are coded and it is kind of acts as a link between the frontend and the backend and what is the backend?.

Backend is a actual data access layer or it is where the persistent data the database that we have created exist it is typically large in volume need sufficient access and so, on. So, in this module we will try to understand as to how such what are the different

requirements and what are the different technologies involved in creating such a layered application.

(Refer Slide Time: 03:55)



Now, if we historically look at; so, here we are just showing three phases initially 60s and 70s where the first database applications started then the interaction used to be takes based from the terminal.

And the those to directly connect to the main frame computer where the data existed through either a direct connection dial up phone or proprietary network. And as we move to the 80s; then we saw the advent of local area networks to application programs or desktop would interact to with the database through these local area networks. And beyond that we have the what we call the web era which is 1990 onwards we in the that is that is about roughly the last 30 years where typically the applications are now based on web browsers.

So, the frontend where we actually interact are web browsers and that connect to the web application server the database everything through an internet. And I must tell you at this point that when you say this is the architecture it does not necessarily mean that the cloud shown as internet will have to be the web, it could be an internet which is created with a set of systems within your organization which we typically call as an intranet or couple of organizations across.

Which we say are extranet or it could even be a set of systems which are connected through the internet protocol within your lap or it could even be a single computer in which all these layers are integrated together, but by internet we mean it is a internet protocol and technologies will be used for doing this interaction.

(Refer Slide Time: 05:52)

Web Interface

- Web browsers have become the de-facto standard user interface to databases
 - Enable large numbers of users to access databases from anywhere
 - Avoid the need for downloading/installing specialized code, while providing a good graphical user interface
 - Javascript, Flash and other scripting languages run in browser, but are downloaded transparently
 - Examples: banks, airline and rental car reservations, university course registration and grading, an so on.
- Mobile Interfaces in Mobile Apps are getting popular
 - These are similar in architecture and workflow with web, but have significant differences
 - Will be discussed later

SWAYAM NPTEL-NOC MOOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition

21.8

©Silberschatz, Korth and Sudarshan

So, web interface has become the de-facto standard which gives a very distributed access to the database enables large number of users to access together. And it avoids the requirement of downloading or installing specialized code into that all that we need is just a web browser. And we have seen we are living through a variety of applications which are of this kind the banking application, the airline and railway reservations car rental hotel booking or web mail systems we will check mail in Gmail or Yahoo those are all different web interfaces through which we actually access a the required set of databases.

And every even every enterprise operations the ERP are now web based and that is become a de facto standard. So, in the web interface along with a web interface what has been imagined of let of the last about 10 years are mobile interfaces that we are getting use to using such applications from our mobile phone or tablet. And these are similar in architecture and workflow as of the web application, but there are significant differences to and at a later point in the next module, we will discuss about the specific requirements of mobile apps in this context as well.

(Refer Slide Time: 07:27)

PPD

- Application Programs and User Interfaces
- **Web Fundamentals**
- Servlets and JSP

WEB FUNDAMENTALS

SWAYAM: NPTEL-NOC MOCs Instructor: Prof. P P Das, IIT Kharragpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.9 ©Silberschatz, Korth and Sudarshan

So, before we move forward in terms of the details of how to build these applications; we need to familiarize ourselves with the basic notions of the web as such. So, we call them as web fundamentals.

(Refer Slide Time: 07:43)

The World Wide Web

- The Web is a distributed information system based on hypertext
- Most Web documents are hypertext documents formatted via the HyperText Markup Language (HTML)
- HTML documents contain
 - text along with font specifications, and other formatting instructions
 - hypertext links to other documents, which can be associated with regions of the text
 - forms, enabling users to enter data which can then be sent back to the Web server

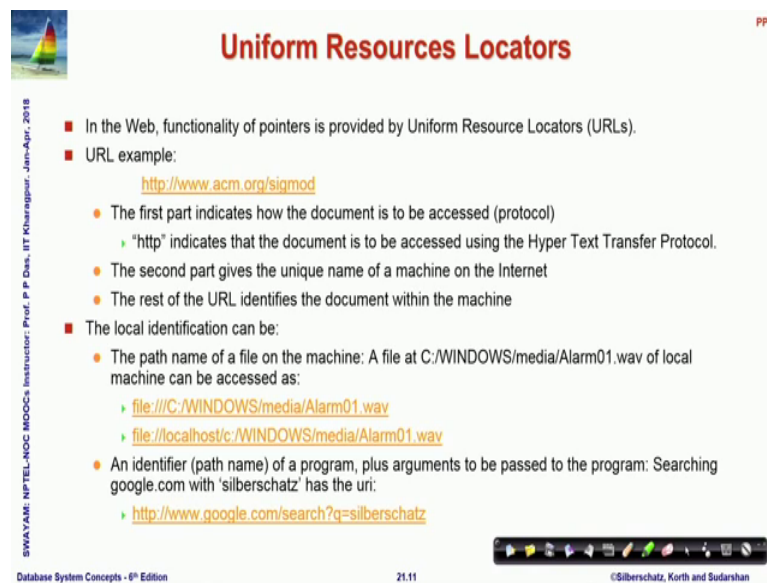
SWAYAM: NPTEL-NOC MOCs Instructor: Prof. P P Das, IIT Kharragpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.10 ©Silberschatz, Korth and Sudarshan

So, web is a distributed information system which is based on hyper text a hyper text is one where you have a part of the text available to you and then you have links we say our hyper links which can connect to the different documents contents.

Which are locate at other places at other servers and typically most web documents are hyper text documents which are formatted in terms of what we know as HTML Hyper Text Markup Language; you will be familiar with that I am sure. So, they contain text and along with that they can have a other components like images, video, the text as specifications for font colors style all that. And in addition there are forms which can be used to enter data and send them back to the web server.

(Refer Slide Time: 08:35)



Uniform Resources Locators

- In the Web, functionality of pointers is provided by Uniform Resource Locators (URLs).
- URL example:
 - <http://www.acm.org/sigmod>
 - The first part indicates how the document is to be accessed (protocol)
 - ▶ "http" indicates that the document is to be accessed using the Hyper Text Transfer Protocol.
 - The second part gives the unique name of a machine on the Internet
 - The rest of the URL identifies the document within the machine
 - The local identification can be:
 - The path name of a file on the machine: A file at C:/WINDOWS/media/Alarm01.wav of local machine can be accessed as:
 - ▶ <file:///C:/WINDOWS/media/Alarm01.wav>
 - ▶ <file://localhost/c:/WINDOWS/media/Alarm01.wav>
 - An identifier (path name) of a program, plus arguments to be passed to the program: Searching google.com with 'silberschatz' has the uri:
 - ▶ <http://www.google.com/search?q=silberschatz>

Database System Concepts - 6th Edition 21.11 ©Silberschatz, Korth and Sudarshan

Now, naturally when we operate on the web we need to have the functionality of pointing to different resources and this is done by what is known as URL or uniform resource locator. So, that is a URL is a procedure to which you can identify and point to a certain specific location of content. So, here I am showing an example of such a URL all of you be familiar with URLs.

But just to look into the different components the first component http colon this http is actually a tells us the way the content would be accessed and this is typically called a protocol http its stands for hyper text transfer protocol which allows you different text to be accessed.

The second component in this URL which is between the two forward slash and the next slash www dot acm dot org identifies uniquely identifies a machine on the internet you will understand this is the symbolic name and the actual machine has what is known as an IP address.

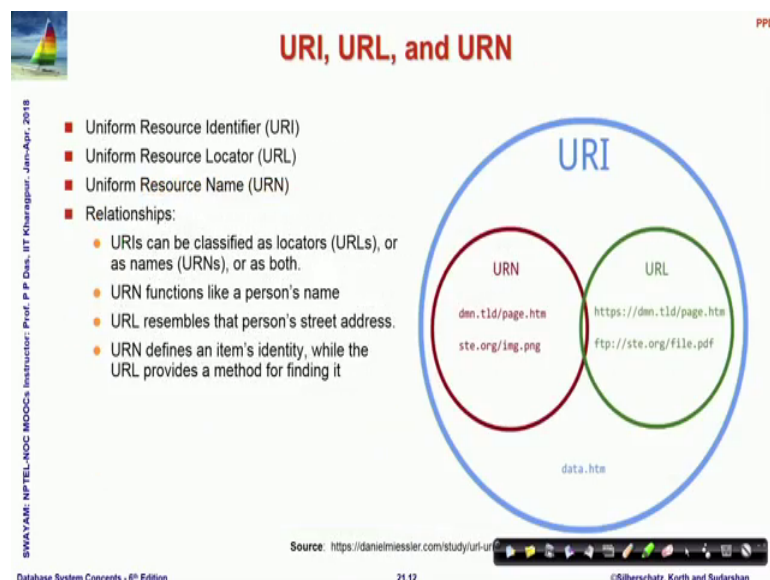
And we will not go into those details, but for us it is enough to understand that www dot acm dot org here is uniquely relatable to a particular machine on the internet. And the last part which is remaining sigmod and if there are more and more and more parts, then it identifies the document inside within that machine. So, URL can be used in a multiple other ways also.

For example I can use URL locator to specify a file in by machine for example, I have shown an example of an AVI file in my c drive in windows and that is done through a similar URL where the protocol is not http the protocol is file telling me that it is actually residing in my local machines.

So, I have shown two ways to look at that and you will see that between the two forms; if you look at the second form you can easily understand that the machine to be identify this is called the local host. And in the first form that part is missing because it is by default the machine where I am running this code and rest of it is same which is basically the identifying the document to be to be located in that machine.

Similarly, this such URL can also in the last example you can see that www dot Google dot com is the basic machine where I am putting the URL and then the rest of it is search.

(Refer Slide Time: 11:56)



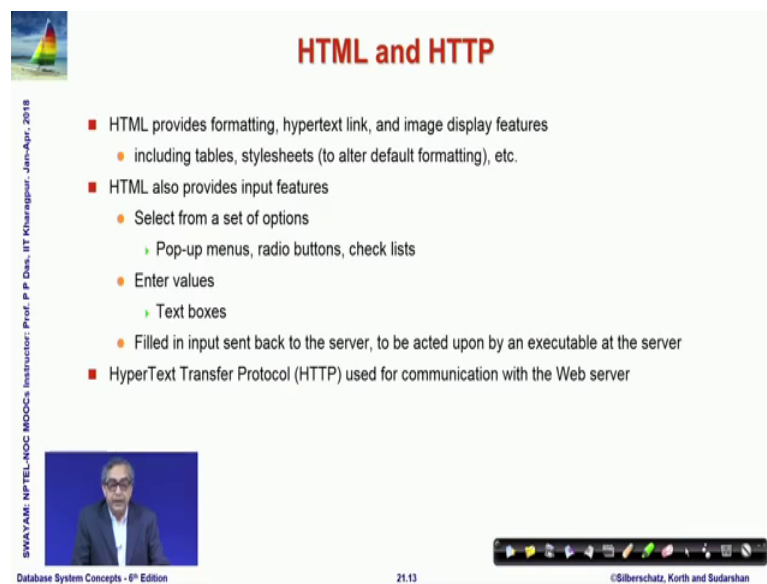
So, which actually takes it to a document where I tell the search to be performed and then there are parameter to this form the parameter is q is equal to silberschatz which is

equivalent to same that I am asking Google dot com to search for contents which have silberschatz in it. So, this is the basic purpose of the uniform resource locator or URL. Incidentally you may have heard the names like URI and URN in addition to URL.

So, they are related, but they mean little bit different things as this venn diagram shows. So, a URI can be either a URL or a URN or it could be both. So, URN functions like a persons name; so, you can conceive it that way universal resource name and URL resembles that of a persons street address. So, URN says what is the name of the content and URL says where that can be found. And URI in general could be either the name or the address or both of them.

In this context of our discussion we will continue to use the term URL only, but I just wanted you to be aware of the other two terms in case you come across them in the text.

(Refer Slide Time: 12:53)

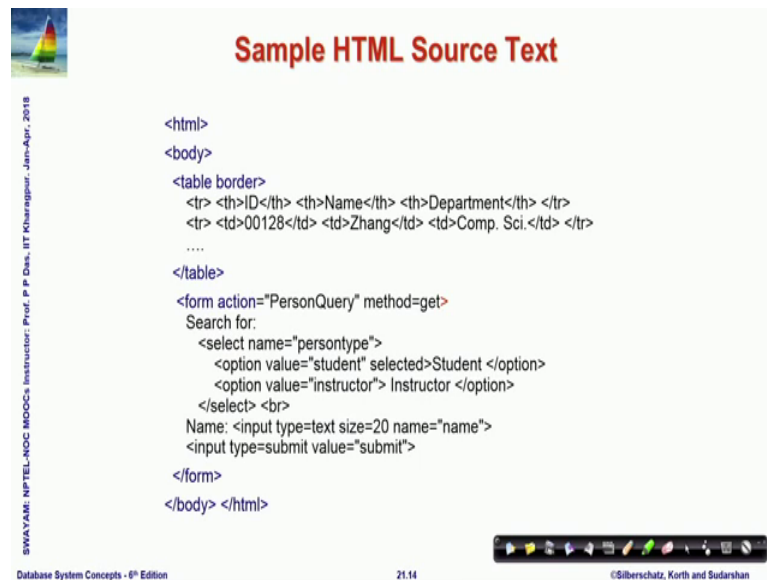


The slide is titled "HTML and HTTP" in red text. It features a small image of a sailboat in the top left corner. The main content is a bulleted list of HTML and HTTP features. On the left side, there is a vertical text string: "SWAYAM: NPTEL-NOC MOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018". At the bottom left, there is a small video inset showing a man speaking. At the bottom right, there is a navigation bar with various icons. The footer contains the text "Database System Concepts - 9th Edition", "21.13", and "©Silberschatz, Korth and Sudarshan".

- HTML provides formatting, hypertext link, and image display features
 - including tables, stylesheets (to alter default formatting), etc.
- HTML also provides input features
 - Select from a set of options
 - Pop-up menus, radio buttons, check lists
 - Enter values
 - Text boxes
 - Filled in input sent back to the server, to be acted upon by an executable at the server
- HyperText Transfer Protocol (HTTP) used for communication with the Web server

So, we know by now that http HTML provides the formatting the hyper text links images and so, on and http provides the protocol through which the contents are exchanged between different machines in the internet. So, you can select from a set of options in terms of a HTML popup menus radio buttons check boxes and so, on; you can enter values to text box and once a form has been filled up that form will be sent back to the server from where it came and would be acted upon by the server http helps in that transfer mechanism.

(Refer Slide Time: 13:39)



Sample HTML Source Text

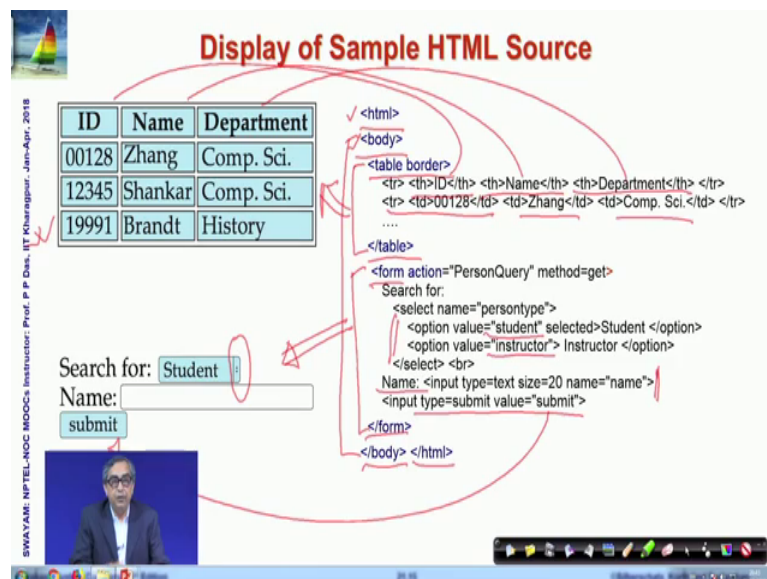
```
<html>
<body>
  <table border>
    <tr> <th>ID</th> <th>Name</th> <th>Department</th> </tr>
    <tr> <td>00128</td> <td>Zhang</td> <td>Comp. Sci.</td> </tr>
    ....
  </table>
  <form action="PersonQuery" method=get>
    Search for:
    <select name="persontype">
      <option value="student" selected>Student </option>
      <option value="instructor"> Instructor </option>
    </select> <br>
    Name: <input type=text size=20 name="name">
    <input type=submit value="submit">
  </form>
</body> </html>
```

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P P Das, IIT Khargpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.14 ©Silberschatz, Korth and Sudarshan

So, the here is a sample HTML code let me show you the effect of this in the next slide.

(Refer Slide Time: 13:46)



Display of Sample HTML Source

ID	Name	Department
00128	Zhang	Comp. Sci.
12345	Shankar	Comp. Sci.
19991	Brandt	History

Search for:

Name:

```
<html>
<body>
  <table border>
    <tr> <th>ID</th> <th>Name</th> <th>Department</th> </tr>
    <tr> <td>00128</td> <td>Zhang</td> <td>Comp. Sci.</td> </tr>
    ....
  </table>
  <form action="PersonQuery" method=get>
    Search for:
    <select name="persontype">
      <option value="student" selected>Student </option>
      <option value="instructor"> Instructor </option>
    </select> <br>
    Name: <input type=text size=20 name="name">
    <input type=submit value="submit">
  </form>
</body> </html>
```

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P P Das, IIT Khargpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.15 ©Silberschatz, Korth and Sudarshan

So, if you look in here then you can see that these are this is what is known as a tag and it kind of the tax are kind of given in the form of parenthesize notation.

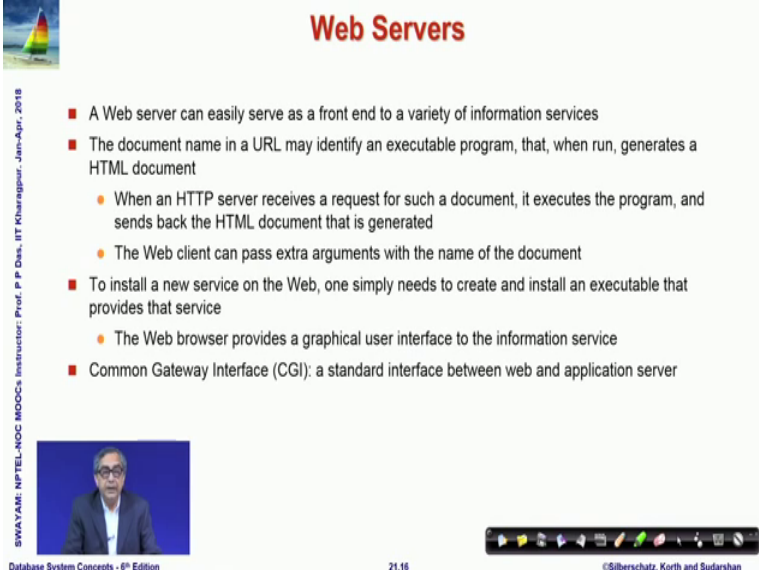
So, it has a opening and a closing and these have to have to actually match and you can see that the opening is written within corner brackets and the closing is write the same tag name, but you put a forward slash for the name. And then between the closing opening and the closing you can write more tabs in a nested manner.

So, here it says that I have HTML which has a body and the body expands this much and then we are saying that there is a table. So, this is the table that you can get to see and then it also says that there is a form and this is the form that you get to see within the table you can see. So, it is saying that this is an ID there is a name; so, it is describing the first row the department. So, you can see each one of them the ID is here the name is here the department is here similarly this is a next row where it is saying it is 0 1; 00128 Zhang Computer Science.

So, this is how you can actually in an HTML in a text form all these details will be given and when it is rendered by the web browser then you will see a table like this. Similarly here we are I am showing an instance of a form which is used to input data. So, we are saying that here is a drop down and it is written out here in terms of options.

So, the first option student is visible here if you drop down you will actually see another option instructor here you will not see that because it is a frozen image. So, and then I have a qualifier name and there is an input text box where you can input any string up to size 20. And once this has been done then you have an input which is submit, which is the submit button here which shows that you can now submit and then this form filled up form will be sent back to the web browser from where it originally came.

(Refer Slide Time: 16:23)



Web Servers

- A Web server can easily serve as a front end to a variety of information services
- The document name in a URL may identify an executable program, that, when run, generates a HTML document
 - When an HTTP server receives a request for such a document, it executes the program, and sends back the HTML document that is generated
 - The Web client can pass extra arguments with the name of the document
- To install a new service on the Web, one simply needs to create and install an executable that provides that service
 - The Web browser provides a graphical user interface to the information service
- Common Gateway Interface (CGI): a standard interface between web and application server

SWAYAM: NPTEL-NOC IMOOCs Instructor: Prof. P. Das, IIT Kharagpur, Jan-April, 2018

Database System Concepts - 9th Edition 21.16 ©Silberschatz, Korth and Sudarshan

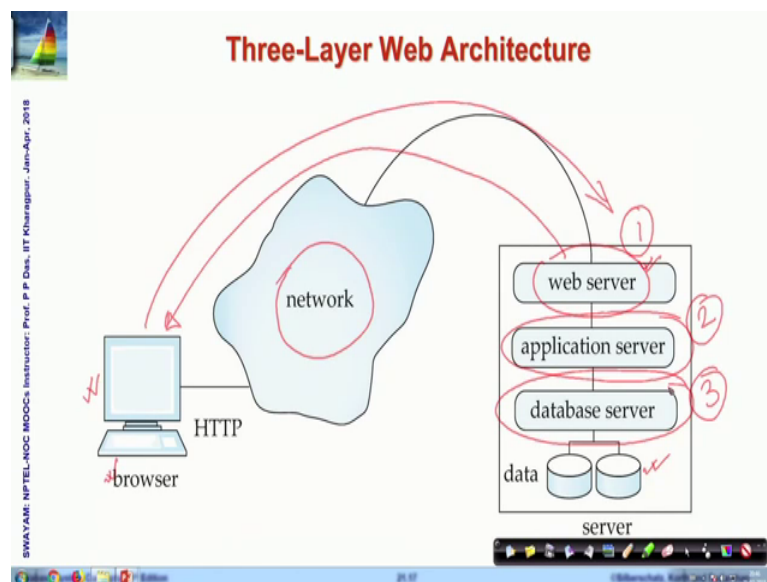
So, this is the basic mechanism of HTML you can learn a little bit more about that and get familiar with it. So, a document name in a URL may identify a program that is written

that generate. So, HTML could be either at the URL in the web server you can either have a HTML which we say is a static HTML or you could actually have a program which when you send the request it actually.

For example, when you are doing Google you said www dot Google dot com slash search question mark q is has a value silberschatz. Then actually at that location there is no HTML currently existing which contains the search result, but instead there is a program which will be executed based on the submission of this form and when run that will generate a HTML document. And once that is generated then this will be passed back to the to your web browser. So, that is a basic mechanism.

So, if you want a new service on the web then you all that you simply need to do is to create and install a new program that will provide that service and through this process we will see how easily this can be done and how web browser provides a graphical interface to this information service. There is there has been another other mechanisms of doing similar things also which was particularly popularly are called the common gateway interface or CGI, but now we have various other ways of doing the same thing.

(Refer Slide Time: 18:03)

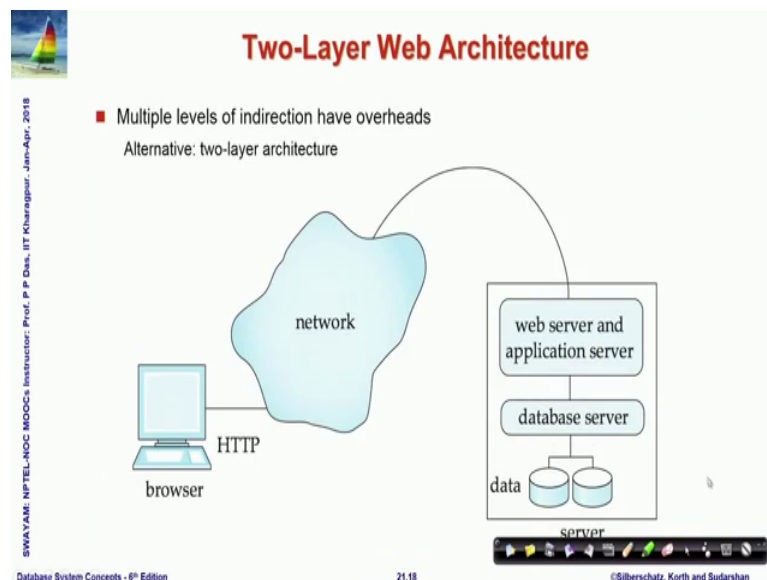


So, in this context then the basic three layers that we started discussing with are here. So, this is the most common way an application is. So, this is where your frontend is where you have the browser where you see that this is the network; we are just in general writing network it could be internet it could be intranet and a web server. So, when you

send a request this is received by the server; web server somehow computes a result HTML and that send back to the browser and that is how the interaction keeps on happening.

So, the browser and the web server together often would be refer to as a frontend because that gives a presentation that presents the results the interaction to you. The next is the application layer which is the business logic where you write and then you have the data access layer or the database server and these are the actual disk where the data exist. So, this is a tier 1 this is tier 2 and this is tier 3 which is a very typical way a web application will be architected and these are the three layers or three tiers that we will usually find.

(Refer Slide Time: 19:22)



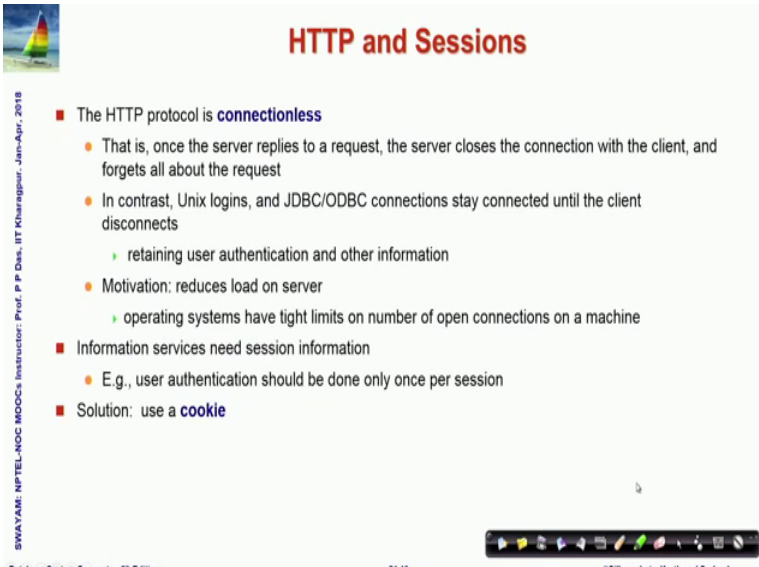
So, often actually three is not a very magical number in terms of tiers; it is possible that you could have more some applications have more tiers. And some applications may choose to have multiple functionality in the same layer for example, web server and application server functionality could be clubbed together and when this is done we say that we will then we have only two layers. So, the frontend and the middle layer are merge together and the backend or the database server becomes a second layer.

So, we would often might want to do that the reason we do that I will just take to back to the three layer view. So, if the question is naturally if we have the web server and we have these connections this is clear; now the question is what is this connection and what is this connection? Is it necessary that they will have to be on the same server physically

or will they be on can be on different server and servers could be connected through a LAN or they themselves could be on different servers over the internet and may they may be connected through internet.

So, all of these are possibilities and the way we connect is the way we will write the application will be will not depend on the way these servers are connected between each other we will often assume that as if they are connected over a net and write it in a way so, that even when they may be connected over a LAN or even when they may actually be on the same machine things will work in the same way.

(Refer Slide Time: 20:57)



HTTP and Sessions

- The HTTP protocol is **connectionless**
 - That is, once the server replies to a request, the server closes the connection with the client, and forgets all about the request
 - In contrast, Unix logins, and JDBC/ODBC connections stay connected until the client disconnects
 - ▶ retaining user authentication and other information
 - Motivation: reduces load on server
 - ▶ operating systems have tight limits on number of open connections on a machine
- Information services need session information
 - E.g., user authentication should be done only once per session
- Solution: use a **cookie**

SWAYAM: NPTEL-NOC IMOOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018
Database System Concepts - 9th Edition 21.19 ©Silberschatz, Korth and Sudarshan

Now, one point that should be born in mind in terms of the http protocol is it is connectionless. So, this is a very very critical concept and it means that once I start the process I have an URL; I submit that and that goes to the server the server runs an application or it is a static page server picks up and returns me that HTML; the http loop is closed.

So, that is the all that happens and when I submit again something based on that. So, I have gone to Gmail mail dot Google dot com I submit that and I get back a form which tells me to put my user ID and login I do that I submit and that when that goes then there is no memory about the earlier interaction that has happen.

So, when my login submission goes it is authenticated in the backend I am able to login and I am given back the first screen of my mail box which is the inbox screen. And as soon as I get that inbox the HTML containing the inbox on my browser that transaction has also been over. So, if I now want to look at a specific mail it has to be a new query and it is not remember anything from the previous query.

So, this connectionless property naturally makes it makes certain things more difficult; you will you will realize that many of the other connections that we do for example, if we login to UNIX system or to a window system if we use some database connections they are connected till the we disconnect them, but in http it is not. So, it is connectionless every time you do you have a separate session. So, naturally the question this was I mean the there are there are reasons of why this is done this way this is to reduce load from the server and so, on.

But naturally the consequences therefore, we cannot remember information from one request response loop to the next. So, if I have logged in to my mail Gmail account and I have seen the I have got the inbox then when I want to check my first mail the system does not know any more that I am logged in because that session request response has is over and now I am making a new request that show me the first mail and I expect to see the whole body.

So, there is no information that is carried from one request to the other which makes http difficult to work with. So, the solution for that is something which is known as a cookie.

(Refer Slide Time: 23:46)

Sessions and Cookies

- A **cookie** is a small piece of text containing identifying information
 - Sent by server to browser
 - Sent on first interaction, to identify session
 - Sent by browser to the server that created the cookie on further interactions
 - part of the HTTP protocol
 - Server saves information about cookies it issued, and can use it when serving a request
 - E.g., authentication information, and user preferences
- Cookies can be stored permanently or for a limited time

SWAYAM: NPTEL-NOC MDOCS Instructor: Prof. P. P. Das, IIT Khargpur, Jan-Apr, 2018

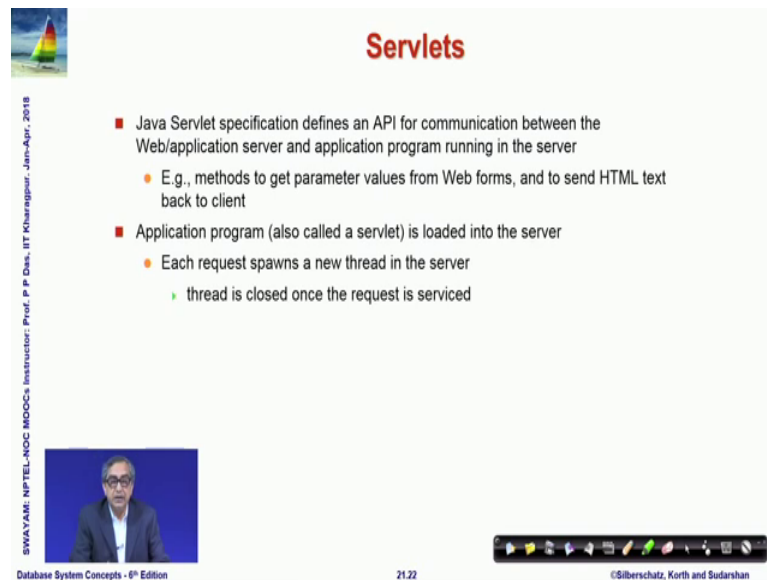
Database System Concepts - 8th Edition 21.20 ©Silberschatz, Korth and Sudarshan

So, a cookie is a small piece of text which contain information which is identifying and which can go back and forth between the browser and the server. So, first it is sent by the sever to the browser and the what the browser does.

So, this happens the first time. So, when we logged in my browser has got a got some cookie from the mail Gmail server. Then the browser can send it back to the server when it is doing the next request so, that I can be identified as a logged in person and. So, the browser can keep it as a I mean locally in its memory or locally here and that is a part of the http protocol.

So, this keeps on this cookie keeps on going back and forth back and forth. So, every time I send a request the cookie actually has to go to tell the server that yes this is the Partha Pratim Das who is already logged in an authenticated himself for checking his mails. So, cookies are a big convenience and they are very important factor of the web applications. So, they can be stored permanently or for a limited period of time.

(Refer Slide Time: 25:04)



Servlets

- Java Servlet specification defines an API for communication between the Web/application server and application program running in the server
 - E.g., methods to get parameter values from Web forms, and to send HTML text back to client
- Application program (also called a servlet) is loaded into the server
 - Each request spawns a new thread in the server
 - ▶ thread is closed once the request is serviced

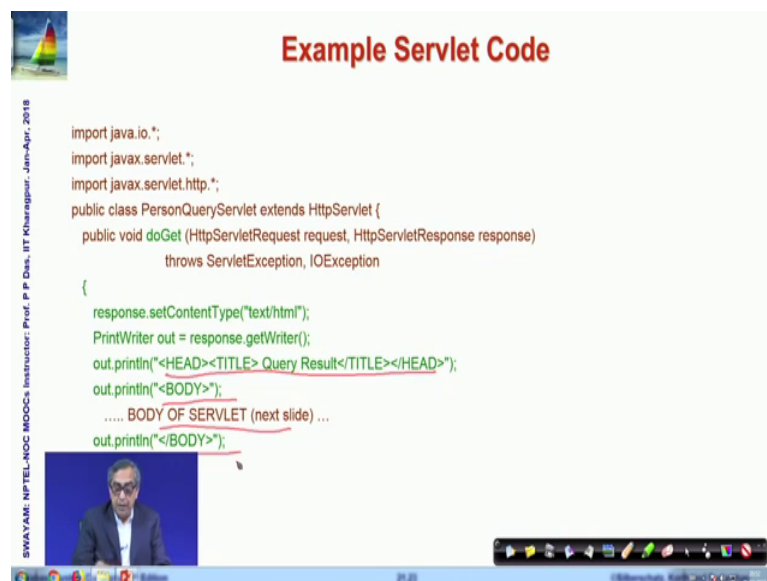
SWAYAM: NPTEL-NOC MDOCS Instructor: Prof. P P Das, IIT Khargpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.22 ©Silberschatz, Korth and Sudarshan

Next let us look into some of the core technologies that are involved the first technology is called a servlet. Servlet is nothing but as you can understand from the name itself is as you have book booklet booklet is a small very small book servlet is a very small server.

So, it is a Java application which can do certain tasks; so, it is an kind of an application program and every time we request then the server actually spawns a new thread and in that thread this servlet would be running and once the request is serviced the thread will be closed.

(Refer Slide Time: 25:42)



Example Servlet Code

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

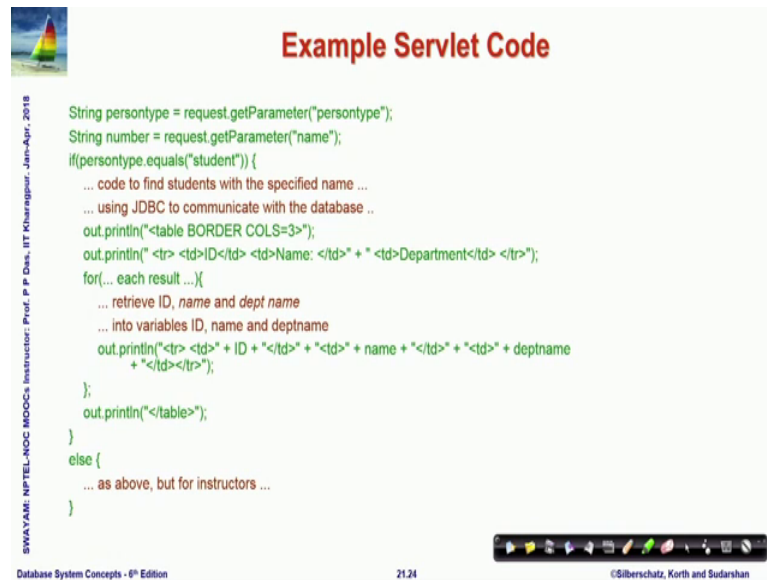
public class PersonQueryServlet extends HttpServlet {
    public void doGet (HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException
    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<HEAD><TITLE> Query Result</TITLE></HEAD>");
        out.println("<BODY>");
        .... BODY OF SERVLET (next slide) ...
        out.println("</BODY>");
    }
}
```

SWAYAM: NPTEL-NOC MDOCS Instructor: Prof. P P Das, IIT Khargpur, Jan-Apr, 2018

Database System Concepts - 8th Edition 21.22 ©Silberschatz, Korth and Sudarshan

So, this is the typical server servlet view. So, which shows that in the servlet you are creating actually creating the requested I mean possible HTML response that you would like to have.

(Refer Slide Time: 25:58)



Example Servlet Code

```
String persontype = request.getParameter("persontype");
String number = request.getParameter("name");
if(persontype.equals("student")) {
    ... code to find students with the specified name ...
    ... using JDBC to communicate with the database ..
    out.println("<table BORDER COLS=3>");
    out.println(" <tr> <td>ID</td> <td>Name: </td> * " <td>Department</td> </tr>");
    for(... each result ...){
        ... retrieve ID, name and dept name
        ... into variables ID, name and deptname
        out.println("<tr> <td>* + ID + "</td>* + "<td>* + name + "</td>* + "<td>* + deptname
        + "</td></tr>");
    };
    out.println("</table>");
}
else {
    ... as above, but for instructors ...
}
```

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.24 ©Silberschatz, Korth and Sudarshan

So, there are; so, there are different details you can read through that. So, this is the typical servlet code. So, it actually is a Java code which through print line will generate different lines of the HTML. Now naturally servlets maintain session the way we talked about.

(Refer Slide Time: 26:11)



Servlet Sessions

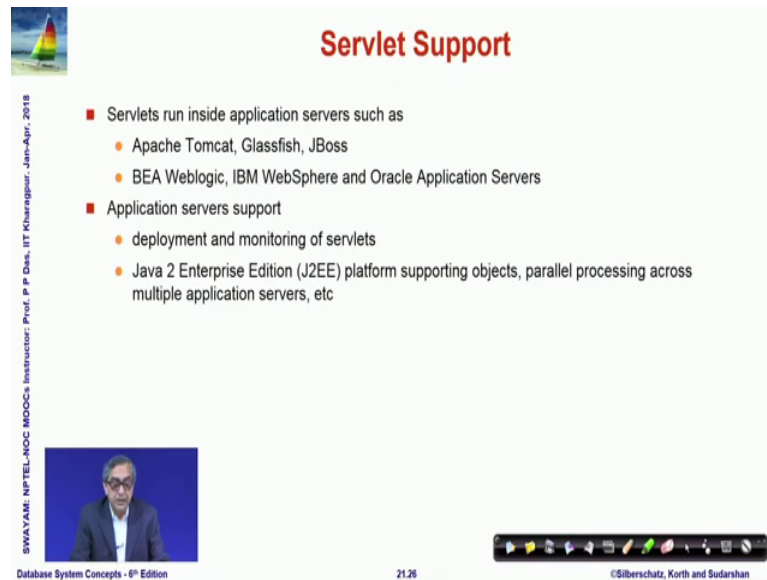
- Servlet API supports handling of sessions
 - Sets a cookie on first interaction with browser, and uses it to identify session on further interactions
- To check if session is already active:
 - if (request.getSession(false) == true)
 - ▶ .. then existing session
 - ▶ else .. redirect to authentication page
 - authentication page
 - ▶ check login/password
 - ▶ request.getSession(true): creates new session
- Store/retrieve attribute value pairs for a particular session
 - session.setAttribute("userid", userid)
 - session.getAttribute("userid")

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.25 ©Silberschatz, Korth and Sudarshan

So, that through an interaction I can continued to be identified and the servlet can check whether the session is on or the session is already over. So, these are these are the different ways of doing that in terms of shaking the user ID and several web servers application servers have support for servlet apache tomcat is one of the very popular one.

(Refer Slide Time: 26:33)



Servlet Support

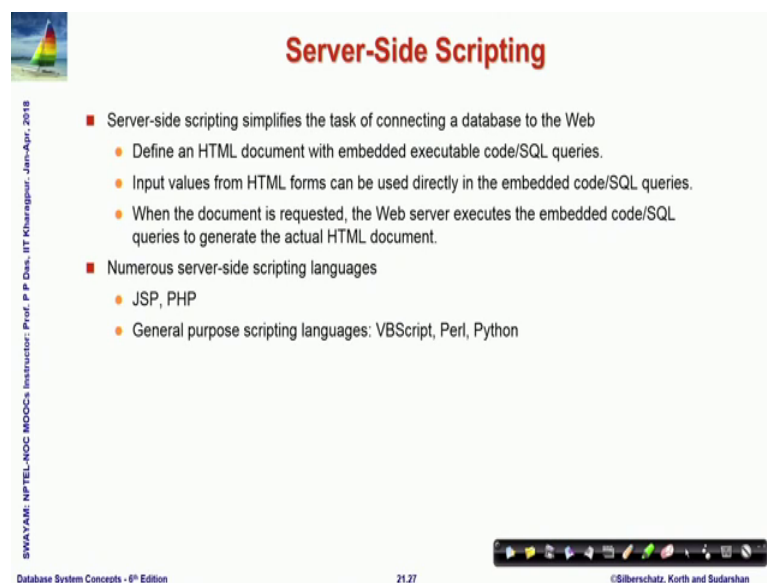
- Servlets run inside application servers such as
 - Apache Tomcat, Glassfish, JBoss
 - BEA Weblogic, IBM WebSphere and Oracle Application Servers
- Application servers support
 - deployment and monitoring of servlets
 - Java 2 Enterprise Edition (J2EE) platform supporting objects, parallel processing across multiple application servers, etc

SWAYAM: NPTEL-NOC MOC's Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.26 ©Silberschatz, Korth and Sudarshan

So, which you must have heard the name of and there are, but there are several other servers as well.

(Refer Slide Time: 26:48)



Server-Side Scripting

- Server-side scripting simplifies the task of connecting a database to the Web
 - Define an HTML document with embedded executable code/SQL queries.
 - Input values from HTML forms can be used directly in the embedded code/SQL queries.
 - When the document is requested, the Web server executes the embedded code/SQL queries to generate the actual HTML document.
- Numerous server-side scripting languages
 - JSP, PHP
 - General purpose scripting languages: VBScript, Perl, Python

SWAYAM: NPTEL-NOC MOC's Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.27 ©Silberschatz, Korth and Sudarshan

Now, along with the servlet there is another concept which is called server side scripting. Server side scripting is a mechanism where you define an HTML document and to within that HTML document can be used. So, you may have some inputs to that and they can be used to directly fire embedded SQL queries.

So, we talked about a madding of SQL query in while we discuss about the basic mechanism of host language and query language. So, here the HTML kind of a language is a host and you can embed the query right in as a part of that. And so, that query goes to the database query server and you get the answer and that answer is placed where your original query was there. So, that you continue to get very easily a my complete HTML as a response.

So, this kind of a mechanism is makes it very easy because a it is quite easy to conceive of the HTML and fill in. So, if I have asked for say logged in to the my mail Gmail service then I have given the input as my user name password and when that got gets authenticated. Then I get a response which is select mail from different respective tables where my authentication is there the user name is PPD and so, on. So, it becomes quite easy to actually create the HTML and there several such scripting language is JSP and PHP are the most popular ones.

(Refer Slide Time: 28:35)

Java Server Pages (JSP)

- A JSP page with embedded Java code

```
<html>
<head> <title> Hello </title> </head>
<body>
<% if (request.getParameter("name") == null)
{ out.println("Hello World"); }
else { out.println("Hello, " + request.getParameter("name")); }
%>
</body>
</html>
```
- JSP is compiled into Java + Servlets

SWAYAM: NPTEL-NOC IITK. Instructor: Prof. P. Das, IIT Kharagpur, Jan-April, 2018

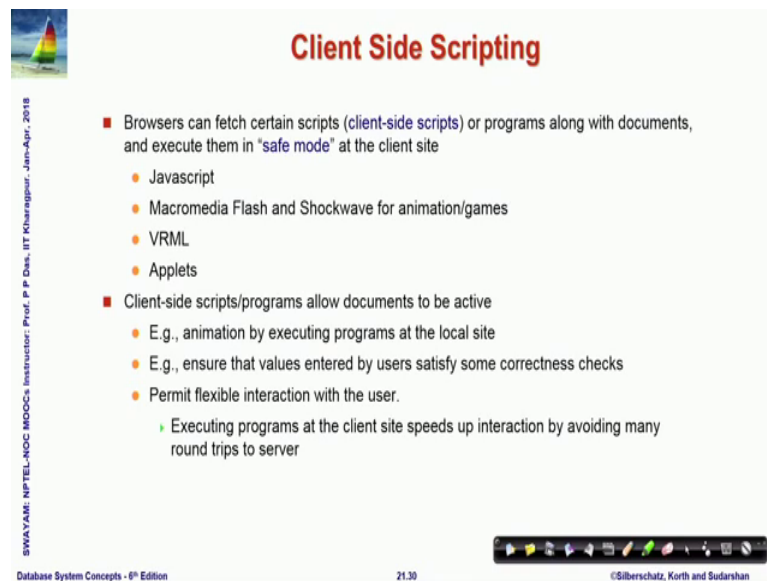
Database System Concepts - 9th Edition 21.28 ©Silberschatz, Korth and Sudarshan

So, this is how a typical JSP will look like. So, you can see that this actually looks like HTML, but inside that you have a, you have some part of a Java code. So, what will

happen the body will get replaced when the when the response has come for example, here the response is doing hello world.

So, when this is executed then whatever is a result will replace the body in the HTML here and the result in the HTML will get generated. There is another mechanism of scripting which is also very popular called PHP. So, this is how it is done.

(Refer Slide Time: 29:19)



Client Side Scripting

- Browsers can fetch certain scripts (client-side scripts) or programs along with documents, and execute them in "safe mode" at the client site
 - Javascript
 - Macromedia Flash and Shockwave for animation/games
 - VRML
 - Applets
- Client-side scripts/programs allow documents to be active
 - E.g., animation by executing programs at the local site
 - E.g., ensure that values entered by users satisfy some correctness checks
 - Permit flexible interaction with the user.
 - Executing programs at the client site speeds up interaction by avoiding many round trips to server

SWAYAM: NPTEL-NOC MOOC's Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-April, 2018

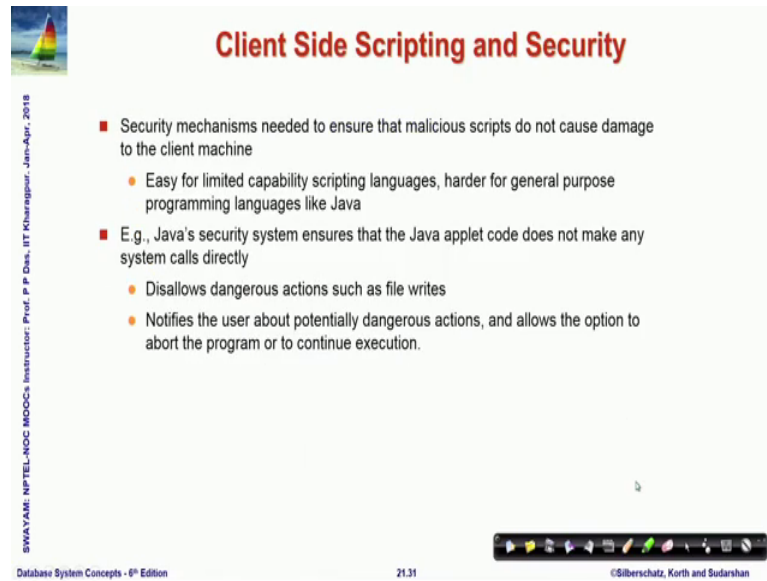
Database System Concepts - 6th Edition 21.30 ©Silberschatz, Korth and Sudarshan

Similarly, you could have script an client side also for example, if you are entering data for say a month and in numeric and you happened to enter 14; then in most cases the page will immediately give a error saying that 14 cannot be a valid month; so, there is a validation involved.

So, in the client side in the browser there are some small script that can run a most typically it is a Java script which can too different authentication which is possible without actually accessing the data in the database. You cannot for example, validate a mail data based on the client side scripting sitting on the browser, but you can validate small things like valid data forms, range of data and so, on.

And it is it is very important because if you could not do that then all you required is you would have send that faulty month numbered 14 to the to the backend server and got an error and you would have come back and then have to correct it, but you can do this locally at the browser itself.

(Refer Slide Time: 30:26)



Client Side Scripting and Security

- Security mechanisms needed to ensure that malicious scripts do not cause damage to the client machine
 - Easy for limited capability scripting languages, harder for general purpose programming languages like Java
- E.g., Java's security system ensures that the Java applet code does not make any system calls directly
 - Disallows dangerous actions such as file writes
 - Notifies the user about potentially dangerous actions, and allows the option to abort the program or to continue execution.

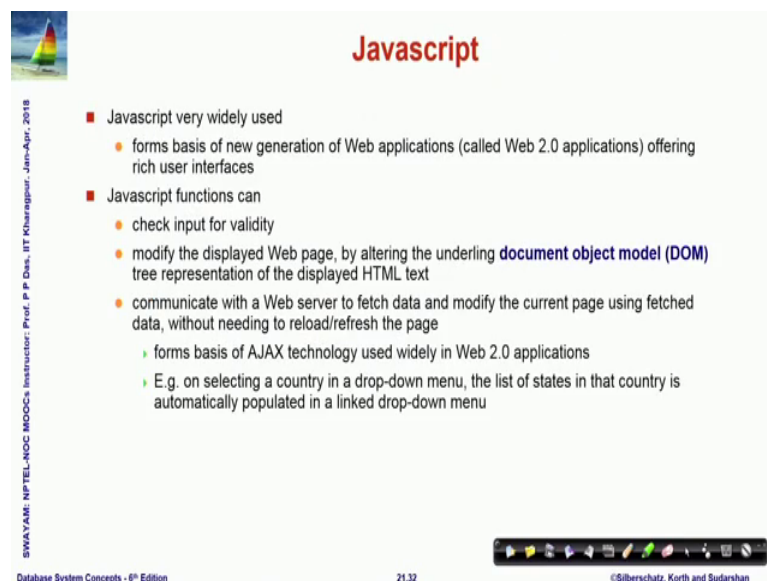
SWAYAM: NPTEL-NOC MDOCS Instructor: Prof. P. P. Das, IIT Khargapur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.31 ©Silberschatz, Korth and Sudarshan

So, client side scripting has a lot of value, but you will have to have to remember that a if you are doing client side scripting then there are security issues.

Because it is quite possible that if you are doing things on the client side that is on the browser then we might also inadvertently or by a malicious intact actually make damages to the machine on which the browser is running. So, there are different kinds of care that is to be taken for example, Java applet which is another way of doing client side computation disallows file writes and so, on.

(Refer Slide Time: 31:03)



Javascript

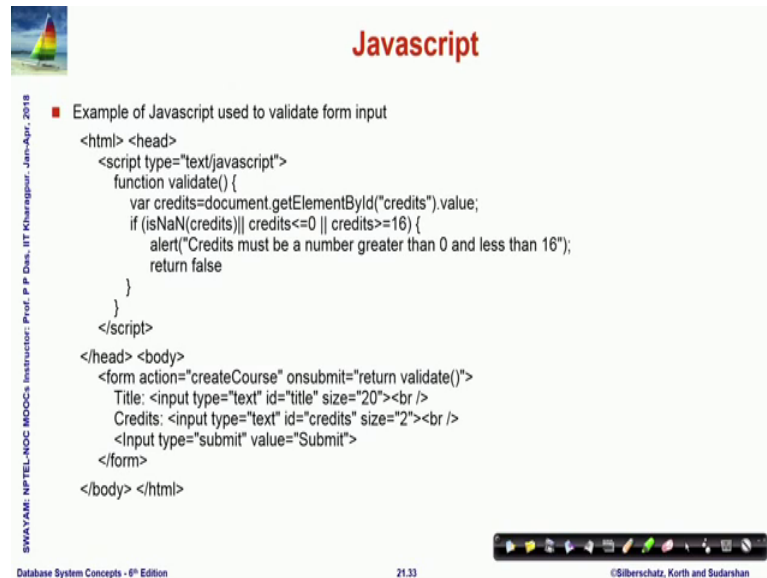
- Javascript very widely used
 - forms basis of new generation of Web applications (called Web 2.0 applications) offering rich user interfaces
- Javascript functions can
 - check input for validity
 - modify the displayed Web page, by altering the underlying **document object model (DOM)** tree representation of the displayed HTML text
 - communicate with a Web server to fetch data and modify the current page using fetched data, without needing to reload/refresh the page
 - forms basis of AJAX technology used widely in Web 2.0 applications
 - E.g. on selecting a country in a drop-down menu, the list of states in that country is automatically populated in a linked drop-down menu

SWAYAM: NPTEL-NOC MDOCS Instructor: Prof. P. P. Das, IIT Khargapur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.32 ©Silberschatz, Korth and Sudarshan

And Java script as I have said is widely used and it can function to check for input validity which I gave an example of it can modify the displayed web page, it can communicate with the web server to fetch data and so, on. And it is you should familiarize yourself with Java script more; it is a very powerful mechanism to do compute the sample things at the client side this is an example that I have given.

(Refer Slide Time: 31:20)



Javascript

- Example of Javascript used to validate form input

```

<html> <head>
<script type="text/javascript">
function validate() {
var credits=document.getElementById("credits").value;
if (isNaN(credits)|| credits<=0 || credits>=16) {
alert("Credits must be a number greater than 0 and less than 16");
return false
}
}
</script>
</head> <body>
<form action="createCourse" onsubmit="return validate()">
Title: <input type="text" id="title" size="20"><br />
Credits: <input type="text" id="credits" size="2"><br />
<input type="submit" value="Submit">
</form>
</body> </html>

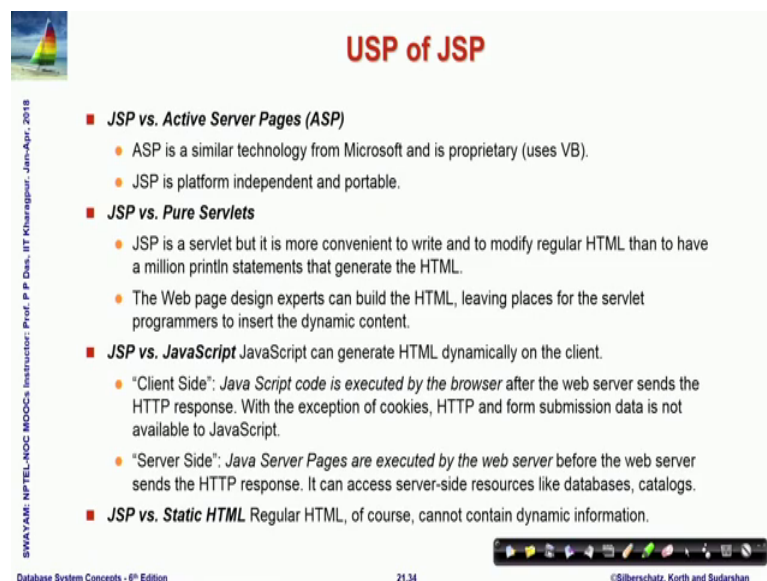
```

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.33 ©Silberschatz, Korth and Sudarshan

So, you can read through and you will be able to if you know Java you will be able to understand Java script very easily, you could get through that.

(Refer Slide Time: 31:33)



USP of JSP

- **JSP vs. Active Server Pages (ASP)**
 - ASP is a similar technology from Microsoft and is proprietary (uses VB).
 - JSP is platform independent and portable.
- **JSP vs. Pure Servlets**
 - JSP is a servlet but it is more convenient to write and to modify regular HTML than to have a million printIn statements that generate the HTML.
 - The Web page design experts can build the HTML, leaving places for the servlet programmers to insert the dynamic content.
- **JSP vs. JavaScript** JavaScript can generate HTML dynamically on the client.
 - "Client Side": *Java Script code is executed by the browser* after the web server sends the HTTP response. With the exception of cookies, HTTP and form submission data is not available to JavaScript.
 - "Server Side": *Java Server Pages are executed by the web server* before the web server sends the HTTP response. It can access server-side resources like databases, catalogs.
- **JSP vs. Static HTML** Regular HTML, of course, cannot contain dynamic information.

SWAYAM: NPTEL-NOC MOOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 6th Edition 21.34 ©Silberschatz, Korth and Sudarshan

And so, there are multiple options of doing such things, but JSP has a unique position because it is very useful in many context wise a JSP has certain USP like active server page which is used in terms of the Microsoft platform is also another mechanism of doing server side scripting, but JSP is better because it is portable in with respect to pure servlets which we showed you in the beginning JSP performs is easier to use.

Because JSP has the structure of the HTML page whereas, in a pure servlet we will have to use print line to print every tag of the HTML which is cumbersome. JSP in contrast with Java script is certainly a different thing because Java script runs on the browser on the client side, JSP runs on the server side and certainly JSP is compared to static HTML is more powerful because it can handle dynamic information.

(Refer Slide Time: 32:39)

Module Summary

- Understood the requirements of Database Application Programs and User Interfaces
- Familiarized with the Fundamentals notions and technologies of Web
- Learnt the notions of Servlets and Java Server Pages

SWAYAM: NPTEL-NOC MOCs Instructor: Prof. P. P. Das, IIT Kharagpur, Jan-Apr, 2018

Database System Concepts - 9th Edition 21.35 ©Silberschatz, Korth and Sudarshan

So, in this that brings us to the end of this current modules; so, what we have done we have understood the basic requirements of database application programs and user interfaces understood the basic terminology of the web and took a look into the core notion, core technologies of application development which is in terms of the servlets and Java server pages.