1. **Commerce and E-Business**

***Module 1 : E-Commerce Introduction & E-Commerce***

**1.1 Definition of E-Commerce**

Electronic commerce, commonly written as e-commerce, is the trading in products or services using computer networks, such as the Internet. Electronic commerce draws on technologies such as mobile commerce, electronic funds transfer, supply chain management, Internet marketing, onlinetransactionprocessing, electronicdatainterchange (EDI), inv entory management systems, and automated data collection systems. Modern electronic commerce typically uses the World Wide Web for at least one part of the transaction's life cycle, although it may also use other technologies such as e-mail

The buying and selling of products and services by businesses and consumers through an electronic medium, without using any paper documents. E-commerce is widely considered the buying and selling of products over the internet, but any transaction that is completed solely through electronic measures can be considered e-commerce. The benefits of e-commerce include its around-the-clock availability, the speed of access, a wider selection of goods and services, accessibility, and international reach. Its perceived downsides include sometimes-limited customer service, not being able to see or touch a product prior to purchase, and the necessitated wait time for product shipping.

To ensure the security, privacy and effectiveness of e-commerce, businesses should authenticate business transactions, control access to resources such as webpages for registered or selected users, encrypt communications and implement security technologies such as the Secure Sockets Layer.

**1.2 Different types of E-Commerce:**

E-Commerce or Electronics Commerce business models can generally categorized in following categories.

* **Business - to - Business (B2B)**
* **Business - to - Consumer (B2C)**
* **Consumer - to - Consumer (C2C)**
* **Consumer - to - Business (C2B)**
* **Business - to - Government (B2G)**
* **Government - to - Business (G2B)**
* **Government - to - Citizen (G2C)**
* **Business - to - Business (B2B)**

**Business-to-Business (B2B)** e-commerce encompasses all electronic transactions of goods or services conducted between companies. Producers and traditional commerce wholesalers typically operate with this type of electronic commerce.

Website following B2B business model sells its product to an intermediate buyer who then sells the product to the final customer.

As an example, a wholesaler places an order from a company's website and after receiving the consignment, sells the end product to final customer who comes to buy the product at wholesaler's retail outlet.

**Business - to - Consumer(B2C)**

The Business-to-Consumer type of e-commerce is distinguished by the establishment of electronic business relationships between businesses and final consumers. It corresponds to the retail section of e-commerce, where traditional retail trade normally operates.

Website following B2C business model sells its product directly to a customer. A customer can view products shown on the website of business organization. The customer can choose a product and order the same. Website will send a notification to the business organization via email and organization will dispatch the product/goods to the customer.

This type of commerce has developed greatly, due to the advent of the web, and there are already many virtual stores and malls on the Internet, which sell all kinds of consumer goods, such as computers, software, books, shoes, cars, food, financial products, digital publications, etc.

**Consumer - to - Consumer (C2C)**

Website following C2C business model helps consumer to sell their assets like residential property, cars, motorcycles etc. or rent a room by publishing their information on the website. Website may or may not charge the consumer for its services. Another consumer may opt to buy the product of the first customer by viewing the post/advertisement on the website.

Consumer-to-Consumer (C2C) type e-commerce encompasses all electronic transactions of goods or services conducted between consumers. Generally, these transactions are conducted through a third party, which provides the online platform where the transactions are actually carried out.

**Consumer - to - Business (C2B)**

In this model, a consumer approaches website showing multiple business organizations for a particular service. Consumer places an estimate of amount he/she wants to spend for a particular service. For example, comparison of interest rates of personal loan/ car loan provided by various banks via website. Business organization who fulfills the consumer's requirement within specified budget approaches the customer and provides its services.

Examples of such practices are the sites where designers present several proposals for a company logo and where only one of them is selected and effectively purchased. Another platform that is very common in this type of commerce are the markets that sell royalty-free photographs, images, media and design elements,

In C2B there is a complete reversal of the traditional sense of exchanging goods. This type of e-commerce is very common in crowdsourcing based projects. A large number of individuals make their services or products available for purchase for companies seeking precisely these types of services or products.

**Business - to - Government (B2G)**

B2G model is a variant of B2B model. Such websites are used by government to trade and exchange information with various business organizations. Such websites are accredited by the government and provide a medium to businesses to submit application forms to the government.

On the Internet, B2G is business-to-government (a variation of the term B2B or business-to-business), the concept that businesses and government agencies can use central Web sites to exchange information and do business with each other more efficiently than they usually can off the Web. For example, a Web site offering B2G services could provide businesses with a single place to locate applications and tax forms for one or more levels of government (city, state or province, country, and so forth); provide the ability to send in filled-out forms and payments; update corporate information; request answers to specific questions; and so forth.

B2G may also includee-procurement services, in which businesses learn about the purchasing needs of agencies and agencies request proposal responses.

B2G may also support the idea of a virtual workplace in which a business and an agency could coordinate the work on a contracted project by sharing a common site to coordinate online meetings, review plans, and manage progress.

B2G may also include the rental of online applications and databases designed especially for use by government agencies.

**Government - to - Business (G2B)**

Government uses B2G model website to approach business organizations. Such websites support auctions, tenders and application submission functionalities.

**Government - to - Citizen (G2C)**

Government uses G2C model website to approach citizen in general. Such websites support auctions of vehicles, machinery or any other material. Such website also provides services like registration for birth, marriage or death certificates. Main objectives of G2C website are to reduce average time for fulfilling people requests for various government services.

This Model is also a part of e-governance.

The objective of this model is to provide good and effective services to each citizen.

The Government provides the following facilities to the citizens through website.

Information of all government departments,

**1.3 Examples of E-Commerce Trade Cycle**

**The E-Commerce Trade Cycle**

A trade cycle is the series of exchanges, between a customer and supplier, that take place when a commercial exchange is executed. A general trade cycle consists of:

Pre-Sales: Finding a supplier and agreeing the terms.

Execution: Selecting goods and taking delivery.

Settlement: Invoice (if any) and payment.

After-Sales: Following up complaints or providing maintenance.

As with physical commerce, the trade cycle is more or less categorised into four areas.

First is the identification of a supplier by a consumer, and the agreement of terms for supply.

The second stage is selecting the goods and taking delivery.

The traditional third stage, payment and invoicing, is usually executed along with the selection of goods, in e-commerce.

The final stage is after-sales, in which complaints are addressed, support given, and any further involvement of the supplier with the product as specified in the terms of supply is engaged.

Example:

As an example of the trade cycle, a consumer will go to Amazon to find a specific product. This is the identification of a supplier. Once the product is added to the shopping basket, the consumer will check out using Amazon‘s specific e-commerce software, and pay for it while specifying an address at which to take delivery, combining the second and third phases. Once the product arrives, and if it is unsatisfactory, the customer returns it to Amazon, who then will provide a refund, a replacement or another service, thereby fulfilling the fourth phase of the e-commerce trade cycle as agreed through that country‘s laws and Amazon‘s own terms and conditions.

**EDI TRADE CYCLE**

Electronic Data Interchange (EDI)

EDI is a product of the two most rapidly advancing technologies in modern times, namely computing and telecommunications.

The convergence of these two technologies has made it possible for a structured string of data to be exchanged between business applications without human intervention.

EDI revolutionizes business communications by removing a complete layer in business practices - the use and processing of paper documents. The rationalization of data flows within a company enhances the integration of business functions and hence facilitates the decision making process.

EDI opens up potent strategies such as "just in time" manufacturing. In addition, it enables companies to forge closer and more effective links with their trading partners.

Paperless trading is growing fast in many countries, in particular because "just in time" stock control usually means more, smaller shipments with very tight delivery schedules that paper documents cannot cope with, and also because EDI is a natural evolution in the international trade cycle. Indeed, one of the principal reasons for using

EDI is the mountain of paper documents produced, moved, handled, corrected, transcribed and copied in normal business transactions.

EDI has none of the disadvantages of paper documents and brings substantial benefits and savings to companies which implement it, such as accuracy (data are received directly from computer files and are not re-entered manually), speed (data are processed by computer without manual intervention and are transmitted quicker than information sent by post or courier and re-entered manually) and savings (it saves on the cost of mailing, copying, filing, distributing and capturing data).

**ADVANTAGES & DISADVANTAGES OF E-Commerce:**

**Advantages:**

**Benefits to Organizations:**

Electronic commerce expands the marketplace to national and international markets. With minimal capital outlay, a company can easily and quickly locate more customers, the best suppliers, and the most suitable business partners worldwide.

Electronic commerce decreases the cost of creating, processing, distributing, storing, and retrieving paper-based information.

Ability for creating highly specialized businesses. For example, dog toys which can be purchased only in pet shops or department and discounte stores in the physical world, are sold now in a specialized www.dogtoys.com (also see www.cattoys.com).

Electronic commerce reduces the time between the outlay of capital and the receipt of products and services.

Electronic commerce initiates business processes reengineering projects. By changing processes, productivity of salespeople, knowledge workers, and administrators can increase by 100 percent or more.

Electronic commerce lowers telecommunications cost-the Internet is much cheaper than VANs. · Other benefits include improved image, improved customer service, newfound business partners, simplified processes, compressed cycle and delivery time, increased productivity, eliminating paper, expediting access to information, reduced transportation costs, and increased flexibility.

**Benefits to Consumers :**

The benefits of EC to consumers are as follows:

Electronic commerce enables customers to shop or do other transactions 24 hours a day, all year round, from almost any location. ·

Electronic commerce provides customers with more choices; they can select Electronic commerce frequently provides customers with less expensive products and services by allowing them to shop in many places and conduct quick comparisons.

In some cases, especially with digitized products, EC allows quick delivery. ·Customers can receive relevant and detailed information in seconds, rather than days or weeks.

Electronic commerce makes it possible to participate in virtual auctions.

Electronic commerce allows customers to interact with other customers in electronic communities and exchange ideas as well as compare experiences.

Electronic commerce facilitates competition, which results in substantial discounts.

**Benefits to Society**

The benefits of EC to society are as follows: ·

Electronic commerce enables more individuals to work at home and to do less traveling for shopping, resulting in less traffic on the roads and lower air pollution.

Electronic commerce allows some merchandise to be sold at lower prices, so less affluent people can buy more and increase their standard of living.

Electronic commerce enables people in Third World countries and rural areas to enjoy products and services that otherwise are not available to them. This includes opportunities to learn professions and earn college degrees.

Electronic commerce facilitates delivery of public services, such as health care, education, and distribution of government social services at a reduced cost and/or improved quality. Health-care services, for example, can reach patients in rural areas.

**DISADVANTAGS OF E-COMMERCE:**

The Limitations of EC:

The limitations of EC can be grouped into technical and nontechnical categories.

**Technical Limitations of EC :**

The technical limitations of EC are as follows:

There is a lack of system security, reliability, standards, and some communication protocols.

There is insufficient telecommunication bandwidth.

The software development tools are still evolving and changing rapidly.

It is difficult to integrate the Internet and EC software with some existing applications and databases.

Vendors may need special Web servers and other infrastructures, in addition to the network servers.

Some EC software might not fit with some hardware, or may be incompatible with some operating systems or other components.

As time passes, these limitations will lessen or be overcome; appropriate planning can minimize their impact.

**NonTechnical Limitations:**

Of the many nontechnical limitations that slow the spread of EC, the following are the major ones.

**Cost and justification:**

The cost of developing EC in-house can be very high, and mistakes due to lack of experience may result in delays.

There are many opportunities for outsourcing, but where and how to do it is not a simple issue.

Furthermore, to justify the system one must deal with some intangible benefits (such as improved customer service and the value of advertisement), which are difficult to quantify.

**Security and privacy :**

These issues are especially important in the B2C area, especially security issues which are perceived to be more serious than they really are when appropriate encryption is used. Privacy measures are constantly improved.

Yet, the customers perceive these issues as very important, and, the EC industry has a very long and difficult task of convincing customers that online transactions and privacy are, in fact, very secure.

**Lack of trust and user resistance :**

Customers do not trust an unknown faceless seller (sometimes they do not trusteven known ones), paperless transactions, and electronic money. So switching from physical to virtual stores may be difficult.

**Other limiting factors:**

Lack of touch and feel online. Some customers like to touch items such as clothes and like to know exactly what they are buying.

Many legal issues are as yet unresolved, and government regulations and standards are not refined enough for many circumstances.

Electronic commerce, as a discipline, is still evolving and changing rapidly. Many people are looking for a stable area before they enter into it.

There are not enough support services. For example, copyright clearance centers for EC transactions do not exist, and high-quality evaluators, or qualified EC tax experts, are rare.

In most applications there are not yet enough sellers and buyers for profitable EC operations.

Electronic commerce could result in a breakdown of human relationships.

Accessibility to the Internet is still expensive and/or inconvenient for many potential customers..)

Despite these limitations, rapid progress in EC is taking place. For example, the number of people in the United States who buy and sell stocks electronically increased from 300,000 at the beginning of 1996 to about 10 million in fall 1999.

As experience accumulates and technology improves, the ratio of EC benefits to costs will increase, resulting in a greater rate of EC adoption. The potential benefits may not be convincing enough reasons to start EC activities

***MODULE 2 : OVERVIEW OF H/W & S/W TECHNOLOGIES FOR E-COMMERCE***

**2.1:CLIENT SIDE PROGRAMMING(DREAM WEAVER,FRONT PAGE):**

**CLIENT SIDE SCRIPTING:**

Client-side scripting generally refers to the class of computer programs on the web that are executed client-side, by the user's web browser, instead of server-side (on the web server).

This type of computer programming is an important part of the Dynamic HTML (DHTML) concept, enabling web pages to be scripted; that is, to have different and changing content depending on user input, environmental conditions (such as the time of day), or other variables.

Client-side scripts are often embedded within an HTML or XHTML document (hence known as an "embedded script"), but they may also be contained in a separate file, to which the document (or documents) that use it make reference (hence known as an "external script"). Upon request, the necessary files are sent to the user's computer by the web server (or servers) on which they reside. The user's web browser executes the script, then displays the document, including any visible output from the script.

Client-side scripts may also contain instructions for the browser to follow in response to certain user actions, (e.g., clicking a button). Often, these instructions can be followed without further communication with the server.

By viewing the file that contains the script, users may be able to see its source code. Many web authors learn how to write client-side scripts partly by examining the source code for other authors' scripts.

**2.2:SERVER SIDE PROGRAMMING:**

**SERVER SIDE SCRIPTING**

Server-side scripting is a technique used in web development which involves employing scripts on a web server which produce a response customized for each user's (client's) request to the website. The alternative is for the web server itself to deliver a static web page.

Scripts can be written in any of a number of server-side scripting languages that are available (see below). Server-side scripting is distinguished from client-side scripting where embedded scripts, such as JavaScript, are run client-side in a web browser, but both techniques are often used together.

Server-side scripting is often used to provide a customized interface for the user.

These scripts may assemble client characteristics for use in customizing the response based on those characteristics, the user's requirements, access rights, etc.

Server-side scripting also enables the website owner to hide the source code that generates the interface, whereas with client-side scripting, the user has access to all the code received by the client.

A down-side to the use of server-side scripting is that the client needs to make further requests over the network to the server in order to show new information to the user via the web browser. These requests can slow down the experience for the user, place more load on the server, and prevent use of the application when the user is disconnected from the server.

For Detailed understanding and clearing your concepts read below:

The Server - This party is responsible for serving pages.

The Client - This party requests pages from the Server, and displays them to the user. In most cases, the client is a web browser.

The User - The user uses the Client in order to surf the web, fill in forms, watch videos online, etc.

Each side's programming, refers to code which runs at the specific machine, the server's or the client's.

Basic Example

1.The User opens his web browser (the Client).

2.The User browses to http://google.com.

3.The Client (on the behalf of the User), sends a request to http://google.com (the Server), for their home page.

4.The Server then acknowledges the request, and replies the client with some meta-data (calledheaders), followed by the page's source.

5.The Client then receives the page's source, and renders it into a human viewable website.

6.The User types Stack Overflow into the search bar, and presses Enter

7.The Client submits that data to the Server.

8.The Server processes that data, and replies with a page matching the search results.

9.The Client, once again, renders that page for the User to view.

**Server-side Programming**

Server-side programming, is the general name for the kinds of programs which are run on the Server.

Uses

Process user input.

Display pages.

Structure web applications.

Interact with permanent storage (SQL, files).

Example Languages

PHP

ASP.Net in C#, C++, or Visual Basic.

Nearly any language (C++, C#, Java). These were not designed specifically for the task, but are now often used for application-level web services.

**Client-side programming**

Much like the server-side, Client-side programming is the name for all of the programs which are run on the Client.

Uses

Make interactive webpages.

Make stuff happen dynamically on the web page.

Interact with temporary storage, and local storage (Cookies, localStorage).

Send requests to the server, and retrieve data from it.

Provide a remote service for client-side applications, such as software registration, content delivery, or remote multi-player gaming.

Example languages

JavaScript (primarily)

HTML\*

CSS\*

Any language running on a client device that interacts with a remote service is a client-side language.

Client-Side vs. Server-Side Code: What’s the Difference?

The majority of the processing will be done at the server and not on the client‘s internet browser.

When a database needs to be accessed on a server, the web application will post the page back to the web server and server-side code will process the request.

In order to fully understand the web application‘s architecture, we must understand postback and the page life cycle, which I have discussed more extensively in previous articles.

Server-Side Code

There are several server-side technologies that can be used when developing web applications. The most popular is Microsoft‘s ASP.NET. In ASP.NET, server-side code uses the .NET Framework and is written in languages like C# and VB.NET. Server-side processing is used to interact with permanent storage like databases or files. The server will also render pages to the client and process user input. Server-side processing happens when a page is first requested and when pages are posted back to the server. Examples of server-side processing are user validation, saving and retrieving data, and navigating to other pages.

The disadvantage of server-side processing is the page postback: it can introduce processing overhead that can decrease performance and force the user to wait for the page to be processed and recreated. Once the page is posted back to the server, the client must wait for the server to process the request and send the page back to the client.

Client-Side Code

The benefits of client-side processing in an ASP.NET web application are programming languages like C# and VB.NET along with the .NET Framework. Languages like C# and VB.NET sit on top of the .NET framework and have all the benefits of object oriented architectures like inheritance, implementing interfaces and polymorphism.

In contrast to server-side code, client-side scripts are embedded on the client‘s web page and processed on the client‘s internet browser. Client-side

scripts are written in some type of scripting language like JavaScript and interact directly with the page‘s HTML elements like text boxes, buttons, list-boxes and tables. HTML and CSS (cascading style sheets) are also used in the client. In order for client-side code to work, the client‘s internet browser must support these languages.

There are many advantages to client-side scripting including faster response times, a more interactive application, and less overhead on the web server. Client-side code is ideal for when the page elements need to be changed without the need to contact the database. A good example would be to dynamically show and hide elements based on user inputs. One of the most common examples is input validation and Microsoft‘s Visual Studio includes a set of client-side validation controls.

**Ajax (Asynchronous JavaScript and XML)**

The general rule is to use server-side processing and page postbacks when the client needs to interact with server-side objects like databases, files, etc. However, the concept of Ajax has changed the rules quite a bit. Ajax is the concept of the client calling the server directly to interact with server objects like a database, without a postback involved.

Ajax is a concept that involves a group of existing technologies such as server-side data, web services and client-side scripting. The client-side scripts will call a web service and the web service processes the database request. The request could be the retrieve and/or save data. Ajax calls are asynchronous, meaning that once the client makes an Ajax call to the web service, the client is not locked and waiting for a response. The web service will send a response back to the client when their task has completed. The client will intercept the response and process the response accordingly.

A significant breakthrough in client-side scripting is jQuery. To quote, jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.

Starting with Visual Studio 2005, Microsoft has offered their Ajax Control Toolkit. This a set of ASP.NET controls that have plenty of built-in client-side processing. With Visual Studio 2008, they‘ve offered AJAX-enabled WCF Services. These web services are streamlined for asynchronous Ajax callbacks and require little client-side scripting.

In conclusion, the amount of the client-side scripting used in web applications will continue to increase as its power, flexibility and simplicity continue to increase.

**2.3:DATABASE CONNECTIVITY:**

A database connection is the means by which a [database server](https://en.wikipedia.org/wiki/Database_server) and its [client](https://en.wikipedia.org/wiki/Client_(computing)) software communicate with each other. The term is used whether or not the client and the server are on different machines.

The client uses a database connection to send [commands](https://en.wikipedia.org/wiki/Command_(computing)) to and receive replies from the server.

A database is stored as a file or a set of files on magnetic disk or tape, optical disk, or some other secondary storage device. The information in these files may be broken down into records, each of which consists of one or more fields.

Fields are the basic units of data storage, and each field typically contains information pertaining to one aspect or attribute of the entity described by the database. Records are also organized into tables that include information about relationships between its various fields. Although database is applied loosely to any collection of information in computer files, a database in the strict sense provides cross-referencing capabilities.

Connections are a key concept in [data-centric](https://en.wikipedia.org/w/index.php?title=Data-centric&action=edit&redlink=1) programming. Since some [DBMSs](https://en.wikipedia.org/wiki/Database) require considerable time to connect, [connection pooling](https://en.wikipedia.org/wiki/Connection_pooling) is used to improve performance. No command can be performed against a database without an "open and available" connection to it.

Connections are built by supplying an underlying driver or provider with a connection

string, which is used to address a specific database or server and to provide instance and user authentication credentials (for example, Server=sql\_box;Database=Common;User

ID=uid;Pwd=password; ).

Once a connection has been built, it can be opened and closed at will, and properties (such as the command time-out length, or transaction, if one exists) can be set. The connection string consists of a set of key-value pairs, dictated by the data access interface of the data provider.

**Types Of Database Connectivity:**

**What is ODBC?**

ODBC is (Open Database Connectivity):

A standard or open application programming interface (API) for accessing a database.

By using ODBC statements in a program, you can access files in a number of different databases, including Access, dBase, DB2, Excel, and Text. It allows programs to use SQL requests that will access databases without having to know the proprietary interfaces to the databases.

ODBC handles the SQL request and converts it into a request the individual database system understands.

More on ODBC You need: the ODBC software, and a separate module or driver for each database to be accessed. Library that is dynamically connected to the application. Driver masks the heterogeneity of DBMS operating system and network protocol. E.g. (Sybase, Windows/NT, Novell driver)

**What is JDBC?**

JDBC is: Java Database Connectivity is a Java API for connecting programs written in Java to the data in relational databases.

Consists of a set of classes and interfaces written in the Java programming language. provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API. The standard defined by Sun Microsystems, allowing individual providers to implement and extend the standard with their own JDBC drivers.

JDBC: establishes a connection with a database sends SQL statements processes the results.

**JDBC vs ODBC**

ODBC is used between applications JDBC is used by Java programmers to connect to databases .

With a small "bridge" program, you can use the JDBC interface to access ODBCaccessible databases.

JDBC allows SQL-based database access for EJB persistence and for direct manipulation from CORBA, DJB or other server objects

**JDBC API :**

he JDBC API supports both two-tier and three-tier models for database access.

Two-tier model -- a Java applet or application interacts directly with the database.

Three-tier model -- introduces a middle-level server for execution of business logic: the middle tier to maintain control over data access. The user can employ an easy-to-use higher-level API which is translated by the middle tier into the appropriate low-level calls.

**The JDBC Steps:**

1. Importing Packages
2. Registering the JDBC Drivers
3. Opening a Connection to a Database
4. Creating a Statement Object
5. Executing a Query and Returning a Result Set Object
6. Processing the Result Set
7. Closing the Result Set and Statement Objects
8. Closing the Connection

**1: Importing Packages**

Program name: LecExample\_1a.java

Purpose: Basic selection using prepared statement

Import packages

import java.sql.\*; //JDBC packages import java.math.\*;

import java.io.\*;

import oracle.jdbc.driver.\*;

**2: Registering JDBC Drivers**

class LecExample\_1a {

public static void main (String args [])

throws SQLException {

Load Oracle driver DriverManager.registerDriver (new oracle.jdbc.driver.OracleDriver());

**3: Opening connection to a Database**

//Prompt user for username and password

String user;

String password;

user = readEntry("username: ");

password = readEntry("password: ");

Connect to the local database Connection conn = DriverManager.getConnection ("jdbc:oracle:thin:@aardvark:1526:teach ", user, password);

**4. Creating a Statement Object**

// Query the hotels table for resort =

'palma nova‘

Please notice the essential trim PreparedStatement pstmt = conn.prepareStatement ("SELECT hotelname, rating FROM hotels WHERE trim(resort) = ?"); pstmt.setString(1, "palma nova");

**5. Executing a Query**

Returning a Result Set Object &

**6. Processing the Result Set**

ResultSet rset = pstmt.executeQuery ();

Print query results while (rset.next ()) System.out.println (rset.getString

s(1)+" "+ rset.getString(2));

**7. Closing the Result Set and Statement Objects**

**8. Closing the Connection**

close the result set, statement, and the connection

rset.close();

pstmt.close();

conn.close();

}

**What is SQLJ?**

SQLJ is a set of programming extensions that allow a programmer using the Java programming language to embed statements that provide SQL database requests.

SQLJ is similar to existing extensions for SQL that are provided for C, FORTRAN, and other programming languages.

IBM, Oracle, and several other companies are proposed SQLJ as a standard and as a simpler and easier-to-use alternative to JDBC.

SQLJ Example:

#sql { … } ;

SQL can span multiple lines

Java host expressions in SQL statement

throws java.sql.SQLException

String bug = ―spider‖;

#sql {

INSERT INTO bugs (name, numLegs)

VALUES (:bug, :(getNumLegs(bug)))

};

**2.4:SESSION TRACKING:**

HTTP is a "stateless" protocol which means each time a client retrieves a Web page, the client opens a separate connection to the Web server and the server automatically does not keep any record of previous client request.

Still there are following three ways to maintain session between web client and web server:

Cookies:

A webserver can assign a unique session ID as a cookie to each web client and for subsequent requests from the client they can be recognized using the recieved cookie.

This may not be an effective way because many time browser does not support a cookie, so I would not recommend to use this procedure to maintain the sessions.

Hidden Form Fields:

A web server can send a hidden HTML form field along with a unique session ID as follows:

<input type="hidden" name="sessionid" value="12345">

This entry means that, when the form is submitted, the specified name and value are automatically included in the GET or POST data. Each time when web browser sends request back, then session\_id value can be used to keep the track of different web browsers.

This could be an effective way of keeping track of the session but clicking on a regular (<A HREF...>) hypertext link does not result in a form submission, so hidden form fields also cannot support general session tracking.

URL Rewriting:

You can append some extra data on the end of each URL that identifies the session, and the server can associate that session identifier with data it has stored about that session.

For example, with http://tutorialspoint.com/file.htm;sessionid=12345, the session identifier is attached as sessionid=12345 which can be accessed at the web server to identify the client.

URL rewriting is a better way to maintain sessions and works for the browsers when they don't support cookies but here drawback is that you would have generate every URL dynamically to assign a session ID though page is simple static HTML page.

The HttpSession Object:

Apart from the above mentioned three ways, servlet provides HttpSession Interface which provides a way to identify a user across more than one page request or visit to a Web site and to store information about that user.

The servlet container uses this interface to create a session between an HTTP client and an HTTP server. The session persists for a specified time period, across more than one connection or page request from the user.

You would get HttpSession object by calling the public method getSession() of HttpServletRequest, as below:

HttpSession session = request.getSession();

You need to call request.getSession() before you send any document content to the client.

**2.5:MIDDLEWARE TECHNOLOGIES FROM E\_COMMERCE PERSPECTIVE**

Evolution of Internet-based computing from local area networks (LANs), after transitioning from unconnected computers to networks, is the hallmark of all business models today. The technological backbone of this evolution is the middleware.

First connecting, then communicating, and finally seamlessly integrating the distributed systems to external sites, customers, suppliers, and trading partners across the world is the real challenge for the business world.

Also required is the talking between client and server over heterogeneous networks, systems architectures,databases, and other operating environment.

All this is facilitated by the middleware technologies that offer undercover functions to seamlessly integrate various applications with information instantly to make it accessible across diverse architectures, protocols,and networks.

Automation of back-end and front-end operations of business is also effected by the middleware. Middleware binds discrete applications, such as Web-based applications and older mainframe-based systems, to allowcompanies to hook up with latest systems anddevelopments that drive new applications without making their in-vestments in legacy systems unyielding

**Functions of Middleware**

Middleware functions are generally classified into:

Application-specific functions to deliver services for different classes of applications such as distributed-database services, distributed-data/object-transaction processing, and specialized services for mobile computing and multimedia.

Information-exchange functions to manage the flow of information across a network—for tasks like transferring data, issuing commands, receiving responses, checking status, and resolving standoffs.

Management and support functions to locate resources, communicate with servers, handle security and failures, and monitor performance.

**Major Types of Middleware**

The selection of middleware technology is determined by what information is required to be communicated, for example, database middleware will be the choice if database is the main requirement.

However, following are the major categories of middleware:

Database Middleware,

Remote Procedure Calls (RPC)

Object Request Broker (ORB),

Application Server Middleware,

Message Oriented Middleware (MOM),

Transaction Processing Monitor (TP),

**Database Middleware:**

The most widely used, easy to install, and relatively economical middleware, Database middleware,is usually chosen to complement other types of middleware and facilitates communication among applications and local or remote databases but cannot transfer calls or objects.

However,database middleware does not allow the two-way communications between servers and clients.

SQL type command is generally subjected to the middleware gateway, which would convey the command to the end database to collect and send the reply of the SQL query back.

Synchronous point-to-point type of communications is the characteristic of database middleware and can pose problems when multiple demands from multiple users produce huge traffic and congestion.

Database middleware is the most mature middleware technology.

**Remote Procedure Calls (RPC)**

Remote Procedure Calls (RPC) permits a client program to call procedures located on a remote server program.

Remote procedure calls is not isolated as distinct middleware level and is entrenched

into the application with calls embedded into the client portion of the client/server application program.

Stubs are developed for both the client and the server to call up synchronously when the client makes a call to the server.

The intricacies of distributed processing are reduced by remote procedure calls by maintaining the semantics of a remote call no matter the client and server are located on the same system or not.

The synchronous nature of the remote procedurecalls makes it most appropriate for smaller applications where all communications are one-to-one and not asynchronous.

**Object Request Broker (ORB):**

Object Request Brokers (ORB) are language-independent, object-oriented, synchronous remoteprocedure calls in which an affiliate function of an object can be brought into play remotely by means of the same essential notation.

Asynchronous communication suitble to large applications can be made possible by extending the main standards as in CORBA and DCOM, the main competing standards.

ORB technologies are based on the reliability of the transport layer, which is required for the functioning.

The application programmer is secured from the details of the client/server approach by using IDL interfaces that allows the application code to call a remote object,as if it were locally supported.

Thus, the maintainability is improved as the object communication details are concealed from the application and isolated on the ORB.

Hence, ORB-based middleware applications are becoming standard for the multi-tier model.

**Message Oriented Middleware:**

Message Oriented Middleware (MOM) or enterprise message technology (EMT), provides asynchronous message delivery.

The messages are lined up, just as objects, permitting the application that sends messages, to carry out other tasks without getting blocked till it receives the response.

Generally located at a higher level than that of remote procedure calls, MOM assembly provides Middleware Components for E-commerce Infrastructure more than simply passing information. MOM also offers provisions for translating data, security,broadcasting data to multiple program, error recovery, and prioritization of messages and requests.

MOM enhances flexibility by allowing applications to switch messages without the requirement of knowing on which platform or processor the other application located.

**Transaction Processing Monitor:**

Transaction Processing Monitor (TP) is over 25 years old technology that controls interactions between a requesting client and databases. It is a database independent technology.

TP provides a three-tier client/server model and ensures an appropriate updating of the databases. This technology facilitates and controls the transport of data between numerous terminals and the application programs serving them.

It can provide services to thousands of clients in distributed client/server environment by multiplexing client transaction requests by type on to a controlled number of processing routines that support particular services.

**2.6 SECURITY ASPECTS w.r.t E-COMMERCE:**

E‐commerce is defined as the buying and selling of products or services over electronic systems. A wide variety of commerce is conducted via e‐commerce, including electronic funds transfer, supply chain management, online transaction processing, electronic data interchange (EDI) and automated data collection systems.

Any secure e‐commerce system must meet four integral requirements:

privacy – information exchanged must be kept from unauthorized parties, o integrity – the exchanged information must not be altered or tampered with,

authentication – both sender and recipient must prove their identities to each other and

non‐repudiation – proof is required that the exchanged information was indeed received.

**Privacy**

Privacy has become a major concern for consumers with the rise of identity theft and impersonation, and any concern for consumers must be treated as a major concern for eCommerce providers

Privacy now forms an integral part of any e-commerce strategy and investment in privacy protection has been shown to increase consumer‘s spend, trustworthiness and loyalty.

We can see that privacy is of major concern to users and in the event of their privacy being compromised users become very agitated and there is an overall negative effect on trust in e-commerce.

**Integrity, Authentication & NonRepudiation**

In any e-commence system the factors of data integrity, customer & client authentication and non-repudiation are critical to the success of any online business.

Data integrity is the assurance that data transmitted is consistent and correct, that is,it has not been tampered or altered in any way during transmission.

Authentication is a means by which both parties in an online transaction can be confident that they are who they say they are and non-repudiation is the idea that no party can dispute that an actual event online took place.

Proof of data integrity is typically the easiest of these factors to successfully accomplish.

A data hash or checksum, such as MD5 or CRC, is usually sufficient to establish that the likelihood of data being undetectably changed is extremely low

Not withstanding these security measures, it is still possible to compromise data in transit through techniques such as phishing or manin-the-middle attacks . These flaws have led to the need for the development of strong verification and security measurements such as digital signitures and public key infrastructures (PKI).

One of the key developments in e-commerce security and one which has led to the widespread growth of e-commerce is the introduction of digital signatures as a means of verification of data integrity and authentication.

In order for a digital signature to attain the same legal status as an ink-on-paper signature, asymmetric key cryptology must have been employed in its production

Such a system employs double keys; one key is used to encrypt the message by the sender, and a different, albeit mathematically related, key is used by the recipient to decrypt the message.

This is a very goodsystem for electronic transactions, since two stranger-parties, perhaps living far apart, can confirm each other‘s identity and thereby reduce the likelihood of fraud in the transaction.

Non-repudiation techniques prevent the sender of a message from subsequently denying that they sent the message. Digital Signatures using public-key cryptography and hash functions are the generally accepted means of providing nonrepudiation communications

**Technical Attacks**

Technical attacks are one of the most challenging types of security compromise an e-commerce provider must face.

Perpetrators of technical attacks, and in particular Denial-of-Service attacks, typically target sites or services hosted on high-profile webservers such as banks, credit card payment gateways, large online retailers and popular social networking sites.

**Denial of Service Attacks:**

Denial of Service (DoS) attacks consist of overwhelming a server, a network or a website in order to paralyze its normal activity .

Defending against DoS attacks is one of the most challenging security problems on the Internet today.

A major difficulty in thwarting these attacks is to trace the source of the attack, as they often use incorrect or spoofed IP source addresses to disguise the true origin of the attack

**NonTechnical Attacks**

Phishing Attacks:

Phishing is the criminally fraudulent process of attempting to acquire sensitive information such as usernames, passwords and credit card details, by masquerading as a trustworthy entity in an electronic communication.

Phishing scams generally are carried out by emailing the victim with a ‗fraudulent‘ email from what purports to be a legitimate organization requesting sensitive information.

When the victim follows the link embedded within the email they are brought to an elaborate and sophisticated duplicate of the legitimate organizations website.

Phishing attacks generally target bank customers, online auction sites (such as eBay), online retailers (such as amazon) and services providers (such as PayPal).

According to community banker (Swann, 2008), in more recent times cybercriminals have got more sophisticated in the timing of their attacks with them posing as charities in times of natural disaster.

Social Engineering:

Social engineering is the art of manipulating people into performing actions or divulging confidential information.

Social engineering techniques include pretexting (where the fraudster creates an invented scenario to get the victim to divulge information), Interactive voice recording (IVR) or phone phishing (where the fraudster gets the victim to divulge sensitive information over the phone) and baiting with

Trojans horses (where the fraudster ‗baits‘ the victim to load malware unto a system).

lSocial engineering has become a serious threat to e-commerce security since it is difficult to detect and to combat as it involves ‗human‘ factors which cannot be patched akin to hardware or software, albeit staff training and education can somewhat thwart the attack