UNIT 3
Scripting

Introduction to Java script

- JavaScript is a dynamic programmin g language.It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed.

- document.getElementById("demo").innerHTML = "Hello JavaScript";
- This example uses the method to "find" an HTML element (with id="demo"**)**, and changes the element content (**innerHTML**) to "Hello JavaScript“

- JavaScript can be placed in the <body> and the <head> sections of an HTML page.

documents

The JavaScript Document object is the container for all HTML HEAD and BODY objects associated within the HTML tags of an HTML document.

Document Object Properties

- alinkColor - The color of active links.
- bgColor –
- cookie - Used to identify the value of a cookie.
- defaultCharset
- domain - The domain name of the document server.
- embeds - An array containing all the plugins in a document.
- fgColor - The text color attribute set in the <body> tag.
- FileCreatedDate - Use this value to show when the loaded HTML file was created
- fileModifiedDate - Use this value to show the last change date of the loaded HTML file
- fileSize
- fileUpdatedDate
- lastModified - The date the file was modified last.
- layers - An array containing all the layers in a document.
- linkColor - The color of HTML links in the document. It is specified in the <body> tag.
- location
- protocol
- readyState
- referrer - The Universal Resource Locator (URL) of the document that we got the link to the present document from.
- security
- title - The name of the current document as described between the header TITLE tags.
- URL - The location of the current document.
- vlinkColor - The color of visited links as specified in the <body> tag/

forms

- Form validation normally used to occur at the server, after the client had entered all the necessary data and then pressed the Submit button. If the data entered by a client was incorrect or was simply missing, the server would have to send all the data back to the client and request that the form be resubmitted with correct information. This was really a lengthy process which used to put a lot of burden on the server.
- JavaScript provides a way to validate form's data on the client's computer before sending it to the web server. Form validation generally performs two functions.
- **Basic Validation** − First of all, the form must be checked to make sure all the mandatory fields are filled in. It would require just a loop through each field in the form and check for data.
- **Data Format Validation** − Secondly, the data that is entered must be checked for correct form and value. Your code must include appropriate logic to test correctness of data.

statements

- In HTML, JavaScript statements are "instructions" to be "executed" by the web browser.

Example

document.getElementById("demo").innerHTML = "Hello Dolly.";

This statement tells the browser to write "Hello Dolly." inside an HTML element with id="demo"

Functions

- A JavaScript function is defined with the **function** keyword, followed by a **name**, followed by parentheses **()**.
- Function names can contain letters, digits, underscores, and dollar signs (same rules as variables).
- The parentheses may include parameter names separated by commas: **(*parameter1, parameter2, ...*)**
- The code to be executed, by the function, is placed inside curly brackets: **{}**
- function *name*(*parameter1, parameter2, parameter3*) {
    code to be executed
  }
- Function **parameters** are the **names** listed in the function definition.
- Function **arguments** are the real **values** received by the function when it is invoked.
- Inside the function, the arguments are used as local variables.

Object in Java Script

- In real life, a car is an **object**.
- A car has **properties** like weight and color, and **methods** like start and stop:

Object	Properties	Methods

	car.name = Fiat
	car.model = 500
	car.weight = 850kg
	car.color = white
		car.start()
		car.drive()
		car.brake()
		car.stop()

- All cars have the same **properties**, but the property values differ from car to car.
- All cars have the same **methods**, but the methods are performed at different times.
- JavaScript variables are containers for data values.

This code assigns a **simple value** (Fiat) to a **variable** named car:

var car = "Fiat";

- Objects are variables too. But objects can contain many values.

This code assigns **many values** (Fiat, 500, white) to a **variable** named car:

var car = {type:"Fiat", model:500, color:"white"};

- The values are written as **name:value** pairs (name and value separated by a colon).

Object Properties

The name:values pairs (in JavaScript objects) are called **properties**

var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};

Object Methods

Methods are **actions** that can be performed on objects

function() {return this.firstName + " " + this.lastName;}

event and event handling

- An HTML event can be something the browser does, or something a user does.
- Here are some examples of HTML events:
- An HTML web page has finished loading
- An HTML input field was changed
- An HTML button was clicked
- Often, when events happen, you may want to do something.
- JavaScript lets you execute code when events are detected.
- HTML allows event handler attributes, **with JavaScript code**, to be added to HTML elements.

Example

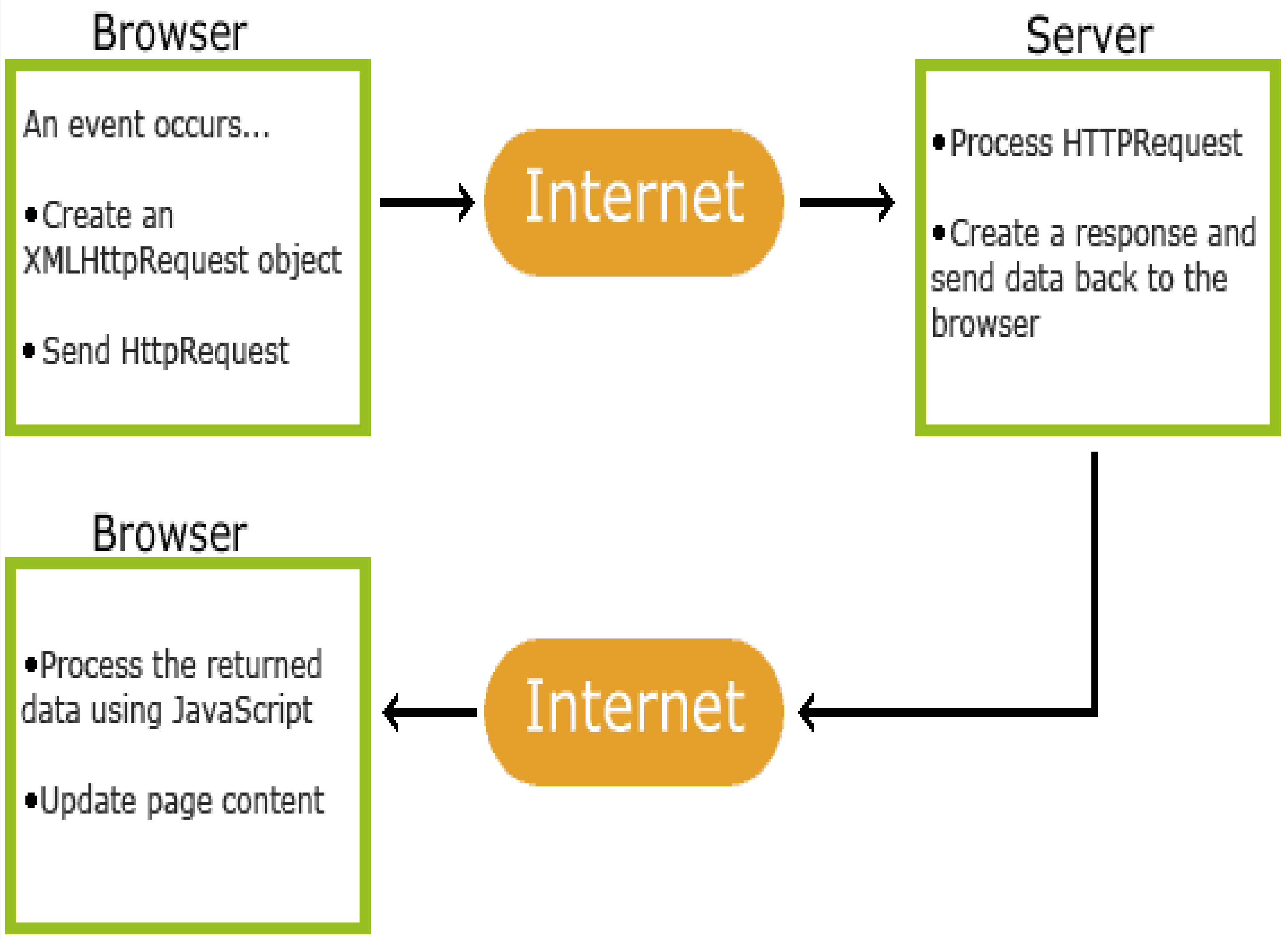
```
<button onclick='getElementById("demo").innerHTML=Date()'>The time is?</button>
```

an onclick attribute (with code), is added to a button element

introduction to AJAX

- AJAX is about updating parts of a web page, without reloading the whole page.
- AJAX = Asynchronous JavaScript and XML.
- AJAX is a technique for creating fast and dynamic web pages.
- AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.
- Classic web pages, (which do not use AJAX) must reload the entire page if the content should change.
- Examples of applications using AJAX: Google Maps, Gmail, Youtube, and Facebook tabs.

How AJAX Works


Browser
An event occurs...
•Create an XMLHttpRequest object
•Send HttpRequest
Internet
Server
•Process HTTPRequest
•Create a response and send data back to the browser
Browser
•Process the returned data using JavaScript
•Update page content
Internet

VB Script

- VBScript is a Microsoft scripting language.
- VBScript is a light version of Microsoft's programming language Visual Basic.
- VBScript is the default scripting language in ASP (Active Server Pages).
- VBScript is a scripting language
- A scripting language is a lightweight programming language
- VBScript is a light version of Microsoft's programming language Visual Basic
- VBScript is the default language in ASP (Active Server Pages)

Java Beans and Web Servers

- A Java Bean is a reusable software component that can be visually manipulated in builder tools.
- Reusable software components are designed to apply the power and benefit of reusable, interchangeable parts from other industries to the field of software construction.

Advantage

- Bean obtains all the benefits of Java's "write-once, run-anywhere" paradigm.
- The properties, events, and methods of a Bean that are exposed to an application builder tool can be controlled.
- A Bean may be designed to operate correctly in different locales, which makes it useful in global markets.
- Auxiliary software can be provided to help a person configure a Bean. This software is
  only needed when the design-time parameters for that component are being set. It
  does not need to be included in the run-time environment.
- The configuration settings of a Bean can be saved in persistent storage and restored
  at a later time.
- A Bean may register to receive events from other objects and can generate events that
  are sent to other objects.

Properties

- A JavaBean property is a named attribute that can be accessed by the user of the object. The attribute can be of any Java data type, including classes that you define.
- A JavaBean property may be read, write, read only, or write only. JavaBean properties are accessed through two methods in the JavaBean's implementation class:
- A read-only attribute will have only a get**PropertyName**() method, and a write-only attribute will have only a set**PropertyName**() method.

BDK

The installation of the JavaBeans Development Kit involves the following.
The JavaBeans Development Kit (BDK) downloads into a single file.
You will need to execute the kit (in Windows) or unpack (in Unix) this file in order to install the complete BDK.
Unlike the Java Development Kit (JDK), none of the BDK components are optional, so the installation is very simple.

Introduction to EJB

- Enterprise JavaBeans (EJB) is the server-side component-based middleware standard for Java. The specification, led and driven by Sun with participation from many key industry vendors, aims at unifying application server vendors to support a standard architecture so that compliant business applications are not only platform independent, but vendor independent.
- EJB is intended for complex distributed systems that use a purchased application server to handle the 'plumbing', or low-level technical workload. Encapsulation and separation of the technical services layer from the business application layer allows application developers to focus on solving business problems.

Benefits of EJB

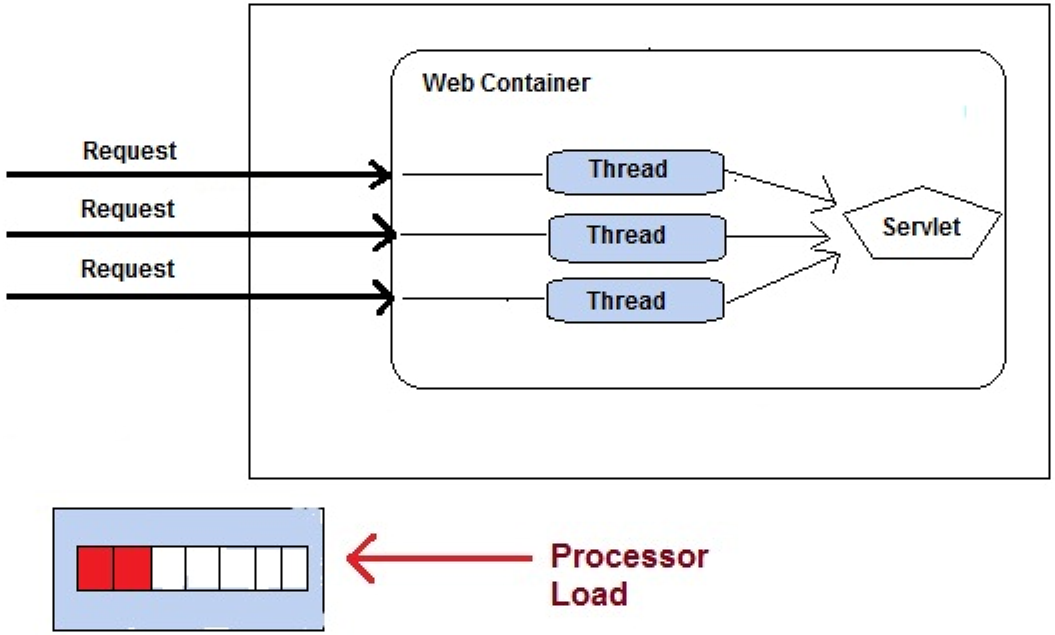
- **Decreases time to market.** EJB greatly simplifies development of complex distributed transactional systems by shielding application builders and architects from the complexity of distributed transactional systems and enabling business-focused design and implementations.
- **Horizontally scalable.** With its distributed architecture, EJB is designed to take advantage of more machines as needed without requiring changes to or redesign of the business application.
- **Reduces vendor and platform dependence.** Being a Java technology, EJB provides a standard mechanism for developing component-based applications regardless of the platform or software vendor used. This allows, for instance, development of a large UNIX-based system to take place in an inexpensive NT environment.
- **Promotes sound architecture.** EJB by specification separates data access, business logic, and presentation into separate layers for maximum flexibility and maintainability.
- **Broad industry adoption.** The specification was the joint effort of industry leaders. As EJB takes hold, it will provide the foundation for a rich COTS server-side component market.
- **Promotes Reuse.** Being component-based and object-oriented, EJB promotes project and enterprise level reuse and consistency.

Java
Beans API

- An application-programming interface (API) is a set of programming instructions and standards for accessing a Web-based software application or **Web tool**. A software company releases its API to the public so that other software developers can design products that are powered by its service.
- For example, Amazon.com released its API so that Web site developers could more easily access Amazon's product information. Using the Amazon API, a third party Web site can post direct links to Amazon products with updated prices and an option to "buy now."
- The JavaBeans API and its implementation are contained in the java.beans package. A few of the classes are used by beans while they run in an application.

Introduction to Servelets

- **Servlet** Technology is used to create web applications. **Servlet** technology uses Java language to create web applications.
- Less response time because each request runs in a separate thread.
- Servlets are scalable.
- Servlets are robust and object oriented.
- Servlets are platform indepe



JSDK

- JSDK (Java Servlet Development Kit) is a package containing all the classes and interfaces needed to develop servlets. JSDK also contains a web server and servlet engine to test your creations. The servlet engine provided in JSDK is a basic one(but free). There are many other servlet engines much more robust and can be interfaced with most major web servers of the market.

Servlet API & Servlet Packages

- Servlets have access to the entire family of Java APIs, including the JDBC API to access enterprise databases.

Servlet Packages

- Servlet API has two packages:

javax.servlet

javax.servlet.http

- javax.servlet: The javax.servlet package contains a number of classes and interfaces . The classes in javax.servlet package are protocol independent.

HTTP package

javax.servlet.http: The javax.servlet.http package contains a number of classes and interfaces. Some of the classess and interfaces in the javax.servlet.http extends those specified in the javax.servlet package.This interface specifies the contact between the web container and a servlet.

Working with HTTP request and response

- **HTTP Request Methods: GET and POST**

Two commonly used methods for a request-response between a client and server are: GET and POST.

GET - Requests data from a specified resource

POST - Submits data to be processed to a specified resource

- HTTP Response Codes indicate whether a specific HTTP requests has been successfully completed. Responses are grouped in five classes: informational responses, successful responses, redirections, client errors, and servers errors.

Security Issues

- Personal Information
- Abuse of Server Log Information
- Transfer of Sensitive Information
- Encoding Sensitive Information in URI's
- Attacks Based On File and Path Names
- DNS Spoofing
- Location Headers and Spoofing
- Proxies and Caching
- Denial of Service Attacks on Proxies