

Internet Technologies Notes

**Contributor: Tarushi
[SPM (DU)]**

Computer Science Notes

Download **FREE** Computer Science Notes, Programs, Projects, Books for any university student of BCA, MCA, B.Sc, M.Sc, B.Tech CSE, M.Tech at
<https://www.tutorialsduniya.com>

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

PLANNER

January-February 2016

Ch-2

Types and
Variables



नेशनल इन्श्योरेन्स
National Insurance

Trusted Since 1906

Friday

29

In Java, every value has a type.

System.out → PrintStream

Saturday 30

Sunday 31

A variable is a storage location in the computer's memory that has a type, a name and a contents.

Monday

1

String greeting = "Hello, World!" ;

PrintStream printer = System.out;

int luckyNumber = 13 ;

printer.println(greeting);

printer.println(luckyNumber);

Tuesday

2

Mixture of lowercase and uppercase letters is sometimes called "camel case" because the uppercase letters stick out like the humps of a camel.

Wednesday

3

type Name variableName = value ;

or

type Name variableName ;

Thursday

4

int LNO = 12 ;

int LNO = 13 ;

Object is an entity that can be manipulated by calling methods.

- PLANNER
February 2016
- National Insurance
Trusted Since 1906
- Friday 5 A method is a sequence of instructions that accesses the data of an object.
- Saturday 6 Sunday 7 `PrintStream` $\begin{cases} \text{println} \\ \text{print} \end{cases}$
- Monday 8 String - length (it counts the no. of characters in a string)
- Tuesday 9 String greeting = "Hello, World!";
int n = greeting.length();
`toUpperCase();` → It creates another String object that contains the characters of the original string, with lowercase letters converted to uppercase.
- Wednesday 10 String name = "Taruhi";
String s_uppercase = name.toUpperCase();
The public interface of a class specifies what you can do with its objects.
- Thursday 11 The implicit parameter of a method call is the object on which the method is invoked.
A parameter is an input to a method.

PLANNER
February 2016



Friday
12

The object on which you invoke the method is also a parameter of the method call, called the implicit parameter.

Saturday 13

Sunday 14

Println() method does not return a value.

public String replace (String target, String replacement)

Monday
15

A method name is overloaded if a class has more than one method with the same name.

Tuesday

16

Advantages of integers over floating point numbers:-

- They take less storage space.
- Processed faster
- Don't cause rounding errors.

Wednesday

17

In Java, numbers are not objects & no. types are not classes.

Rectangle box = new Rectangle (5, 10, 20, 30);

Thursday

18

The process of creating a new object is called construction.

box = new Rectangle (20, 50, 20, 30);

PLANNER

Accessor & Mutator
Array & ArrayList

February 2016



Trusted Since 1906

Friday

19

An accessor method does not change the state of its implicit parameter.
A mutator method changes the state.

Saturday 20

`box.translate(15, 25);` → It moves a rectangle by a certain distance.

Sunday 21

Ch-7

Monday

22

ARRAYS AND ARRAY LIST

Tuesday

23

Array of objects

`BankAccount[] accounts = new BankAccount[10];`

Wednesday

24

`data.length` → returns length of array.
4 final public instance variable -
Bounds error → when we try to use the index that is not in the array.

Thursday

25

`int [] primes = {2, 3, 5, 7, 11};`

PLANNER *Array list* *<BankAccount>()*
 February-March 2016 *obj. add(new BankAccount(1))*



ARRAY LISTS

26

- Array lists can grow and shrink as needed.
- The `ArrayList` class supplies methods for many common tasks, such as inserting and removing elements.

Saturday 27

Sunday 28

`ArrayList<BankAccount> accounts = new ArrayList<BankAccounts>();`

↳ type parameter

29

`accounts.add(new BankAccount(1001));`

The `ArrayList` manages a sequence of objects.

Tuesday 1

`ArrayList` is called a generic class.

By replacing `BankAccount` with any other class and get a different array list type.

`ArrayList<T>` → collects objects of type `T`.

`accounts.get(2);`

2

`BankAccount anAccount = new BankAccount(1729);`
`accounts.set(2, anAccount);`

Thursday 3

The `set` method can only overwrite existing values.

3

`accounts.add(i, a);` → It adds the object `a` at position `i` & moves all elements by one position.



PLANNER

March 2016

Friday

4

`accounts.remove(1)` → removes the element at position 1, moves all elements after the removed element down by one position.

Saturday 5

`Array - a.length;`

Sunday 6

`ArrayList - a.size();``String - a.length();`

Monday

7

Versions of Java prior to version 5.0 had only an untyped class `ArrayList`. Whenever you retrieve an element from an untyped array list, the compiler requires you to use a cast.

Tuesday

8

`BankAccount a = (BankAccount) a.get(0);`

The cast is needed because the compiler does not keep track of the objects that were inserted into the array list, & the array list get method has return type `Object`.

Wednesday

9

To treat primitive type values as objects, you must use wrapper classes.

Thursday

10

`byte → Byte``boolean → Boolean``char → Character``double → Double``float → Float``int → Integer`

PLANNER

March 2016

Wrapper class is a class
whose object wraps
or contains a primitive
data type.

नेशनल इन्�श्योरेन्स
National Insurance
Trusted Since 1906

long → Long
short → Short

Friday

11

Each wrapper class object contains a value
of the corresponding primitive type.

Saturday 12

Sunday 13

Auto-Boxing (auto-wrapping) → Conversion
between primitive types and the
corresponding wrapper classes. Is automatic.

Monday

14

Double d = 29.95; → auto-boxing

double x = d; → auto-unboxing

x = d.doubleValue();

double x = d.doubleValue();

Auto-boxing even works inside arithmetic
expressions.

Tuesday

15

Double e = d + 1;

ArrayList<Double> data = new ArrayList<
<Double>();

data.add(29.95);

Wednesday

16

double x = data.get(0);

Enhanced For loop

Thursday

17

double [] data = ...;

double sum = 0;

for (double e : data)

sum = sum + e;

4



PLANNER

March 2016

Friday

18

The enhanced for loop traverses all elements of a collection.

Difference b/w for-each loop & for loop:

Saturday 19 In for each loop, the element variable e is assigned values data[0], data[1]. In for loop, the index variable i is assigned values 0, 1, and so on.

Monday

21

To count values in an array list, check all elements & count the matches until you reach the end of the array list.

Tuesday

22

```
int[][] b = new int[5][0];
for (int i=0; i< b.length; i++)
    b[i] = new int[i+1];
```

Wednesday

23

An array variable stores a reference to the array.
Use clone to copy elements of array.

Thursday

24

`double[] prices = (double[]) data.clone();`
Clone method has the return type Object.
`System.arraycopy (from, fromStart, to, toStart,
count);`

PLANNER

April 2016

Ch-8

Introduction to JavaScriptनेशनल इन्श्योरेन्स
National Insurance

Trusted Since 1906

Friday

- 1 An OOP environment always offers event driven programming. This means that the programming environment will recognize object based events & allow the connection of code snippets to these events.
- 2 When an events occurs, the code snippets will execute.

Monday

3 JavaScript is a scripting language created by Netscape.

4 The Netscape client browser product is called Netscape commerce Server.

Tuesday

5 Netscape has a product called Live Wire, which permits server side JavaScript code, to connect to Relational Database Management Systems (RDBMS).

Wednesday

6 Livewire database drivers also support a number of relational databases.

<HEAD> OR <BODY>
<SCRIPT>... </SCRIPT>
</HEAD> OR </BODY>

Thursday

7 JavaScript uses the filename.html & the HTTP protocol to transport itself from the web server to the client's browser where the Java Script executes & processes client information.

PLANNER
April 2016

- Friday 8 The default scripting language of Internet Explorer is VB Script. NC does not support VB support script.
- Saturday 9 <FORM> ... </FORM>
It can be used to create a user Request-form.
- Sunday 10 <INPUT> ... </INPUT>
It can be used to instantiate HTML objects in the HTML form to facilitate the capture of user data.
- Monday 11 Objects - Text, TextArea, RadioButtons, Push Buttons, Check Boxes.
- Tuesday 12 JavaScript is used to validate and/or process such user input.
Once user input passes the validation tests applied, and/or has been processed by client side, the form data captured will have to be passed backward to the web server from where the HTML file originated.
- Wednesday 13 User input information will be assembled in the web server, converted to an HTML file & returned to the user. Such an HTML page is created on Demand.
- Thursday 14 An HTML page create on demand, based on user input, is a dynamically created HTML page.

April 2016

Friday

15

HTML is static.

Advantages of Javascript :-

(1) Short development cycle.

(2) Ease of learning.

(3) Small size scripts.

(4) An interpreted language - It requires no compilation steps.

(5) Embedded within HTML - Javascript

does not require any special or separate editor for programs to be written, edited or compiled.

(6) Minimal syntax - Easy to learn

(7) Quick development

19 Javascript does not require time-consuming compilations.

(8) Designed for simple, small programs.

(9) Performance

20

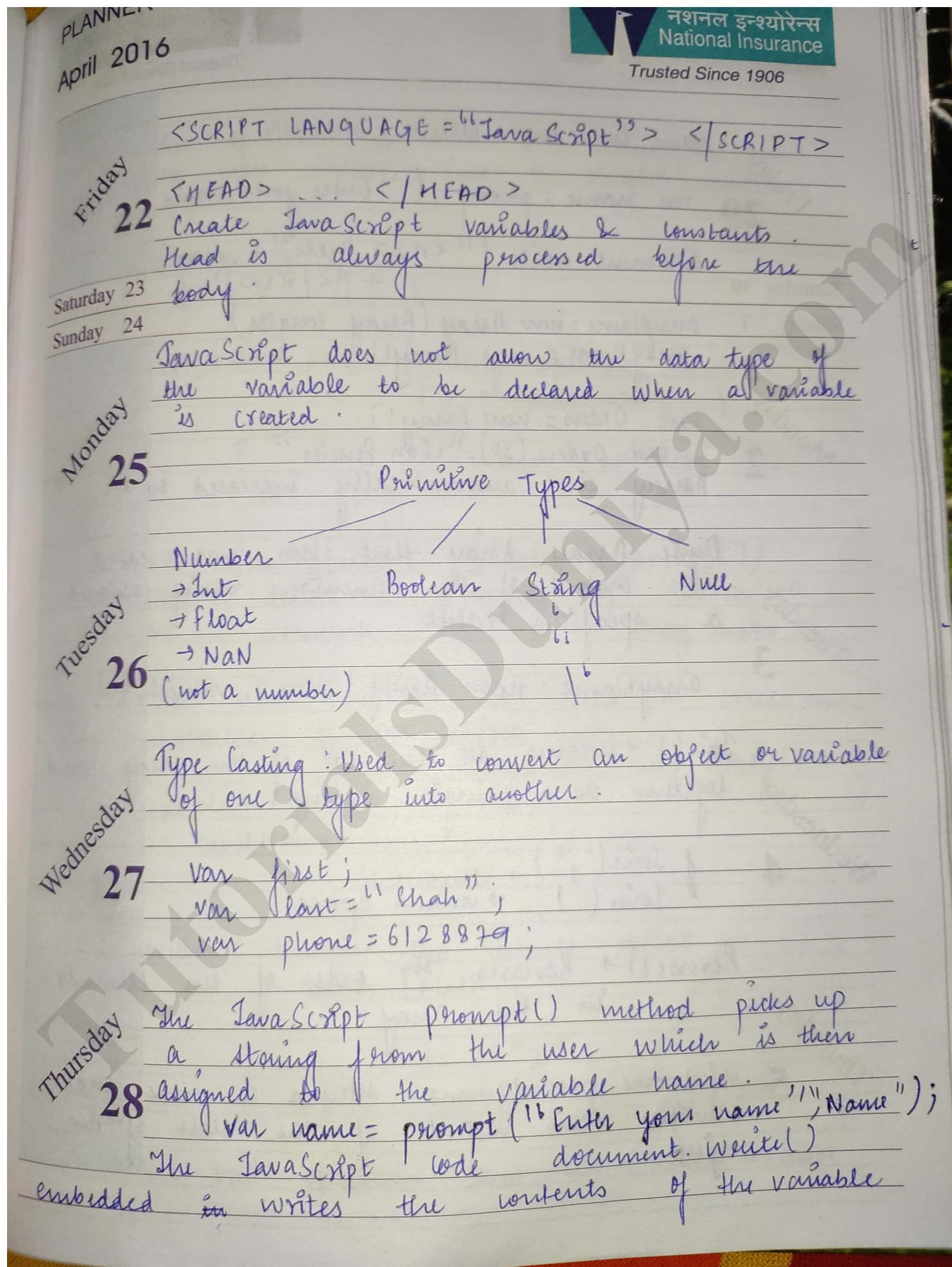
(10) Designed for programming User events: Javascript supports Object Event based programming.

(11) Easy debugging & testing.

21

(12) Platform Independence - It is a language understood by any Javascript enabled browser.

Thursday



PLANNER

April-May 2016


 National Insurance
Trusted Since 1906
P
N

Friday

29

name to the client browser -

`var name = prompt ("Enter your name", "Name");
document.write ("

Hello" + name); + "</h2>");`

Saturday 30

Sunday 1

`arrayName = new Array (Array length)
arrayName = new Array(),`

Monday

2

cust_Orders = new Array()

cust_Orders [50] = "100 Pencils"

Array size automatically increased to 51.

Tuesday

3

Dense Array - Array that has been created with each of its elements being assigned a specific value.

`arrayName = new Array (value 0, value 1, ..., value n)`

Join() → returns all elements of the array joined together as a single string

4
`join (",");
join (); → comma space.`

Reverse() → Reverses the order of the elements in the array.

5

objectname.propertyname - returns a value that identifies some aspect of the state of an object.

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

May 2016



Friday

6

An operation is used to transform one or more values into a single resultant value.

Saturday 7

Sunday 8

logical Operators
& &, ||, !

$pq + ab = pqab$ (String operator)
(+ → concatenation)

Monday

9

? : Temporary operator is
too supported.

Tuesday

10

The delete Operation (delete myArray[5])
Used to delete a property of an object
or an element at an array index.

Wednesday

11

The new operator is used to create an instance of an object type.

myArray = new Array()

Label

Label Name:

Statement

with → Identifies the default object
with (Math)

{

d = p1 * 2 * 1;

}

12

The void operation does not return a value.

May 2016

Friday

13

`eval()` → Used to convert a string expression to a numeric value.

Saturday 14

`parseInt()` → Convert a string value to an integer.
↳ returns the first integer contained in a string or 0 if the string does not begin with an integer.

Sunday 15

`var string2Num = parseInt('123xyz');`

Monday

16

`var u = "xyz";
u = 0(NaN);`

Tuesday

17

`parseFloat()` → First floating point number or 0.
<head>

`function name(,)`

`y
</head>`

Wednesday

18

Parsed → Interpreter converts into a pseudo-compiled bytecode.

Thursday

19

`Alert Dialog Box`

`alert()` method

takes a string

as an argument

& displays an

alert dialog box

in the browser

window.

`Dialog Box`

`Prompt Dialog Box`

`prompt()` method

instantiates the

prompt dialog box

which displays a

specified message -

`Confirm Dialog Box`

→ A pre-defined

message

→ OK & Cancel.



PLANNER
May 2016



Friday
20

The JavaScript & HTML continue processing until the OK button is clicked.

Saturday 21

```
alert ("<Message>");
```

Sunday 22 The prompt dialog box provides a single data entry field, which accepts user input.

Monday
23

Prompt dialog box :-

- Displays a predefined message.
- Displays a textbox & accepts user input.
- Displays OK & Cancel.

```
prompt ("<Message>", "<Default Value>");
```

Tuesday
24

```
confirm ("<Message>");
```

Wednesday
25

Thursday
26

PLANNER

May-June 2016

Ch-9
The JavaScript Document
Object Model

नेशनल इन्श्योरेन्स
National Insurance
Trusted Since 1906

Friday

27

An HTML page is painted in a browser. The browser assembles all the elements contained in the HTML page, downloaded from the web server, in its memory.

Saturday 28

The browser paints objects in browser window.

Sunday 29

Once HTML page is painted in the browser window, the browser can no longer recognize individually HTML elements.

Monday

30

JavaScript enabled browsers are capable of recognizing individual objects in HTML page after the page has been painted in the browser because the JavaScript enabled browser recognizes & uses the Document Object Model (DOM).

Tuesday

31

DOM defines the logical structure of documents & the way a document is accessed & manipulated.

Wednesday

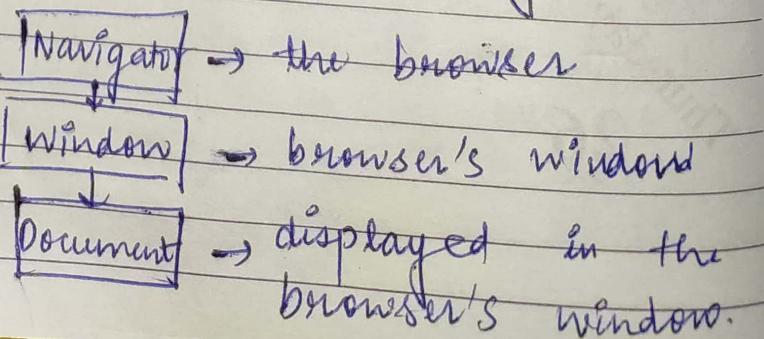
1

Using DOM, JavaScript enabled browsers identify the collection of web page objects (web page in a browser window).

Thursday

2

HTML objects have a descending relationship with each other.



PLA
June 2016

Friday 3

Form

```

graph TD
    Form --> Text_Boxes[Text Boxes]
    Form --> Labels[Labels]
    Form --> Radio_Button[Radio Button]
    Form --> Check_Boxes[Check Boxes]
    Form --> Push[Push]
  
```

Saturday 4

Sunday 5 Java Script's DOM (instance hierarchy)
Navigation
Windows

Monday 6

```

graph TD
    Document[Document] --> Anchor[Anchor]
    Document --> Link[Link]
    Document --> Form[Form]
    Form --> TextBox[TextBox]
    Form --> Textarea[Textarea]
    Form --> RadioButton[RadioButton]
    Form --> Checkbox[Checkbox]
    Form --> Select[Select]
    Form --> Button[button]
  
```

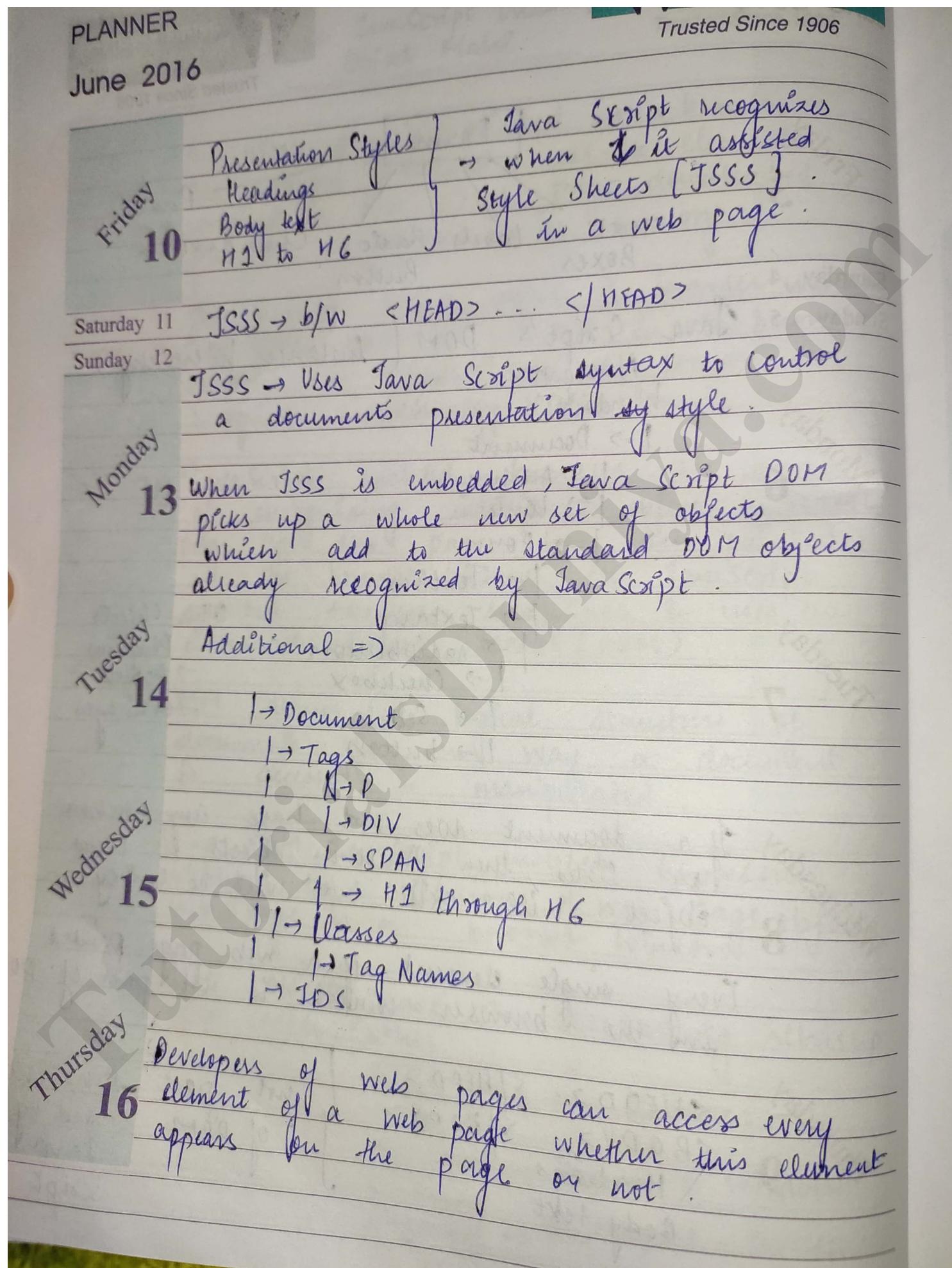
Tuesday 7 Other Objects
→ Plug-ins
→ Applets
→ Images

Wednesday 8 If a document does not have any anchor or links, then Anchor objects & Links object will exist but will be empty.

Thursday 9 Every single element of a Web page painted in the browser window is not part of DOM.

<HEAD> </HEAD>
<BODY> </BODY>
Headings
Body text

} not part → not recognized by Java Script



- PD
June 2016
- National Insurance
Trusted Since 1906
- Friday 17 **Java Script Objects**
- (1) Properties that determine the functionality of the object.
- Saturday 18 (2) Methods that allow access to these properties.
- Sunday 19
- Monday 20 (3) Events that allow JavaScript code snippets to be connected to the object by being mapped to JavaScript event handlers.
- When a predefined event occurs the code snippet will execute.
- Tuesday 21 Java Script can access the methods of all objects belonging to the DOM & JSSS DOM.
- Java Script is an object-based programming language it is not full OOP.
- It does not fully support basic OOP capabilities like classification, inheritance, encapsulation & information hiding.
- It provides recognition of a no. of predefined browser & screen objects.
- It can control the behaviour of these objects through their properties & methods (at run time).

June 2016

Friday

24

TextBox

Used in an HTML form to accept user input.

Saturday 25

It is an object that belongs to DOM.

Sunday 26

JS facilitates access to all the methods of an textbox.

Monday

27

ObjectName.PropertyName

Methods of an object are used to set or get value of an object's property.

Tuesday

28

When any JS enabled browser loads a web page, the browser automatically creates a number of JS objects that map to the DOM.

DOM provides JS access to the HTML objects that are contained in the web page.

Wednesday

29

Table 9.1 Diagram 9.3

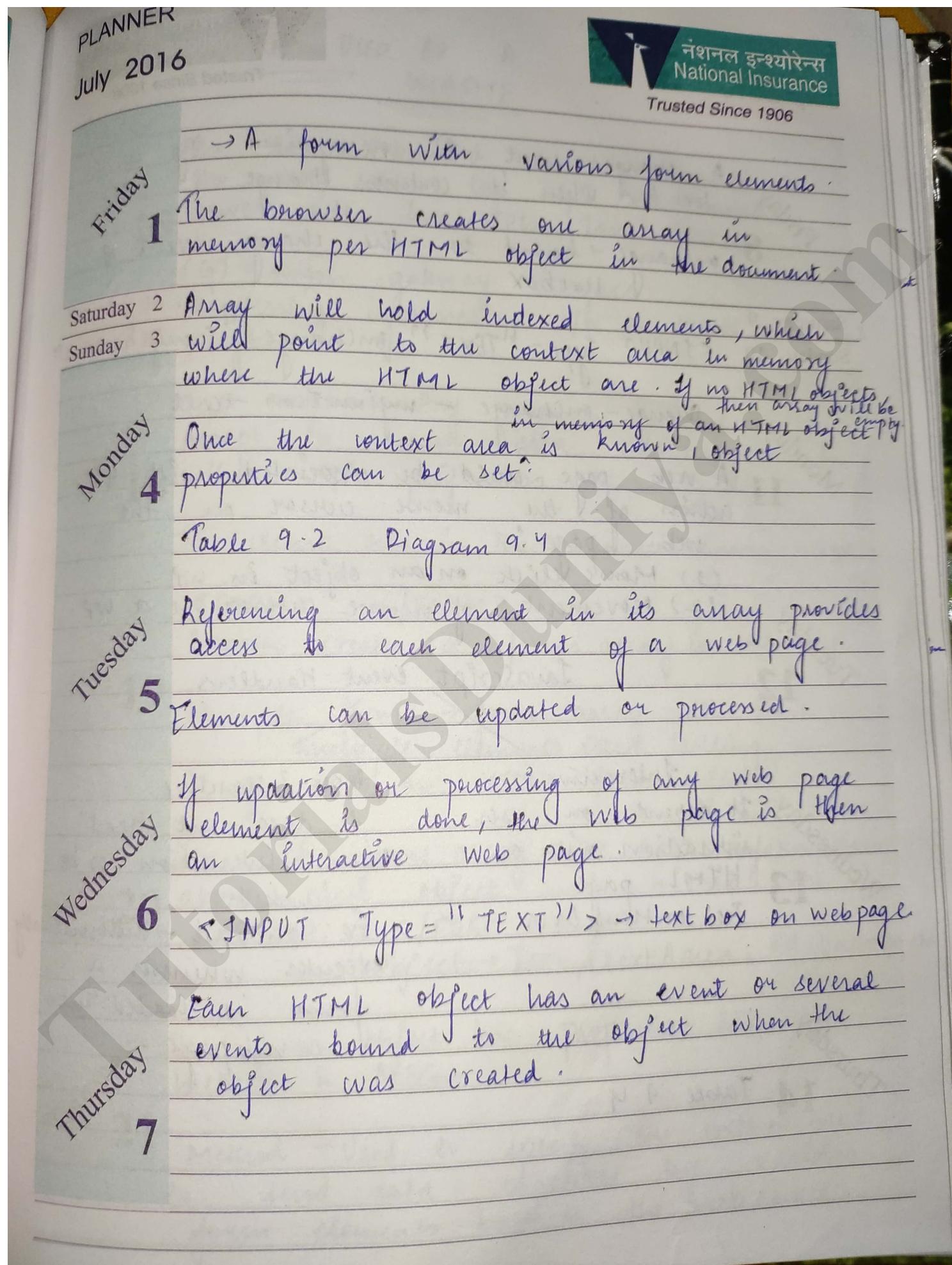
HTML Objects

- Images
- Image Maps
- Hyperlinks
- Frames
- Anchors
- Applets

→ Multimedia objects such as audio files.

Thursday

30



PLANNER

July 2016

National Insurance
Trusted Since 1906

Friday

8

A change event is recognized by a text box when its contents change.

Saturday 9

Sunday 10 `<INPUT Type = "TEXT" onchange = "< my function">"`

Change - onchange - myfunction → call

Monday

11

A web page could be associated with the action of the mouse cursor on the web page :-

(1) Mouse Click on an object in WP.

(2) Movement of mouse cursor on a WP.

Tuesday

12

JavaScript event Handlers

Wednesday

13

Interactive

It depends on user interaction with an HTML page.

Ex: onMouseOver

Non-Interactive

It does not need user interaction to be invoked.

Ex: onLoad → automatically executes whenever a form is loaded into a web page.

Thursday

14

Table 9.4

Ch-10

FORMS USED BY A WEBSITE

PLANNER
JULY 2016

National Insurance
Trusted Since 1906

Friday 15
The data submitted can be processed at the web server by CGI programs, server side JavaScripts, Java Servlets.

Saturday 16
Transferring information b/w a WWW & a CGI program.

Monday 18
CGI program - Any program designed to accept & return data that conforms to the CGI specification.

JS → client side - in the browser window - work.

Tuesday 19
<FORM> ... </FORM>
Browser creates a forms array in memory.

Wednesday 20
Multiple forms → form array will have multiple elements each holding a reference to an HTML form object.
The JS forms array also holds information about each object used within the <FORM> ... </FORM> tags.
↳ Objects → Text, Text Area, RB, Buttons etc

Thursday 21
The form object has properties like name, method & Action.

Method → Used to specify the method used to send data captured by various form elements back to the web server.

PLANNER

July 2016

नशनल इन्श्योरेन्स
National Insurance
Trusted Since 1906

Friday

22

Method - Get OR POST

Saturday 23

Sunday 24

The Get method sends the data captured by form elements to the web server encoded into a URL, which points to a web server. Used for small amount.

Max^m data that can be sent back to the web server - 1024 bytes.

Monday

25

The Post method sends the data captured by form elements back to the web server as a separate bit-stream of data. Used for large amount.

Tuesday

26

The Action attribute points to the URL of a program on the web server that will process the form data captured & being sent back.

Wednesday

27

Perl Ans C → Hxt Unix

Table 10.1

Table 10.2

Thursday

2) TEXT ELEMENT

28

Data entry st. accepts a fields used in HTML forms. Single line of text entry.

PLANNER
July-August 2016

National Insurance
Trusted Since 1906

Friday 29

Properties

- name
- value

Events

- focus()
- blur()
- select()
- change()

Methods

- focus()
- blur()
- select()

Event Handlers

- onFocus()
- onBlur()
- onSelect()
- onChange()

1 `<INPUT Name=" " Type="Text" Value=" ">`

2) PASSWORD ELEMENT

2 All keystrokes for this field are displayed as an asterisk [*].

Wednesday

Properties

- name
- value
- defaultValue

Methods

- focus()
- blur()
- select()

Thursday

3 **Events**

- focus()
- blur()
- select()
- change()

Event Handlers

- onFocus()
- onBlur()
- onSelect()
- onChange()

4 `<INPUT Name=" " Type="Password" Value=" ">`

PLANNER		National Insurance Trusted Since 1906	PLA AUS
August 2016			
Friday	5	<u>⇒ BUTTON ELEMENT</u> Properties → name → value	Methods → click()
Saturday	6		
Sunday	7	Event → click()	Event Handler → onclick()
Monday	8	<code><INPUT Name = " " Type = "Button" Value = " " ></code>	
Tuesday	9	<u>⇒ SUBMIT ELEMENT</u> It submits the current data held in each data aware form element to a web browser server.	Method → click()
Wednesday	10	Properties → name → value	
Thursday	11	Event → click()	Event Handler → onclick()
		<code><INPUT name = " " Type = "Submit" Value = " " ></code>	
		<u>⇒ RESET ELEMENT</u>	
		Properties → name → value	Methods → click()

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

PLANNER
August 2016

Friday

12

Events

→ click()

Event Handler

→ onclick()

Saturday 13

CHECKBOX ELEMENT

Sunday 14

Properties

→ name

→ value

→ checked

Method

→ click()

Monday

15

Event

→ click()

Event Handler

→ onclick()

Tuesday

16

<INPUT Name=" " Type="checkbox" Value="YES/No" CHECKED>

Wednesday

RADIO ELEMENT

17

Properties

→ checked

→ index

→ length

→ name

Method

→ clicked()

Event

→ clicked()

Thursday

18

Event Handler

→ onclicked()

<INPUT Name=" " Type="radio" Value="1/0" CHECKED="CHECKED">

PLANNER

August 2016

Trusted Since 1906

Friday

19

⇒ TEXTAREA ELEMENT

Provides a way to create a custom sized, multiple line, text entry object, which can be placed on an HTML form.

Saturday 20

Sunday 21

Properties

- defaultChecked
- name
- value

Methods

- Focus()
- Blur()
- Select()

Monday

22

Events

- Focus()
- Blur()
- Select()

Event Handlers

- onFocus()
- onBlur()
- onSelect()

Tuesday

23

```
<INPUT TYPE="TextArea" Name=" " "   
ROW="10" COLS="25">  
<H2>Enter data here </H2> </TEXTAREA>
```

Wednesday

24

⇒ SELECT AND OPTION ELEMENT

↳ Drop down list

```
<SELECT Name="Items">  
<OPTION SELECTED> - - . </OPTION>  
<OPTION> - - .  
</SELECT> </OPTION>  
</OPTION>
```

Thursday

25

PLANNER
August-September 2016

National Insurance
Trusted Since 1906

⇒ MULTI CHOICE SELECT LISTS

Friday 26

```
<SELECT Name=" " SIZE=" " MULTIPLE>
<OPTION SELECTED> . . . </OPTION>
</SELECT>
```

Saturday 27

Sunday 28

memvarirname="arrayname.indexvalue">
Value of any element in array.

Monday 29

Properties

- selectedIndex
- defaultSelected
- index
- selected
- text
- value

Methods

- Blur
- Focus
- Change

Tuesday 30

Event Handlers

- onBlur()
- onFocus()
- onChange()

Events

- Blur()
- Focus()
- Change()

Wednesday 31

Thursday 1

Ch-9

PLANNER
September 2016

Introducing JavaServer Pages

National Insurance
Trusted Since 1906

Friday 9 JSP is designed to increase the efficiency in which programmers & even non-programmers can create web content.

Saturday 10 JSP is a technology for developing web pages that include dynamic content.

Sunday 11 When a user asks for a JSP page, the server executes the JSP elements, merges the results with the static parts of the page, & sends the dynamically composed page back to the browser.

Tuesday 13 CGI → Common Gateway Interface
The web server has to create a new OS process, load an interpreter and a script, execute the script, & then tear it all down again.

Wednesday 14 The JavaServer Pages Standard Tag Library is a collection of useful JSP tags which encapsulates the common functionality common to many JSP applications.

Thursday 15 JSTL Tags
→ Core Tags
→ Formattting Tags
→ SQL Tags
→ XML Tags
→ JSTL Functions

PLANNER

September 2016 Core tags

Trusted Since 1906

SEP

Friday

16 → <c:out>
It displays the result of an expression.

Saturday 17 <%@ taglibs prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

Sunday 18

Monday

19 It can automatically tags so that they are not evaluated as actual tags.

Ex: <c:out value="foo customer. address street"/> escape XML evaluated

Tuesday

20 Attributes
 → Value → Information to input (Required)
 → Default → Fallback information to output
 → escapeXml → True if the tag should escape special XML characters.

Wednesday

21 <c:out value="\${'<tag>', '}' /> → <tag>, &

→ <c:set>
Sets the result of an expression evaluation in a scope.
setProperty

Thursday

22 Attributes
 → Value → Information to save
 → Target → Name of the variable whose property should be modified
 → Property → Property to modify



PLANNER

September 2016

a. sum
1000

Friday

23

Saturday 24

Sunday 25

Monday

26

Tuesday

27

Wednesday

28

Thursday

29

→ var → Name of the variable to store information
 → scope → Scope of variable to store information.

```
<c:set var="salary" scope="session"
       value="$\{2000*2\}" />
```

```
<c:out value="\${salary}" /> → 4000
```

→ <c:remove>

It removes a variable from either a specified scope or the first scope where the variable is found (if no scope is specified)

Attributes

→ var (Name of the variable to remove) → required
 → scope (Scope " " " " ")

```
<c:remove var="salary" />
```

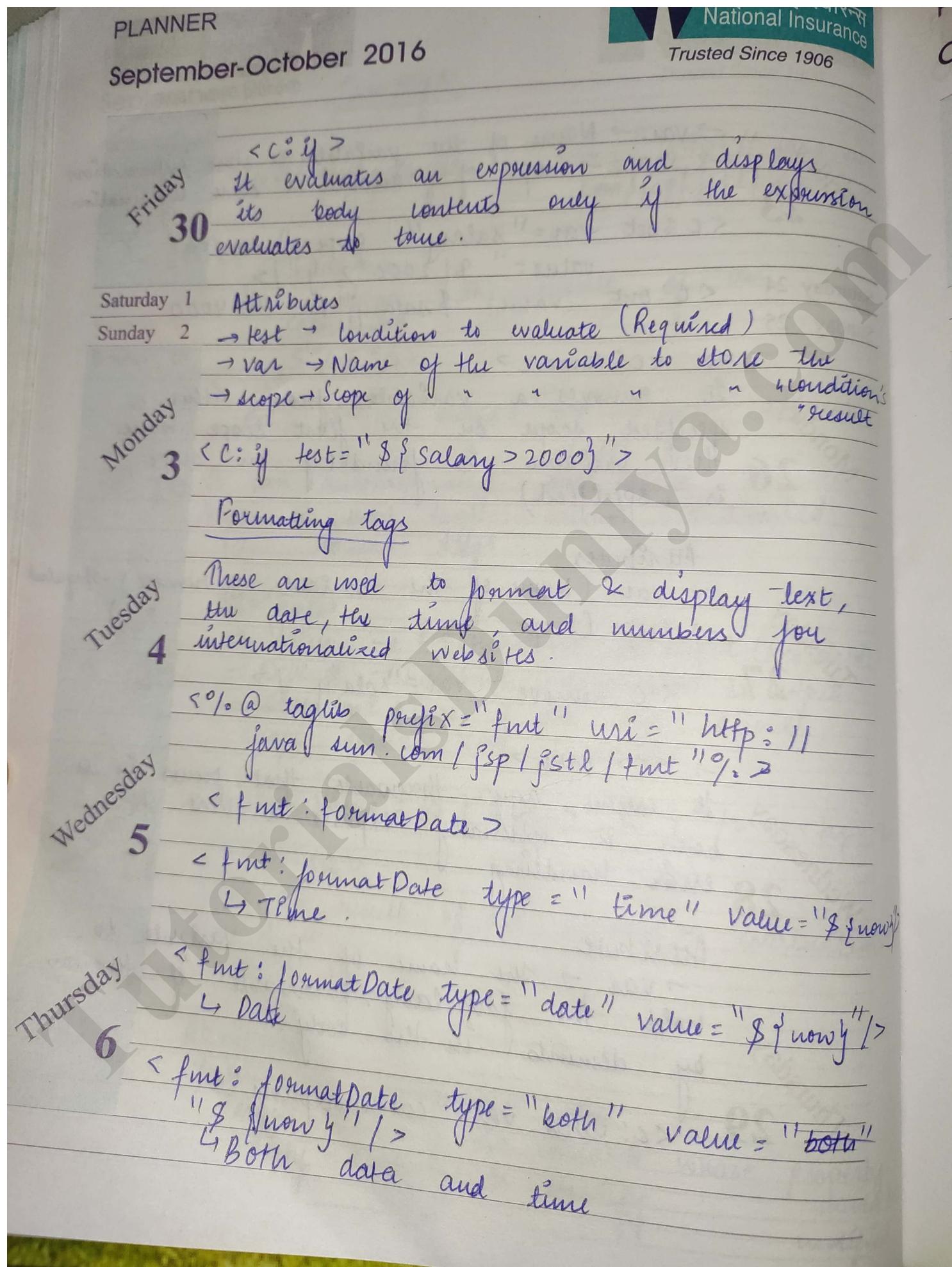
→ <c:catch>

It catches any throwable that occurs in its body & optionally exposes it. Used for error handling.

Attribute

→ var → The name of the variable to hold the java.lang.Throwable if thrown by elements in the body.

```
<c:catch var="catchException" />
```



PLANNER
October 2016Friday
7

Value - Date value to display (Required)
 Type - Date, time or both.

Saturday 8 It is used for interacting with RDBMS such as Oracle, MySQL.

Monday 10 `<%@ taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>`

Tuesday 11 `<sql:setDataSource>`
 It sets the data source configuration variable or saves the data source information in a scoped variable that can be used as input to the other actions.

Wednesday 12 `<sql:setDataSource var="snapshot" driver="com.mysql.jdbc.Driver" url="jdbc:mysql://localhost/TEST" user="Tanush" password="Tanush123"/>`

Thursday 13 `<sql:query dataSource="#snapshot" var="result">`
`<sql:query>`
`<%@ page import="java.sql.*"%>`
`<body>`
`</sql:query>`



PLANNER

October 2016

Friday

14

<sql:query>
it executes an SQL select statement
and saves the result in a scoped
variable.

Saturday 15

<sql:query datasource=" \${snapshot}"
var="result">

Sunday 16

Select * from employees;

</sql:query>

Monday

17

XML Tags

It is used to create and manipulate the XML documents.

Tuesday

18

<%@ taglib prefix="x" uri="http://
java.sun.com/jsp/jstl/xml"%>

JSTL functions

→ String manipulating functions

Wednesday

19

<%@ taglib prefix="fn"
uri="http://java.sun.com/jsp/jstl/functions"%>

Thursday

20

योरेन्स
urance
6

PLANNER
October 2016



Trusted Since 1906

JSP Java Beans

Friday
21

A Java Bean is a specially constructed Java class written in the Java language and coded acc. to the Java Beans API specifications.

Saturday 22
Sunday 23

Monday
24

Properties (Named attribute that can be accessed by the user of the object)
 / \
 getPropertyName() set PropertyName()
 Accessor method Mutator method.

→ getFirstName() → setFirstName()
 Accessor method. → Mutator method.

Tuesday
25

The useBean action declares a Java Bean you use in a JSP.

<jsp:useBean id="bean's name" scope=
 "bean's scope" typeSpec/> ↓
 page, request,
 session, app. based

Wednesday
26

Unique name

<jsp:useBean id="date" class="java.util.Date"/>

<p> The date / time is <%= date %>

Thursday
27

Accessing Java Beans Properties

<jsp:useBean id="id" class="bean's class"
 scope="bean's scope">

PLANE
October-November 2010
Trusted Since 1906
NC

Friday 28

<jsp:outProperty name="bean's id" property="property name" value="value"/>

<jsp:getProperty name="bean's id" property="property name"/>

Saturday 29

Sunday 30

<jsp:useBean>

Monday 31

Other core tags

→ <c:choose>

The works like a switch statement in that it lets you choose b/w a number of alternatives.

case → <c:when>

default → <c:otherwise>

Tuesday 1

→ choose tag does not have any attribute.

→ when tag has 1 attribute

→ test (condition to evaluate)

→ otherwise has no attribute.

Wednesday 2

<%@ taglib prefix="c" uri="%">

<html>

<jsp:useBean id="Clock" class="java.util.Date"/>

<c:choose>

<c:when test="`if (Clock.hours < 12)`">

Good Morning!

<c:when>

<c:otherwise>

Good evening!

Thursday 3



PLANNER
November 2016



Friday
4

< / c: otherwise >
< / c: choose >
< / body >

Saturday 5

Scripting elements
It provides the ability to insert java code inside the JSP.
→ scriptlet tag
→ expression tag
→ declaration tag

Monday 7

Scriptlet tag
It is used to execute java source code in JSP.

Tuesday 8

<% java source code %>

<% out.print ("Welcome") ; %>

Wednesday 9

Expression Tag

The code placed within this is written to the output stream of the response.

<% = statement %>

<% = "Welcome" %>

Thursday 10

Declaration Tag

Used to declare fields and methods.

<% ! field or method declaration %>

is
den
in
is
start

PLANNER

November 2016

Trusted Since 1906

Friday

11

Difference b/w JSP scriptlet tag & declaration tag.
 It can only declare variables (not methods).
 It can declare variables as well as methods.

Saturday 12

Sunday 13 Its declaration is placed inside the - jspService() method.

Its declaration is placed outside the - jspService() method.

Monday 14

`<% int data = 50; %>`
`<%= "Value is: " + data %>`

Tuesday

15

`<%!`
`int cube (int n) {`
 `return n * n * n;`

`}``%>`

`<%= "Cube is: " + cube(3) %>`

Wednesday

16

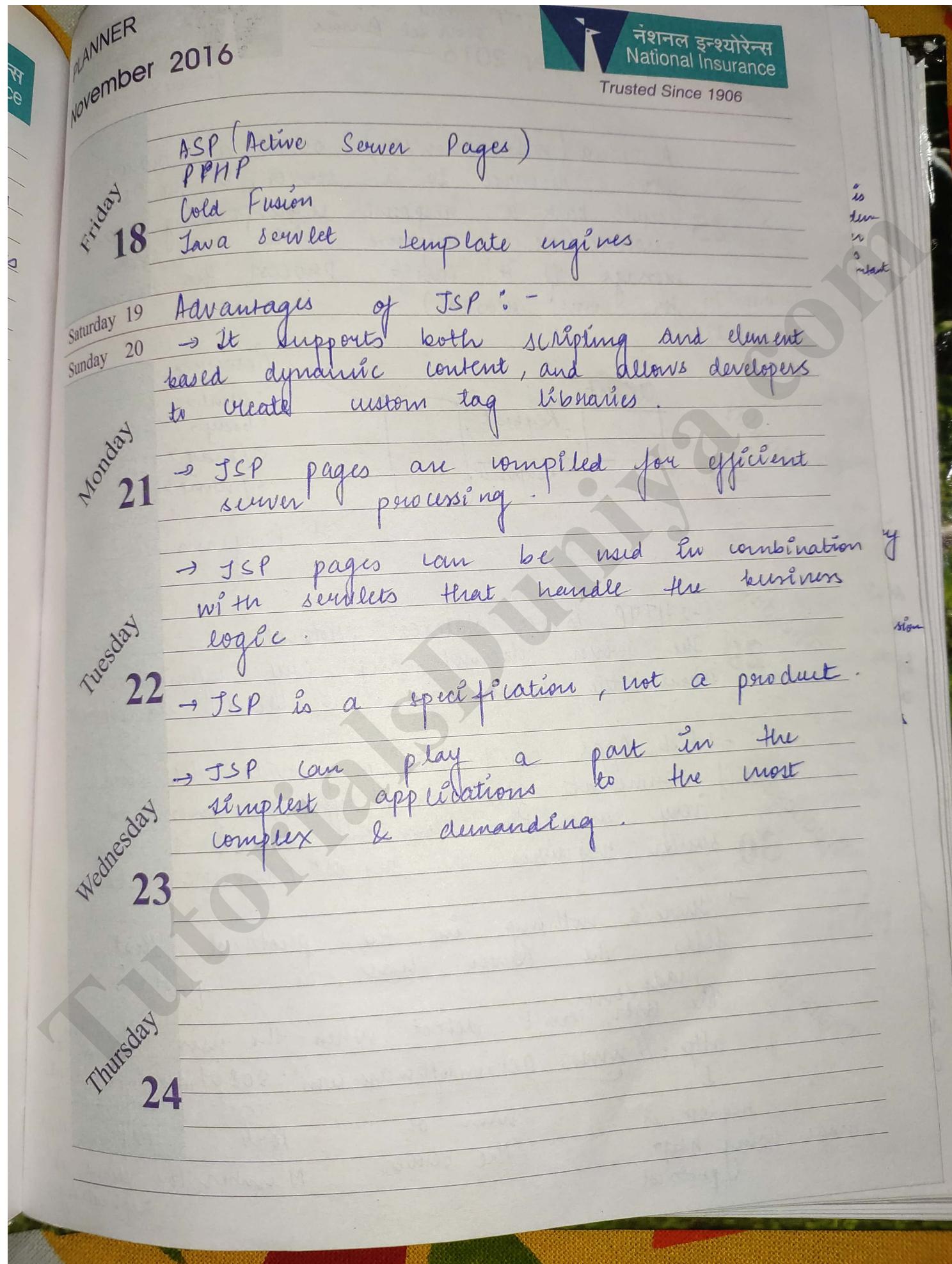
A JSP page is always compiled before it's processed by the server. It compiles each JSP

Thursday

17

Each JSP executable code requested and directly on all the requests.

The page is compiled into the first time its invoking the resultant



TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

PLANNER

November-December 2016

Ch-2
HTTP and Java
Servlet Basics

Friday

25

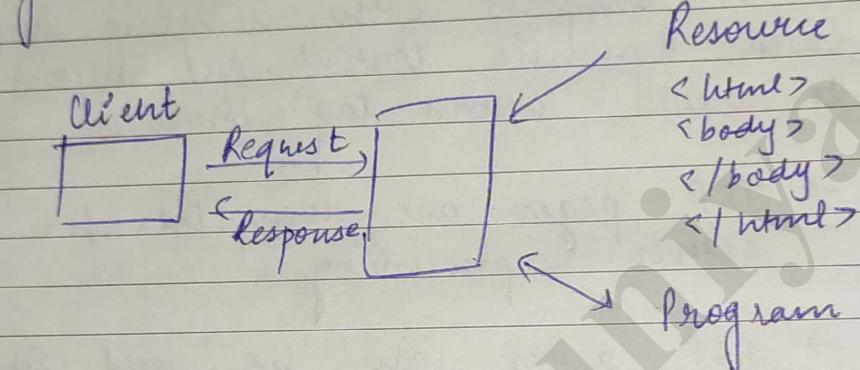
A client (web browser) sends a request for a resource to a server, & the server responds back a response corresponding to a resource (or a message if it can't process the request for some reason)

Saturday 26

Sunday 27

Monday

28



Tuesday

29

→ HTTP is a stateless protocol.
The server does not keep any information about the client after it sends its response.

Wednesday

30

→ Web Apps can't easily provide the kind of immediate feedback typically found in Every interaction b/w the clients & the server requires a request/response exchange.

Thursday

1

→ There's nothing in the protocol that tells the server how a request is made.

The user can't detect when the user closes the browser.

request is made using http protocol.

name of the server

Port

Resource that Number the client is requesting.

नशनल इन्श्योरेन्स
National Insurance
Part of Established Since 1906

PLANNER
December 2016

Port NO. - 80 (standard port for HTTP requests)

Identifies a resource, by its location.

Friday 2
When the request has been delivered to the server, the location is given & the browser uses the URL information to create the request message it sends to the specified server using the specified protocol.

Saturday 3
HTTP request message consists of 3 things:-

- A request line
- Request headers
- A request body

URN → Globally unique identifier

Monday 15
Request line → It starts with the request method

GET / index.html HTTP/1.1 name, followed by

Request method Resource identifier Protocol version identifier & name.

Default request method. used by the browser.

A GET request is used to retrieve a resource from the server. The request headers provide additional information the server may use to process the request.

Host: www.getionsoftware.com

User-Agent: Mozilla/5.0 (Windows; U; Win 9x 4.90; en-US; rv:1.0.2)

Accept: image/gif, image/jpeg, image/pjpeg, image/png, */*

Accept-Language: en

Accept-Charset: ISO-8856-1, *, utf-8.

Wednesday 7
It tells the server the hostname used in the URL. Used to distinguish b/w multiple virtual web servers sharing the same web server process.

the Wednesday 7
method &
asks for the
resource named/
index.html
be returned
using the 8
HTTP/1.1 protocol
version.



PLANNER

December 2016

Friday
9

Saturday 10
Sunday 11

Monday
12

When the web server receives the request it looks at ~~Wednesday~~ ^{Tuesday} URI and decides, ~~now~~ ¹³ on information to handle it.

→ Header → LAST-Modified : Mon, 20 Dec 2019 23:22:42 GMT

(gives the date & time for Wednesday)

when the resource was last modified

~~14~~ & it can

may handle

it internally by simply reading the file from the system, the file is an HTML file. ~~15~~ and the component is responsible for some request to some

The server can use this to send different types of responses to different types of browsers. The User-Agent header contains information about the type of browser making the request. The server can use this to send different types of responses to different types of browsers.

The Accept headers provide information about the languages & file formats the browser accepts.

Can be used to adjust the response to the capabilities of the browser.

Responses Message It starts with the name of

→ A status line a protocol, following by a status code & a short

→ Response Header description of status code.

→ Response body.

→ Execution successful of request.

HTTP / 1.1 200 OK

Date : What type of response data

Status : 200

The body contains

Content - Type : text / html

Servlet - Engine : Tomcat Web Server / 5. 0

Content - Length : 59 → how large is the response.

The status line starts with the name of the protocol, followed by a status code of the status

and a short description of the status

code.

that responsible for the resource working of the VR1.

for the VR1.

PLANNER message
December 2016

```
<html>
<body>
<h2> Hello World </h2>
</body>
</html>
```

नेशनल इंश्योरेन्स
National Insurance

Trusted Since 1906

- 16 **URl with a query string**
 → `http://www.weather.com/tomorrow?city=Hermosa+Beach&state=CA`
- The query string starts with a question mark (?) & consists of name / value pairs separated by &. These names & values must be URL encoded. (`? = %23 F`)
- A GET request always uses a query string to send parameters values while a POST request always sends them as part of the body.

17 Saturday 17 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Monday

19 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Tuesday

20 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Wednesday

20 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Thursday

20 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Friday

20 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Saturday

20 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.

Sunday

20 A GET request, parameters can easily be saved as a bookmark, hard coded as a link, & the response cached by the browser.

→ No harm is caused if the browser sends the request again.

A POST request, can't be bookmarked easily, the browser would have to save both the URI & the request message body.

→ The browser asks the user if it's okay to send the request again.

Methods that HTTP specifies:-

1) OPTIONS → It is used to find out what options a server on a resource offers.

2) HEAD → It is used to get a response with all headers generated by a GET request but without the body.

3) PUT → It is used to store the message body content on the server as a resource identified by the URI.

21 4) DELETE → It is used to delete the resource identified by the URI.



PLANNER

December 2016

Friday

5) TRACE → It is used for testing the communication b/w the client & the server.

23

SERVLETS

Saturday 24

It is a piece of code that adds a new functionality to a server just like CGI. A servlet based solution doesn't belong to one specific vendor.

Sunday 25

Advantages :- → Platform & Vendor independence programming language so can be used on any OS with Java runtime environment.

Monday

26

Servlets are developed in Java & can take advantage of all other Java technologies.

Tuesday

27

Servlet execute in a process that is running until the servlet based app is down. Each servlet request is executed as a separate thread in this permanent process. Servlets can also access resources that remain loaded in the process memory b/w requests.

→ Scalability

→ Robustness & security typed programming language

Servlets use special interfaces to server resources that cannot be shared.

28

Container

Connection b/w a Web server & the servlets is servlet container.

It provides the run time environment for all the servlets on the server. It is responsible for loading & invoking those servlets.

Vulnerable to attacks.

29

The container loads a servlet class when PLANNER it receives the first request for the December 2016-January 2017, gives it a chance to initialize itself, then asks Plugins it to process request.



The container Add-ons containers :- These are used to add new features to web servers without native servlet support. They can run in the same OS process as the Web server in a separate process.

Sunday resources¹

& save its state

Mondays

Standalone servers - It includes web server functionality to provide full support for HTTP.

2

The servlet container is responsible for mapping an incoming request to a registered servlet to handle the resource identified by the URI & passing the request message to that servlet.

3

After the request is processed, container converts the response created by the servlet into an HTTP response message & send it back to the client.

4

Container support a standard portable way to package all these resources, along with a web application deployment descriptor containing information about how all the resources fit together.

5

The deployment descriptor containing information about how all the

Tuesday

Wednesday

Thursday

Date :



The deployment descriptor & all the other web application files are arranged in a well-defined hierarchy within an archive file, called a WAR (web application archive)

Within the container, each web application is represented by a servlet context.

It is associated with a unique URI path prefix called the context path.

Each context is self-contained & does not know anything about other applications running in the same container.

References between the servlets & JSP pages in the applications are commonly relative to the context path & are referred to as context-relative paths.

3 important parts of the servlet :-

(1) Web application structure.

(2) The deployment file format.

(3) Ability to share objects among components in an application.

January							February							March							April							May							June						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S							
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7				
6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7									

PARIVAR

Ch-3

Date :

JSP Overview

A JSP page is a web page with JSP elements for generating the part that differ for each request.

Everything in the page that isn't a JSP element is called template text.
HTML, WML, XML

When a JSP page request is processed, the template text and dynamic content generated by the JSP elements are merged, the result is sent as the response to the browser.

The pure servlet based approach have a few problems:-

(1) Changing the look or feel, requires the servlet code to be updated & recompiled.

Sol: Separate the request processing and business logic code from presentation.

The server needs a JSP container to process JSP pages.

The JSP container is responsible for intercepting requests for JSP pages.

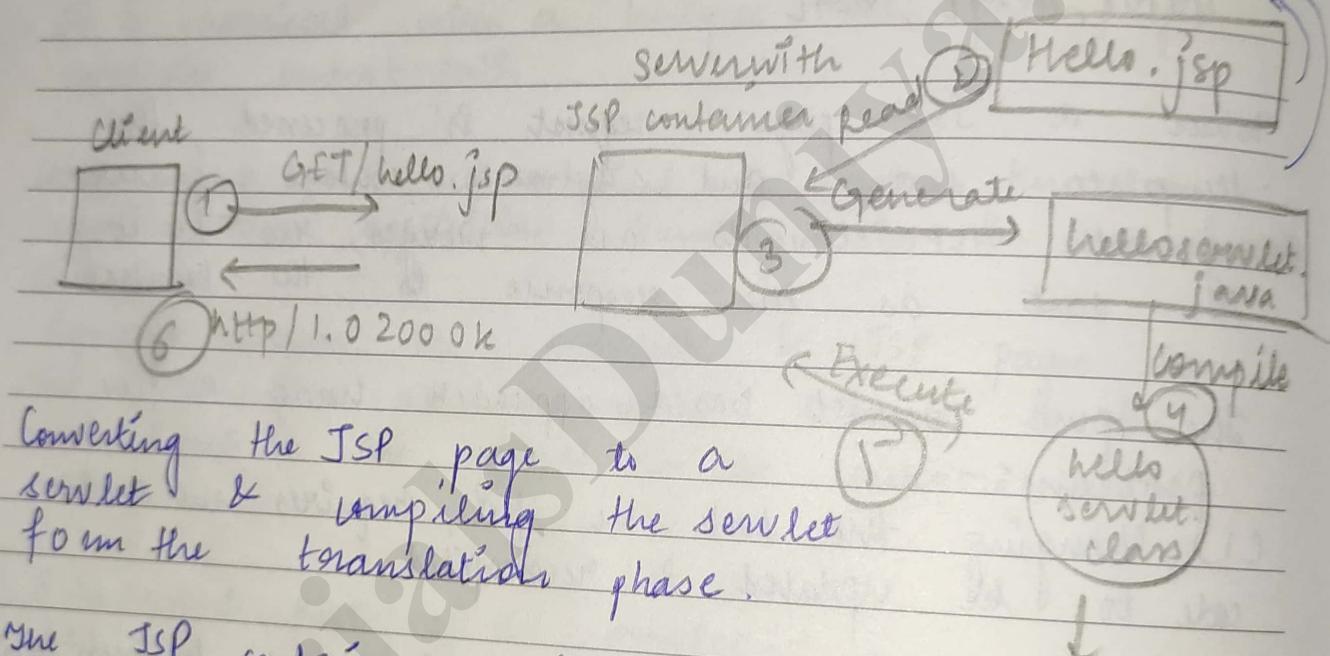
July	August	September	October	November	December									
S	M	T	W	T	F	S	S	M	T	W	T	F	S	
31		1 2	1 2 3 4 5 6	1 2 3	1		1	2 3	1 2 3 4 5	1 2 3	1 2 3	1 2 3		
3 4 5 6 7 8 9	7 8 9 10 11 12 13	4 5 6 7 8 9 10	2 3 4 5 6 7 8	6 7 8 9 10 11 12	13 14 15 16 17 18 19	20 21 22 23 24 25 26	25 26 27 28 29 30	1 2 3 4 5	6 7 8 9 10 11 12	13 14 15 16 17 18 19	20 21 22 23 24 25 26	27 28 29 30	25 26 27 28 29 30 31	2016
10 11 12 13 14 15 16	14 15 16 17 18 19 20	11 12 13 14 15 16 17	9 10 11 12 13 14 15	16 17 18 19 20 21 22	23 24 25 26 27 28 29	23 24 25 26 27 28 29								
17 18 19 20 21 22 23	21 22 23 24 25 26 27	18 19 20 21 22 23 24	16 17 18 19 20 21 22	27 28 29 30										
24 25 26 27 28 29 30	28 29 30 31	25 26 27 28 29 30												

Date :



To process all JSP elements in the page, the container first turns the JSP page into a servlet (JSP page implementation). All templates and text is converted to printout statements & all JSP elements are converted to Java code that implements the dynamic behaviour. The container then compiles the servlet class.

Translation phase



Converting the JSP page to a servlet & compiling the servlet from the translation phase.

The JSP container initiates the Request processing phase automatically when it receives the first request for a page. This occurs during the translation phase.

Prescompilation of a JSP page: Initiation of the translation phase

January		February		March		April		May		June	
S	M	S	M	S	M	S	M	S	M	S	M
2	3	4	5	6	7	8	9	10	11	12	13
0	10	11	12	13	14	15	16	17	18	19	20
1	17	18	19	20	21	22	23	24	25	26	27
6	24	25	26	27	28	29	30	31	1	2	3

PERSONAL ACCIDENT POLICY - For individuals/family



Date :

Decompilation of a JSP page is a way to avoid hitting the first user with this delay.

The JSP container is also responsible for invoking the JSP page implementation class to process each request & generate the response. This is called the request processing phase.

As long as the JSP page remains unchanged, any request goes straight to the request processing phase.

When the JSP page is modified, it goes through the translation phase again before entering the request processing phase.

JSP container + servlet container = web container

JSP elements

- directive
- action
- scripting
- Expression language

July							August							September							October							November							December									
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S										
31					1	2	1	2	3	4	5	6	7	1	2	3	4	5	6	30	31	1	2	3	4	5	1	2	3	4	5	6	7	8	9	10								
10	11	12	13	14	15	16	7	8	9	10	11	12	13	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
17	18	19	20	21	22	23	14	15	16	17	18	19	20	11	12	13	14	15	16	17	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
24	25	26	27	28	29	30	21	22	23	24	25	26	27	28	29	30	31	25	26	27	28	29	30	23	24	25	26	27	28	29	30	31	20	21	22	23	24	25	26	27	28	29	30	31

MOTOR INSURANCE - Covers loss to vehicle & third party.

TUTORIALSDUNIYA.COM

Get FREE Compiled Books, Notes, Programs, Question Papers with Solution etc of the following subjects at <https://www.tutorialsduniya.com>

- C and C++
- Java
- Data Structures
- Computer Networks
- Android Programming
- PHP Programming
- JavaScript
- Java Server Pages
- Python
- Microprocessor
- Artificial Intelligence
- Machine Learning
- Computer System Architecture
- Discrete Structures
- Operating Systems
- Algorithms
- DataBase Management Systems
- Software Engineering
- Theory of Computation
- Operational Research
- System Programming
- Data Mining
- Computer Graphics
- Data Science

-
- ❖ DU Programs: <https://www.tutorialsduniya.com/programs>
 - ❖ TutorialsDuniya App: <http://bit.ly/TutorialsDuniyaApp>
 - ❖ C++ Tutorial: <https://www.tutorialsduniya.com/cplusplus>
 - ❖ Java Tutorial: <https://www.tutorialsduniya.com/java>
 - ❖ JavaScript Tutorial: <https://www.tutorialsduniya.com/javascript>
 - ❖ Python Tutorial: <https://www.tutorialsduniya.com/python>
 - ❖ Kotlin Tutorial: <https://www.tutorialsduniya.com/kotlin>
 - ❖ JSP Tutorial: <https://www.tutorialsduniya.com/jsp>

Date :



Directive Elements

It specifies information about the page itself that remains the same b/w requests.

`<%@ page ... %>` → Define page-dependent attributes such as scripting language, error page, & buffering requirements.

`<%@ include ... %>` → Includes a file during the translation phase.

`<%@ taglib ... %>` → Declares a tag library containing custom actions used in the page.

The page directive is used to provide instructions to the container.

`<%@ page attribute = "Value" %>`
`<%@ include file = "relative url" %>`

The taglib directive declares that your JSP page uses a set of custom tags, identifies the location of other tags in your JSP means for identifying the custom tags.

January							February							March							April							May							June						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S							
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7				
6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7									

MARINE TRANSIT INSURANCE

Date :



<%@ taglib uri="uri" prefix="prefixoftag">

<%@ page contentType="text/html" %>

<%@ include file="relative uri" %>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

Standard Action Elements

It performs actions based on information that is required at the exact time the JSP page is requested by a browser.

<jsp:action-name attribute="Value"/>

jsp:include → Includes a file at the time the page is requested.

jsp:useBean → Makes a javaBeans component available in the page.

jsp:setProperty → Sets the property of a JavaBean

jsp:getProperty → Inserts the property of a JavaBean into the output.

July	August	September	October	November	December								
S	M	T	W	T	F	S	S	M	T	W	T	F	S
31			1 2	3 4	5 6	7 8	9 10	11 12	13 14	15 16	17 18	19 20	21 22
1 2	3 4	5 6	7 8	9 10	11 12	13 14	15 16	17 18	19 20	21 22	23 24	25 26	27 28
3 4	5 6	7 8	9 10	11 12	13 14	15 16	17 18	19 20	21 22	23 24	25 26	27 28	29 30
10 11	12 13	14 15	16 17	18 19	20 21	22 23	24 25	26 27	28 29	30 31	1 2	3 4	5 6
17 18	19 20	21 22	23 24	25 26	27 28	29 30	25 26	27 28	29 30	1 2	3 4	5 6	7 8
24 25	26 27	28 29	29 30	31			23 24	25 26	27 28	29 30	1 2	3 4	5 6

The Beginning of Chap.

Date :



`<jsp:forward>` - forwards the request to a new page.

`<jsp:plugin>` - Generates browser-specific code that makes an object or embed tag for (HTML) the Java plugin.

`<jsp:param>` - Adds a parameter value to a request handed off to another servlet or JSP page using `include` or `forward`.

`<jsp:include page="relative URL"/>`

Ex: `date.jsp`.

`<P>Today's date: <%=(new java.util.Date()).toLocaleString()%></p>`

`main.jsp`.

`<html>`

`<head>`

`<title> </title>`

`</head>`

`<body>`

`<jsp:include page="date.jsp" flush="true"/>`

`<jsp:forward`

`page="date.jsp"`

buffer flushed
before it is included

January						
S	M	T	W	T	F	S
2	3	4	5	6	7	8
0	10	11	12	13	14	15
1	17	18	19	20	21	22
6	24	25	26	27	28	29
	30	31				

`February`

`S M T W T F S`

`1 2 3 4 5 6 7`

`8 9 10 11 12 13`

`14 15 16 17 18 19 20`

`21 22 23 24 25 26 27`

`28 29`

`March`

`S M T W T F S`

`1 2 3 4 5 6 7`

`8 9 10 11 12 13 14`

`15 16 17 18 19 20 21`

`22 23 24 25 26 27 28`

`29 30 31`

`April`

`S M T W T F S`

`1 2 3 4 5 6 7`

`10 11 12 13 14 15 16`

`20 21 22 23 24 25 26`

`27 28 29 30 31`

`May`

`S M T W T F S`

`1 2 3 4 5 6 7`

`8 9 10 11 12 13 14`

`15 16 17 18 19 20 21`

`22 23 24 25 26 27 28`

`June`

`S M T W T F S`

`1 2 3 4`

`5 6 7 8 9 10 11`

`12 13 14 15 16 17 18`

`19 20 21 22 23 24 25`

`VARISHTHA MEDICAL`

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

Date :



`<jsp:useBean> → It searches
<jsp:useBean id="name" class="package class"/>`

`<jsp:setProperty name="name" property="someproperty"/>
</useBean>
jsp:`

`<jsp:getProperty name="name" property="" />`

for testbean.java

```
public class TestBean {
    private String message = "No message specified";
    public String getMessage() {
        return message;
    }
}
```

`public void setMessage(String message)`

`this.message = message;
}`

main.jsp

`<jsp:useBean id="test" class="action.TestBean"/>`

`<jsp:setProperty name="test" property="message"
 value="Hello JSP"/>`

`<jsp:getProperty name="test" property="message"/>`

July						
S	M	T	W	T	F	S
31			1	2		
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

August						
S	M	T	W	T	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

September						
S	M	T	W	T	F	S
			1	2	3	
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

October						
S	M	T	W	T	F	S
30	31				1	
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

November						
S	M	T	W	T	F	S
			1	2	3	4
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

December						
S	M	T	W	T	F	S
			1	2	3	
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

OFFICE PACKAGE INSURANCE - Multi risks cover suitable for office operations.

2
0
1
6

Date :



Trusted Since 1906

The plugin action is used to insert java components into a JSP page.

```
<jsp:plugin type="applet" >
<jsp:param name="fontcolor" value="red"/>
```

Scripting Elements

It allows to add small pieces of code in a JSP page.

`<% ... %>` → Used Scriptlet - Used to embed scripting code
 ↳ comment

`<%= %>` → Expression - Used to embed scripting code expressions when the new `java.util.Date()` result shall be added to the response.

`<%! - %>` → Declaration, used to declare instance variables & method.
`<%! int i=0; %>`

EL expressions

It is a simple language for accessing request through application classes & data made available for dates & data made available for

January							February							March							April							May							June																		
2	S	M	T	W	T	F	S	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31															
0	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																	
1	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
6	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31							

POLYU

Date :



A JavaBeans component is a Java class that complies with certain coding conventions.

MVC (Model - View - Controller)

The key point of using MVC is to separate logic into three (distinct) units.

→ Model → business logic & data

→ View → presentation

→ controller → Request processing



Ch-5 Generating Dynamic Content Date :

.jsp → it tells the server that the page needs to be processed by the JSP container.

```
Ex: <%@ page contentType="text/html" %>
<% taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>
<head>
<body>
<h1> JSP is easy </h1>
<%          Comment           %>
    1+2+3 = <c:out value="${1+2+3}" />
</body>
</html>
```

- XML → It contains information about security requirement, how all the resources fit together & other facts about the applications.

`.war` → create (far command on ZIP program)
`/index.html`

/index.html

I cover gift
Pch5 easy. isp

/WEB-INF/web.xml

Java classes / JSP SourceServlet.

application " / lib / orataglib-3.0.jar
deployment descriptor

January							February						
S	M	T	W	T	F	S	S	M	T	W	T	F	S
31				1	2			1	2	3	4	5	6
3	4	5	6	7	8	9		7	8	9	10	11	12
10	11	12	13	14	15	16		14	15	16	17	18	19
17	18	19	20	21	22	23		21	22	23	24	25	26
24	25	26	27	28	29	30		28	29				

S	M	T	W	T	F	S
			1	2	3	4
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

April						
S	M	T	W	T	F	S
3	4	5	6	7	8	1
10	11	12	13	14	15	9
17	18	19	20	21	22	16
24	25	26	27	28	29	23

May						
S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

June						
S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25

Date :



lib & classes (All app. class files are stored here)
JAR
compressed archives of Java class files

Page directive specifies the MIME type for the content the page produces
text/plain → preformatted plain text.

Directives - Translation phase

Action - sumti me

Action elements are also called tag libraries.

`<prefix:action-name attr1="value1"></prefix:action-name>`

- 1) It makes it possible for actions in different libraries to have the same name.
 - 2) It makes it possible for the container to figure out to which library a specific action belongs.

PROJECT INSURANCE - A policy designed for projects risks.



The taglib directive's uri attribute is a unique identifier for the tag library, which the container uses to find the information it needs to process the action.

Actions

Standard Custom

These actions are the actions defined by the JSP specification itself.

<jsp:useBean>

Core

XML Internationalization

<jsp:getProperty>

Conditional

Processing - alization

<jsp:setProperty>

Processing &

Processing & for-

<jsp:include>

looping,

access

<jsp:forward>

importing

making (SQL)

<jsp:plugin>

data from

data, such as

<jsp:param>

external

format Read &

<jsp:attribute>

sources.

as transform- & parse

Defines dynamically defined XML element's attribute.

accessing

localized

information, insert

individual elements.

localized

information

in a page.

* Functions - Generic Expression Language functions

January		February		March		April		May		June	
S	M	S	M	S	M	S	M	S	M	S	
2	3	4	5	6	7	8	9	10	11	12	13
0	1	2	3	4	5	6	7	8	9	10	11
1	10	11	12	13	14	15	16	17	18	19	20
6	17	18	19	20	21	22	23	24	25	26	27
24	25	26	27	28	29	28	29	30	31	27	28
29	30	31									

SPECIAL CONTINGENCY POLICY - You name the risk, we design the policy.

Date :



Sets the value of an action attribute based on body of this element.

<jsp:body> → Sets the action element body on the body of this element.

<jsp:element> → Dynamically generates an XML element.

<jsp:text> Used to write template text in JSP pages & documents.

<body>

<jsp:element name = "XmlElement">

<jsp:attribute name = "XmlElement Attr">

Value for the attribute

</jsp:attribute>

<jsp:body>

Body for XML element

</jsp:body>

</jsp:element>

XML elements can be generated at request time.

<jsp:text> Temp date data </jsp:text>

July						
S	M	T	W	T	F	S
31		1	2			
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

August						
S	M	T	W	T	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

September						
S	M	T	W	T	F	S
		1	2	3		
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

October						
S	M	T	W	T	F	S
30	31		1	2	3	4
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

November						
S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

December						
S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

JANATA PERSONAL ACCIDENT - An accidental insurance for Low income group.

Date :



The JSP Expression Language.

$\$ \{ \}$

↳ literals, variable, variables representing application data, addition, -, *, /, %.

$$1+2+3 = \$ \{ 1+2+3 \}$$

EL supports literal no.'s, Booleans, strings & null.

., [], (), ?: , +, - , * , / or div , % or mod
 ↓ ↓ ↓ ↓ conditional = = or eq , != or ne
 Access array subexpression test < or lt , > or gt
 a property = or le , >= or ge , && or and , || or or
 | or not , empty , functioning)

empty variable values

EL can also contain variables.

Variables are named references to data (objects), created by the application or made available implicitly by the EL.

page scope

page scope - A collection of all page scope variable.
 requestScope = " " of all request - scope variable.

January		February		March		April		May		June	
S	M	T	W	T	F	S	S	M	T	W	F
2	3	4	5	6	7	8	9	10	11	12	13
0	1	2	3	4	5	6	7	8	9	10	11
1	10	11	12	13	14	15	16	13	14	15	16
6	17	18	19	20	21	22	23	20	21	22	23
24	25	26	27	28	29	30	27	28	29	30	31

GROUP MEDICLAIM INSURANCE - For employees of an Industry / organization.

Date :



Trusted Since 1906

sessionScope - " " " " session " " " "
 applicationScope - " " " " application " " "
 param - " " " " " request parameter values
 as a single String value per parameter.

paramValues - " " " " " " " " " "
 " " " array per parameter.

headers - A collection of all request header values
 as a single String value per header.

headerValues - " " " " " " " " "
 " " " String array " " " .

cookies - " " " " " cookie " " " " "
 javax.servlet.http.Cookie value per cookie.

initParam - A collection of all application initialization
 parameters values as a single String value per
 value.

pageContext - An instance of the javax.servlet.jsp.Page
 context class, providing access to various
 request data.

July							August							September							October							November							December																													
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S																														
31				1	2		1	2	3	4	5	6		1	2	3	30	31	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3																															
1	2	3	4	5	6	7	7	8	9	10	11	12	13	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																							
2	3	4	5	6	7	8	14	15	16	17	18	19	20	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3																											
3	4	5	6	7	8	9	14	15	16	17	18	19	20	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31							
4	5	6	7	8	9	10	14	15	16	17	18	19	20	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
5	6	7	8	9	10	11	14	15	16	17	18	19	20	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
6	7	8	9	10	11	12	15	16	17	18	19	20	21	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
7	8	9	10	11	12	13	16	17	18	19	20	21	22	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
8	9	10	11	12	13	14	17	18	19	20	21	22	23	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
9	10	11	12	13	14	15	18	19	20	21	22	23	24	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31					
10	11	12	13	14	15	16	19	20	21	22	23	24	25	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31						



Date :

```
<c:out value="${param.username}" />  
<c:out value="${param['user-name']}" />
```

Ch-6 USING JAVA BEANS COMPONENTS IN JSP PAGE

A JavaBeans component, or just a Bean is often used in JSP as the container for the dynamic content to be displayed by a web page.

When JSP is combined with servlets, the bean can be created & initialized with data by the servlet & passed to a JSP page that simply adds the bean's data to the response.

A bean is a java class that follows certain coding conventions, so it can be used by tools as a component in a larger application.

8
ear

- Import package (package com . org . jsp . beans . mofd)
- NO argument constructor

Date :



- A property is read-only, write-only or read/write.
- Bean class should implement java.io.Serializable or Externalizable to allow a tool to save & restore the bean's state.

Advantages of bean :-

- (1) It can encapsulate all information about the item it represents in one simple package.
- (2) The bean can encapsulate all the rules about its properties.

<jsp: useBean id="cartoon" class="com.ora.jsp.beans.mtd.CartoonBean" />
id, scope (page, request, session, application), class type, beanName.

<jsp: getProperty name="cartoon" property="fileName"/>
It obtains the current value of a bean property and inserts it directly into the response body.

<jsp: setProperty name="msg" property="category">
<jsp: attribute name="value" trim="true" />
<jsp: getProperty name="myBean" property="myProperty"/>

July		August		September		October		November		December	
S	M	S	M	S	M	S	S	S	M	T	F
31		1	2	1	2	3	4	5	6	7	8
3	4	5	6	7	8	9	10	11	12	13	14
10	11	12	13	14	15	16	17	18	19	20	21
17	18	19	20	21	22	23	24	25	26	27	28
24	25	26	27	28	29	30	31				

A better way of safeguarding one's home.											
S	M	T	W	T	F	S	S	1	2	3	4
30	31		1	2	3			1	2	3	5
		2	3	4	5	6	7	6	7	8	9
		9	10	11	12	13	14	13	14	15	16
		16	17	18	19	20	21	20	21	22	23
		23	24	25	26	27	28	27	28	29	30

Date :



`</jsp:attribute>` → This tag action can only be used in the body of a JSP element. It evaluates its body & sets the attribute named by the mandatory name attribute of its parent action element to the output produced by the nested JSP elements.

→ Using the JSP Expression Language
The EL also supports access to bean properties.

```
<jsp:useBean id="cartoon" class="" />

```

SETTING BEAN PROPERTIES

(1) Using the `<jsp:setProperty>` action

```
<jsp:useBean id="name" class="" />
<jsp:setProperty name="msg" property="category" />
```

→ It supports param (used to set properties)

(2) Using the JSTL

`<c:set>` Action (flexible more)
as a static string

```
<c:set target="${msg}" />
```

property = "category"
value = "thoughts" />

January							February							March							April							May							June						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S							
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7				
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7					
3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7						
4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7							
5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7								
6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7									
7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7										
8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7											
9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7												
10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7													
11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7														
12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7															
13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																	
15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																		
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																			
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																				
18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																					
19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																						
20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																							
21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																								
22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																									
23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																										
24	25	26	27	28	29	30	31	1	2	3	4	5	6	7																											
25	26	27	28	29	30	31	1	2	3	4	5	6	7																												
26	27	28	29	30	31	1	2	3	4	5	6	7																													
27	28	29	30	31	1	2	3	4	5	6	7																														
28	29	30	31	1	2	3	4	5	6	7																															
29	30	31	1	2	3	4	5	6	7																																
30	31	1	2	3	4	5	6	7																																	
31	1	2	3	4	5	6	7																																		

PERSONAL ACCIDENT POLICY

Date :



Automatic Type Conversions
Boolean. valueOf (String)

Byte.

String. charAt (0)

Double. valueOf (String)

Integer

Float

Long

Short

new String (String)

Property editor associated with a bean can convert the string to a date object.

The value returned by `<@sp: getProperty>`, all EL expression directly in template text, or `<@c:out>` is always converted to a String - no matter what Java type it represents.

July	August	September	October	November	December
S	S	S	S	S	S
M	M	M	M	M	M
S 31 M 1 T 2 W 3 T 4 F 5 S 6	S 1 M 2 T 3 W 4 T 5 F 6 S 7	S 4 M 5 T 6 W 7 T 8 F 9 S 10	S 1 M 2 T 3 W 4 T 5 F 6 S 7	S 1 M 2 T 3 W 4 T 5 F 6 S 7	S 1 M 2 T 3 W 4 T 5 F 6 S 7
J 7 A 8 U 9 S 10 E 11 P 12 R 13 O 14 C 15 N 16 D 17 V 18 B 19 G 20 H 21 Y 22 F 23 J 24 L 25 K 26 X 27 Z 28 M 29 W 30	A 7 S 8 U 9 S 10 E 11 P 12 R 13 O 14 C 15 N 16 D 17 V 18 B 19 G 20 H 21 Y 22 F 23 J 24 L 25 K 26 X 27 Z 28 M 29 W 30	S 11 M 12 T 13 W 14 T 15 F 16 S 17 O 18 C 19 B 20 R 21 G 22 H 23 Y 24 F 25 J 26 L 27 K 28 X 29 Z 30	O 1 C 2 B 3 R 4 G 5 H 6 Y 7 F 8 J 9 L 10 K 11 X 12 Z 13 M 14 W 15 C 16 B 17 R 18 G 19 H 20 Y 21 F 22 J 23 L 24 K 25 X 26 Z 27 M 28 W 29 C 30	N 13 D 14 C 15 R 16 G 17 H 18 Y 19 F 20 J 21 L 22 K 23 X 24 Z 25 M 26 W 27 C 28 B 29 R 30	D 20 C 21 R 22 G 23 H 24 Y 25 F 26 J 27 L 28 K 29 X 30

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 