



NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

TOURIST COMPANY WEB PAGE DESIGN

Graduation Project
COM-400

Student: Mohammad Al-Sheyab (20034267)

Supervisor: Assist. Prof. Dr. Firudin Muradov

Nicosia - 2007

ACKNOWLEDGMENTS

First of all I am happy to complete the task which I had given with blessing of God and also I am grateful to all the people in my life who have, supported me, advised me, taught me and who have always encouraged me to follow my dreams and ambitions. My dearest parents, my brothers and sisters, my friends and my tutors. They have taught me that no dream is unachievable. As in the words of Walt Disney "If you can dream it, you can do it".

I wish to thank my advisor, *Assist. Prof. Dr. Firudin Muradov*, for intellectual support, encouragement, and enthusiasm, which made this project possible, and his patience for correcting both my stylistic and scientific errors.

My sincerest thanks must go to my friends, Salem Almarashdeh, Ahmad Kassim, Alaa Abu hadba, Ahmad Abu shhab, Ayman Abu khurma, Juma Alkatib, and all friends who shared their suggestions and evaluations throughout the completion of my project. The comments from these friends enabled me to present this project successfully.

Finally, I wish by this project to be useful for all students, especially Computer Engineering to support our improvement.

And above, I thank God for giving me stamina and courage to achieve my objectives.

To all of them, my love and respect

ABSTRACT

With the increased popularity of the Internet, computer technologies are receiving more and more attention. The Web is a hypertext system that runs over the Internet as one of its services. The World Wide Web has opened a whole new arena for information systems, distance learning courses and the access to remote resources. The Web can be used to enhance education through remote access to resources or experts, or it can be used for commercial purposes.

The aim of the project is to design a website for a Tourist Company. Users can work on any computer, browse documents and share their views on related topics.

The following techniques are used in the design of the Web site:

HTML is basic language for design the website,

JavaScript it used to make some activate or to validate the data,

Database is used to save or represent data from or to website,

ASP is used to make a connection between the pages or between the pages and

Database.

TABLE OF CONTENTS

AKNOWLEDGMENT	i
ABSTRACT	ii
TABLE OF CONTENTS	iii
INTRODUCTIONS	viii
CHAPTER ONE: HYPER TEXT MARKUP LANGUAGE	1
1.1 Introduction	1
1.1.1 What is an HTML file?	1
1.2 HTML And The World Wide Web	1
1.2.1 What is the World Wide Web?	1
1.2.2 How does the WWW work?	1
1.2.3 How does the browser fetch the pages?	1
1.2.4 How does the browser display the pages?	2
1.3 HTML Elements	2
1.3.1 HTML Tags	2
1.3.2 Tag Attributes	2
1.4 Basic HTML Tags	3
1.4.1 Headings	3
1.4.2 Paragraphs	3
1.4.3 Line Breaks	4
1.5 HTML Character Entities	4
1.5.1 Non-breaking Space	4
1.5.2 The Most Common Character Entities:	4
1.6 HTML Links	5
1.6.1 The Href Attribute	5
1.6.2 The Target Attribute	5
1.7 HTML Tables	5
1.7.1 Table attribute	6
1.7.1.1 The Border Attribute	6
1.7.1.2 The align attribute	6

1.7.1.3 The width attributes	7
1.8 HTML Forms and Input	7
1.8.1 Forms	7
1.8.1.1 Type of method	7
1.8.1.2 Action	7
1.8.2 Input	7
1.8.2.1 Text Fields	8
1.8.2.2 Radio Button	8
1.8.2.3 Submit Button	9
1.10 Summary	9
CHAPTER TWO: ACTIVE SERVER PAGE	10
2.1 Introduction	10
2.1.1 What is ASP?	10
2.1.2 ASP Compatibility	10
2.1.3 What is an ASP File?	10
2.1.4 How Does ASP Differ from HTML?	11
2.2 Setting ASP Environments	11
2.3 HTML Compared to ASP	11
2.4 ASP Processing	13
2.5 ASP Syntax	14
2.5.1 The Basic Syntax Rule	14
2.5.2 The Response Object	14
2.5.3 VBScript	14
2.6 ASP Variables	15
2.6.1 Lifetime of Variables	15
2.6.2 Session Variables	15
2.6.3 Application Variables	16
2.7 ASP Forms and User Input	16
2.7.1 User Input	16
2.7.2 Request.QueryString	16
2.7.3 Request.Form	17

2.7.4 Form Validation	17
2.8 ADO Database Connection	18
2.8.1 Create a DSN-less Database Connection	18
2.8.2 Create an ODBC Database Connection	18
2.8.3 Create an ODBC SQL	19
2.8.3.1 The Insert Operation	19
2.8.3.2 The Delete Operation	20
2.8.3.3 The Search Operation	20
2.8.3.4 The Update Operation	21
CHAPTER 3. JAVASCRIPT	22
3.1 Introduction	22
3.1.1 What is JavaScript?	22
3.1.2 Are Java and JavaScript the same?	22
3.1.3 What can a JavaScript Do?	22
3.2 JavaScript How TO?	23
3.2.1 How to Put a JavaScript Into an HTML Page	23
3.2.2 How to Handle Older Browsers	23
3.3 JavaScript Where To	24
3.3.1 Where to put the JavaScript	24
3.3.1.1 Scripts in the head section	24
3.3.1.2 Scripts in the body section	25
3.3.1.3 Scripts in both the body and the head section	25
3.4 JavaScript Variables	25
3.4.1 Declare a Variable	26
3.4.2 Assign a Value to a Variable	26
3.5 JavaScript Functions	26
3.5.1 How to Define a Function	26
3.5.2 How to Call a Function	27
3.5.3 The return Statement	27
3.6 JavaScript Conditional Statements	27
3.6.1 If and IL.else Statement	28
3.6.1.1 Without ELSE	28

3.6.1.2 With ELSE	28
3.6.2 Conditional Operator	29
3.7 JavaScript Looping	29
3.7.1 For Loop	29
CHAPTER 4. DATABASE. MICROSOFT ACCESS	
DATABASE.	31
4.1 Introduction	31
4.1.1 What is a database?	31
4.1.2 Database Management System (DBMS)	31
4.1.3 Database Design	32
4.2 Microsoft Access	32
4.3 Relations Between Tables	33
4.4 Tables	33
4.4.1 Message	33
4.4.2 Reservation	35
4.4.3 Offers	37
4.4.4 OfferReservation	38
CHAPTER 5. TOURIST COPMANY WEB PAGE DESIGN	40
5.1 Introduction	40
5.2 First Page	42
5.2.1 Project. Html	42
5.3 User Message	44
5.3.1 Search. Asp	44
5.3.2 Activate. Asp	46
5.4 Collecting Registration Details	47
5.4.1 Register. Asp	47
5.4.2 Function of JavaScript (VerifyData)	50
5.4.3 Receive.Asp	51
5.4.4 Exhibit. Asp	52
5.4.5 Offer. Asp	56
5.4.6 Offers. Asp	57

5.5 Manager Site	59
5.5.1 Active User Name and Password	59
5.5.1.1 User. Asp	59
5.5.2 Select. Asp	60
5.5.3 Price. Asp	60
5.5.4 Searching Procedure	62
5.5.4.1 Num. Asp	62
5.5.4.2 Find. Asp	63
5.5.5 Deleting Procedure	66
5.5.5.1 Delete. Asp	66
5.5.6 Updating Procedure	67
5.5.6.1 Modify. Asp	67
5.5.6.2 Mod. Asp	67
5.6 About Jordan	68
5.7 Cities	68
CONCLUSION	69
REFERENCES	70

INTRODUCTION

The Hyper Text Markup Language (HTML), Active server page (ASP), and Active Data Objects (ADO) are the main - field of Web designing.

The World Wide Web (www) work by Web pages, Web information is stored in documents called My Project.

If you want to do run for the program on your computer without connection to internet you have to install the Internet Information Service (IIS) and determined the home directory to run the program.

JavaScript was designed to add interactivity to HTML page, and the JavaScript is used in the web page to improve the design, validate forms or data, and much more. JavaScript was developed by Netscape and is the most popular scripting language on the Internet.

The ADO Connection Object is used to create an open connection to a data source. Through this connection, you can access and manipulate a database.

Chapter One: describes what the World Wide Web, the difference between a Web browser and a Web server, also what HTML, how HTML "markup tags" are used to format a Web page and gives you some basic guidelines for using HTML tags (paragraph, heading, image, forms, and tables, etc...).

Chapter Two: describes an Active Server Page (ASP) fundamental and has an explanation about ASP files and how ASP works, also this chapter describes the relation between ASP and HTML, ASP variable which brief explanation about Session variables and Application variables and the connection between ASP pages and Database..

Chapter Three: introduces JavaScript discusses some of the fundamental concepts of JavaScript, How to Put a JavaScript into an HTML Page, Where to put the JavaScript in the head section or in the body section, and provides basic examples.

Chapter Four: describes the database and database management system (DBMS), and also we have covered some of the basics for a Microsoft Access database and database objects.

Chapter Five: is the last chapter of the project, to describe the web page techniques, and how we can run ASP on the computer (machine).

CHAPTER 1. HYPER TEXT MARKUP LANGUAGE (HTML)

1.1 Introduction

Hyper Text Markup Language (HTML) used to create web pages. Every web page has a basic structure; there are titles, headings, paragraphs, graphics and maybe even an itemized list of entries.

HTML uses a defined set of standards. All you need to do is use your favorite web editor and create a document with these labeled elements.

1.1.1 What is an HTML File?

- HTML stands for Hyper Text Markup Language.
- An HTML file is a text file containing small markup tags.
- The markup tags tell the Web browser how to display the page.
- An HTML file must have an htm or html file extension.
- An HTML file can be created using a simple text editor.

1.2 HTML And The World Wide Web

1.2.1 What is the World Wide Web?

- The World Wide Web (WWW) is most often called the Web.
- The Web is a network of computers all over the world.
- All the computers in the Web can communicate with each other.
- All the computers use a communication standard called HTTP.

1.2.2 How Does the WWW Work?

- Web information is stored in documents called Web pages.
- Web pages are files stored on computers called Web servers.
- Computers reading the Web pages are called Web clients.
- Web clients view the pages with a program called a Web browser.

1.2.3 How Does the Browser Fetch the Pages?

- A browser fetches a Web page from a server by a request.
- A request is a standard HTTP request containing a page address.

1.2.4 How Does the Browser Display the Pages?

- All Web pages contain instructions for display.
- The browser displays the page by reading these instructions.
- The most common display instructions are called HTML tags.
- HTML tags look like this `<p>This is a Paragraph-c/p>`.

1.3 HTML Elements

HTML documents are text files made up of HTML elements and HTML elements are defined using HTML tags.

1.3.1 HTML Tags

- HTML tags are used to mark-up HTML elements.
- HTML tags are surrounded by the two characters « and ».
- The surrounding characters are called angle brackets.
- HTML tags normally come in pairs like `` and ``.
- The first tag in a pair is the start tag; the second tag is the end tag.
- The text between the start and end tags is the element content.
- HTML tags are not case sensitive; `` means the same as ``.

1.3.2 Tag Attributes

Tags can have attributes. Attributes can provide additional information about the HTML elements on your page.

This tag defines the body element of your HTML page: `<body>`. With an added `bgcolor` attribute, you can tell the browser that the background color of your page should be red, like this: `<body bgcolor="red">`.

This tag defines an HTML table: `<table>`. With an added `border` attribute, you can tell the browser that the table should have no borders: `<table border="0">`

- Attributes always come in name/value pairs like this: `name="value"`.
- Attributes are always added to the start tag of an HTML element.

1.4 Basic HTML Tags

The important tags in HTML are tags that define headings, paragraphs and line breaks.

Tag	Description
<html>	Defines an HTML document
<body>	Defines the document's body
<h1> To <h6>	Defines header 1 to header 6
<p>	Defines a paragraph
 	Inserts a single line break
<hr>	Defines a horizontal rule

1.4.1 Headings

Headings are defined with the <h1> to <h6> tags. <h1> defines the largest heading. <h6> defines the smallest heading.

```
<h1> This is a heading</h1>  
<h2> This is a heading</h2>  
<h3> This is a heading</h3>  
<h4> This is a heading</h4>  
<h5> This is a heading</h5>  
<h6> This is a heading</h6>
```

HTML automatically adds an extra blank line before and after a heading.

1.4.2 Paragraphs

Paragraphs are defined with the <p> tag.

```
<p> This is a paragraph</p>  
<p> This is another paragraph</p>
```

HTML automatically adds an extra blank line before and after a paragraph.

1.4.3 Line Breaks

The `
` tag is used when you want to end a line, but don't want to start a new paragraph. The `
` tag forces a line break wherever you place it.

The `
` tag is an empty tag. It has no closing tag.

1.5 HTML Character Entities

Some characters have a special meaning in HTML, like the less than sign (`<`) that defines the start of an HTML tag. If we want the browser to actually display these characters we must insert character entities in the HTML source.

A character entity has three parts: an ampersand(`&`), an entity name or`#` and an entity number, and finally a semicolon(`;`).

To display a less than sign in an HTML document we must write: `<` or `<`;

The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good in almost all browsers.

Note: that the entities are case sensitive.

1.5.1 Non-Breaking Space

The most common character entity in HTML is the non-breaking space. Normally HTML will truncate spaces in your text. If you write 10 spaces in your text HTML will remove 9 of them. To add spaces to your text, use the ` ` character entity.

1.5.2 The Most Common Character Entities:

Result	Description	Entity Name	Entity Number
<code><</code>	Less than	<code>&lt;</code>	<code>&#60;</code>
<code>></code>	Greater than	<code>&gt;</code>	<code>&#62;</code>
<code>&</code>	ampersand	<code>&amp;</code>	<code>&#38;</code>

"	Quotation mark	"	"
'	apostrophe	' (does not work in IE)	'

1.6 HTML Links

HTML uses a hyperlink to link to another document on the Web. HTML Links uses the <a> (anchor) tag to create a link to another document.

An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor:

```
<a href=ur|'c-Text to be displayed-c/a>
```

1.6.1 The Href Attribute

The href attribute is used to address the document to link to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.

1.6.2 The Target Attribute

With the target attribute, you can define where the linked document will be opened.(new page, main frame ,etc).

The line below will open the document in a new browser window:

```
<a href="ur|" target="_blank"> Text to be displayed-c/a>
```

1.7 HTML Tables

Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the -cid>tag).

The letters td stands for "table data," which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

```
<table>
<tr>
<td>row 1, cell 1</td>
</tr>
<tr>
<td>row 2, cell 1</td>
</tr></table>
```

1.7.1 Table Attribute

There are many attributes in the table, the most common is align, width, border, cellpadding, cellspacing etc.

1.7.1.1 The Border Attribute

If you do not specify a border attribute the table will be displayed without any borders. Sometimes this can be useful, but most of the time, you want the borders to show.

To display a table with borders, you will have to use the border attribute:

```
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
</table>
```

1.7.1.2 The Align Attribute

The align attribute is used as Center, Left, or Right to place the table on the page relative to browser.

```
<table align="center">
```

Here, will place a table on the center.

1.7.1.3 The Width Attribute

The table width is number of pixels or a percentage of the total page width.

```
<table width="100">
```

OR

```
<table width="50%">
```

1.8 HTML Forms and Input

1.8.1 Forms

Form elements are elements that allow the user to enter information (like text fields, text area fields, drop-down menus, radio buttons, checkboxes, etc.) in a form.

A form is defined with the `<form>` tag.

```
<form method="hot to send" action="URL of script">
```

```
<input>
```

```
<input>
```

```
</form>
```

1.8.1.1 Type of Method

There are two types of Methods used :

- Post: Not very popular but used for large quantity data transfers.
- Get : More popular but used to transfer a single line of text.

1.8.1.2 Action

Action is the attribute that defines the receiving URL. The receiving page must contain a script that can collect the incoming data from the sender.

1.8.2 Input

The most used form tag is the `<input>` tag. The type of input is specified with the type attribute. The most commonly used input types are explained below.

1.8.2.1 TextFields

Text fields are used when you want the user to type letters, numbers, etc. in a form.

```
<form>
Name: <input type="text" name="name">
<hr>
Surname: <input type="text" name="surname">
</form>
```

How it looks in a browser:

```
Name: |
Surname: | _ _ _ _ _
```

Note that the form itself is not visible. Also note that in most browsers, the width of the text field is 20 characters.

1.8.2.2 Radio Buttons

Radio Buttons are used when you want the user to select one of a limited number of choices.

```
<form>
<input type="radio" name="R" value="good">Very Good
<hr>
<input type="radio" name="R" value="bad"> Bad
</form>
```

How it looks in a browser:

```
E VeryGood
C Bad
```

Note that only one option can be chosen.

CHAPTER 2 ACTIVE SERVER PAGES (ASP)

2.1 Introduction

Active Server Pages (ASP) is a technology that enables the development of dynamic web sites. ASP was developed by Microsoft to allow server side development. ASP files are HTML files with special tags containing source code that provide the dynamic content. An ASP file can contain text, HTML tags and scripts. Scripts in an ASP file are executed on the server.

2.1.1 What is ASP?

- ASP stands for Active Server Pages.
- ASP is a program that runs inside IIS.
- IIS stands for Internet Information Services.
- IIS comes as a free component with Windows 2000.
- IIS is also a part of the Windows NT 4.0 and XP Option Pack.
- The Option Pack can be downloaded from Microsoft.
- PWS is a smaller - but fully functional - version of IIS.
- PWS can be found on your Windows 95/98 CD.

2.1.2 ASP Compatibility

- ASP is a Microsoft Technology.
- To run IIS you must have Windows NT 4.0 or later.
- To run PWS you must have Windows 95 or later.
- Chili ASP is a technology that runs ASP without Windows OS.
- Instant ASP is another technology that runs ASP without Windows.

2.1.3 What is an ASP File?

- An ASP file is just the same as an HTML file.
- An ASP file can contain text, HTML, XML, and scripts.
- Scripts in an ASP file are executed on the server.
- An ASP file has the file extension ".asp".

2.1.4 How Does ASP Differ From HTML?

- When a browser requests an HTML file, the server returns the file.
- When a browser requests an ASP file, IIS passes the request to the ASP engine. The ASP engine reads the ASP file, line by line, and executes the scripts in the file. Finally, the ASP file is returned to the browser as plain HTML.

2.2 Setting ASP Environments

To set-up your ASP environment you need to install your web and application server. The two popular options are Microsoft's Internet Information Server (IIS) or Personal Web Server (PWS). The first one is preferred because that is most likely to be your production environment in a real world situation. IIS requires a minimum of Windows NT 4.0 and you install it as part of NT Option Pack. PWS can be installed on Windows 98, XP and so has the advantage of letting you learn without having a server.

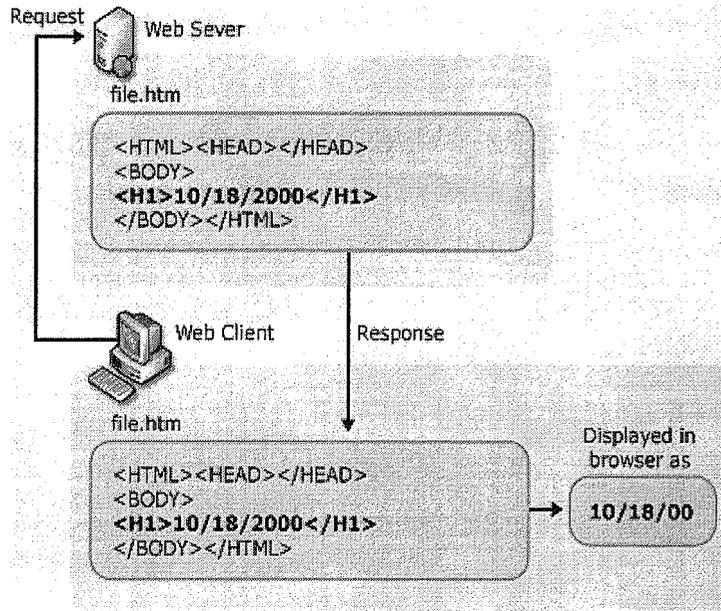
Once installed you normally create a virtual directory in your web server. In IIS you will right mouse click on the web server entry and create a virtual directory. You specify a name for your application and browse to choose a folder on your hard drive to store your source code files (e.g. C:\inetpub\wwwroot\yourappname). You can probably accept the default settings at this stage (play with the settings once you know more).

You should be ready to go. You can edit your source files with any text editor. Microsoft would recommend Visual Interdev and this colours keywords etc but there are others. These could include Homesite, Dreamweaver, Textpad, UltraEdit or even good old Notepad.

2.3 HTML Compared to ASP

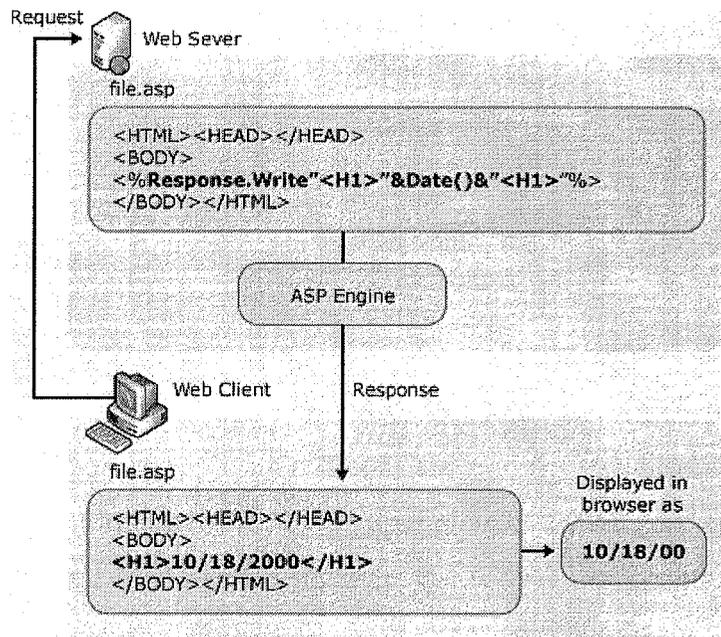
HTML is the simplest language for writing Web pages, but it allows you to create only static Web pages. When a Web client requests a static HTML file from a Web server, the Web server sends the HTML file directly to the client without any computation being done. The client's browser then processes the HTML code in the file and displays the content.

The following illustration shows the transmission of a static file where the displayed date will never change.



VBScript is the simplest language for writing ASP pages. All the code samples in the Creating ASP Pages section are written in VBScript except for samples that are duplicated in JScript for comparison. When a Web client requests an ASP file from a Web server, the Web server sends the ASP file through its ASP engine, where all the server-side script code is executed or converted into **HTML** code. The converted code is then sent to the Web client.

The following illustration shows the transmission of dynamically generated content where the displayed date reflects the date at the time of the request.



If you are an HTML author, you will find that server-side scripts written in ASP are an easy way to begin creating more complex, real-world Web applications. If you have ever wanted to store HTML form information in a database, personalize Web sites according to visitor preferences, or use different HTML features based on the browser, you will find that ASP provides a compelling solution. For example, previously, to process user input on the Web server you would have had to learn a language such as Perl or C to build a conventional Common Gateway Interface (CGI) application. With ASP, however, you can collect HTML form information and pass it to a database using simple server-side scripts embedded directly in your HTML documents. If you are already familiar with scripting languages such as VBScript or Jscript.

2.4 ASP Processing

An ASP page is requested the same way that an HTML page is requested. A request can optionally contain a query string after a question mark (?), using the following syntax:

```
http://Server_name/MyASPFile.asp?var1=Name&var2=Sumame
```

When the server receives a request for an ASP file, it processes server-side script code contained in the file to build the HTML Web page that is sent to the browser. In addition to server-side script code, ASP files can contain HTML (including related client-side scripts) as well as calls to COM components that perform a variety of tasks, such as connecting to a database or processing business logic.

IIS processes an ASP file in the following order when a request is received from a client:

1. If an ISAPI filter is installed on the Web site, the ISAPI filter is processed first. This is true for all applications.
2. If the ASP application contains a Global.asa file in the root directory, the Global.asa is processed. Global.asa files specify event scripts and declare objects that have session or application scope. They don't display content; instead they stores event information and objects used globally by the ASP application.
3. In the requested ASP file, IIS separates the script blocks from the static HTML code blocks, reserving the static code in the response body.

4. IIS processes the script blocks. The script blocks might include transaction processing, database access calls, or calls to COM components in which case COM+ handles some of the processing.
5. After the ASP page script blocks are processed, their output is injected into the response body with the static HTML code.
6. The response is sent to the client.

2.5 ASP Syntax

You cannot view the ASP source code by selecting "View source" in a browser; you will only see the output from the ASP file, which is plain HTML. This is because the scripts are executed on the server before the result is sent back to the browser.

2.5.1 The Basic Syntax Rule

An ASP file normally contains HTML tags, just like an HTML file. However, an ASP file can also contain server scripts, surrounded by the delimiters `<%` and `%>`. Server scripts are executed on the server, and can contain any expressions, statements, procedures, or operators valid for the scripting language you prefer to use.

2.5.2 The Response Object

The Write method of the ASP Response Object is used to send content to the browser. For example, the following statement sends the text "Hello World" to the browser:

```
<% response.write("Hello World!")%>
```

2.5.3 VBScript

Using VBScript on the server in an ASP page isn't very different from using it in applications or on ordinary Web pages. Nearly all of the VBScript commands are available for use on the server. VBScript commands that interact with the user, however, are not available. For example, imagine a command that opens a dialog box on the server. No one is around to dismiss it, and the system can do nothing until someone dismisses it!

The VBScript statements that present user interface elements are `InputBox` and `MsgBox`. In addition, the VBScript function `CreateObject` is replaced by a method of

the Server object. This is necessary to track the object instances on the server side. You can add comments to your script just as you normally do. However, you cannot add comments inside an output expression. An output expression is an expression or value that is evaluated and written to the Web page. It is contained within the delimiters `<%= and%>`.

However, the default scripting language is VBScript:

```
<% response.write("Hello World!")%>
```

The example above writes "Hello World!" into the body of the document.

2.6 ASP Variables

A variable is used to store information.

If the variable is declared outside a procedure it can be changed by any script in the ASP file. If the variable is declared inside a procedure, it is created and destroyed every time the procedure is executed.

2.6.1 Lifetime of Variables

A variable declared outside a procedure can be accessed and changed by any script in the ASP file.

A variable declared inside a procedure is created and destroyed every time the procedure is executed. No scripts outside the procedure can access or change the variable.

To declare variables accessible to more than one ASP file, declare them as session variables or application variables.

2.6.2 Session Variables

Session variables are used to store information about ONE single user, and are available to all pages in one application. Typically information stored in session variables are name, id, and preferences.

2.6.3 Application Variables

Application variables are also available to all pages in one application. Application variables are used to store information about ALL users in a specific application.

2.7 ASP Forms and User Input

The Request.QueryString and Request.Form commands may be used to retrieve information from forms, like user input.

2.7.1 User Input

The Request object may be used to retrieve user information from forms.

For example: from register.asp.

User input can be retrieved in two ways: With Request.QueryString or Request.Form.

2.7.2 Request.QueryString

The Request.QueryString command is used to collect values in a form with method='tget'. Information sent from a form with the GET method is visible to everyone (it will be displayed in the browser's address bar) and has limits on the amount of information to send.

If a user typed "Bill" and "Gates" in the form example above, the URL sent to the server would look like this:

```
http://www.w3schools.com/receive.asp?text1=Mohammad&text2=Alsheyab
```

Assume that the ASP file "receive.asp" contains the following script:

```
Thank you
```

```
<%
```

```
response.write(request.querystring( "text2 "))
```

```
%>
```

```
Or
```

```
<%=text2%>
```

The browser will display the following in the body of the document:

[Thank you Alsheyab

2.7.3 Request.Form

The Request.Form command is used to collect values in a form with method="post". Information sent from a form with the POST method is invisible to others and has no limits on the amount of information to send.

If a user typed "Bill" and "Gates" in the form example above, the URL sent to the server would look like this:

[<http://www.w3schools.com/receive.asp>

Assume that the ASP file "receive.asp" contains the following script:

```
Thank you
```

```
<%
```

```
response.write(request.form("text1  "))
```

```
response.write(" " & request.form("text2"))
```

```
%>
```

```
Or
```

```
<%=text 1%>
```

```
<%=text2%>
```

The browser will display the following in the body of the document:

[Thank you Mohammad Alsheyab

2.7.4 Form Validation

User input should be validated on the browser whenever possible (by client scripts). Browser validation is faster and you reduce the server load. You should consider using server validation if the user input will be inserted into a database. A good way to validate a form on the server is to post the form to itself, instead of jumping to a

different page. The user will then get the error messages on the same page as the form. This makes it easier to discover the error.

2.8 ADO Database Connection

Before a database can be accessed from a web page, a database connection has to be established. The ADO Connection object is used to create an open connection to a data source. Through this connection, you can access and manipulate a database.

2.8.1 Create a DSN-less Database Connection

The easiest way to connect to a database is to use a DSN-less connection. A DSN-less connection can be used against any Microsoft Access database on your web site.

If you have a database called "message.mdb" located in a web directory like "D:\my project\Data base/", you can connect to the database with the following ASP code:

```
<%  
dbpath="D:\my project\Data base\message.mdb"  
strProv="Driver={Microsoft Access Driver (*.mdb)};  
DBQ="& dbpath  
set objConn = server.createobject("ADODB.Connection")  
objConn.Open strProv  
%>
```

Note, from the example above, which you have to specify the Microsoft Access database driver (Provider) and the physical path to the database on your computer.

2.8.2 Create an ODBC Database Connection

If you have an ODBC database called "message.mdb" you can connect to the database with the following ASP code:

```
<%  
    set objConn=server.createobject("ADODB.Connection")  
    objConn.Open "message"  
%>
```

With an ODBC connection, you can connect to any database, on any computer in your network, as long as an ODBC connection is available.

2.8.3 Create an ODBC SQL

We can make here connection between the ASP page and the Database by using SQL.

SQL lets you perform four basic operations:

- INSERT - Add data.
- DELETE - Remove data.
- SELECT - Retrieve data.
- UPDATE - Change data.

2.8.3.1 The Insert Operation

When you send the information form the HTML page to the ASP page for example from register.asp to the receive.asp this page will receive this information by using VBScript when you write this code:

```
<%@ LANGUAGE="VBScript"%>  
<%dim name, surname  
    dim objConn, dbpath, SQL, result, strProv, myfile  
    name=request.form("txtl ")  
    surname=request.form("txt2")%>
```

After receiving this information you can store it in the Database by using SQL when you are using this code for example name of my table in Database is Table2:

```
<%SQL="InsertIntoTable2(N ame,Sumame) Values(' '&name&' ','&surname&')"  
    set myfile= objConn.Execute(SQL)%>
```

This data it will save in my table in Database.

2.8.3.2 The Delete Operation

Suppose you want to delete some data from the table in the Database of course you can make this delete by using the SQL when you write this code after determine which row you want to delete for example find.asp page determine which row you want to delete the delete.asp page it will receive the locate of row and make delete for the data inside that row in the Table2:

```
<%SQL="Delete* from Table2 where SerialNumber ="&t  
set myfile= objConn.Execute(SQL)%>
```

Before write this code you must determine the path of Database and make connection to it, also write the code to receive the data from the HTML page or other pages.

2.8.3.3 The Search Operation

Suppose some body need to make search about something in the table in database, for example the find.asp make search for name when it receive the name from the price.asp so you have to write this code:

```
<%SQL="select* from Table2 where Name=""&name&""  
set myfile= objConn.Execute(SQL)%>
```

Of course after making connection to the database, so when the page make search it will show the information in table by using this code:

```
<table>  
<tr>  
<th>  
<tr direTtr'c-Surname-c/pc-c/th>
```

```

<th>
<p dire'Ttr'c-Namec/pc-c/th>
-c/tr>
<%
While not myfile.eof%>
<tr>
<td><%=myfile("Sumame")%></td>
<td><%=myfile( "Name ")%><td>
</tr>
<%
myfile.movenext
wend
myfile.close
set rnyfile=Nothing
%>
</table>

```

2.8.3.4 The Update Operation

When the users need to change some data in the table in database you can make this changed by using this code:

```

<%SQL="update Table2 set Name=""&name&"" , Sumame=""&sumame&""
set myfile= objConn.Execute(SQL)%>

```

Of course when he make connection to the database and received the new data from the HTML page for example modify.asp and determined which row you want to make update for it the mod.asp it will make this update in the Table2.

CHAPTER 3 JAVASCRIPT

3.1 Introduction

JavaScript is used in millions of Web pages to improve the design, validate forms, and much more.

JavaScript was developed by Netscape and is the most popular scripting language on the Internet.

3.1.1 What is JavaScript?

- JavaScript was designed to add interactivity to HTML pages.
- JavaScript is a scripting language - a scripting language is a lightweight programming language.
- A JavaScript is lines of executable computer code.
- A JavaScript is usually embedded directly in HTML pages.
- JavaScript is an interpreted language (means that scripts execute without preliminary compilation).
- Everyone can use JavaScript without purchasing a license.
- JavaScript is supported by all major browsers, like Netscape and Internet Explorer.

3.1.2 Are Java and JavaScript the Same?

NO!

Java and JavaScript are two completely different languages!

Java (developed by Sun Microsystems) is a powerful and very complex programming language - in the same category as C and C++.

3.1.3 What Can a JavaScript Do?

- JavaScript gives HTML designers a programming tool - HTML authors are normally not programmers, but JavaScript is a scripting language with a very simple syntax! Almost anyone can put small "snippets" of code into their HTML pages.

- JavaScript can put dynamic text into an HTML page - A JavaScript statement like this: `document.write("<h 1>" + name + "</h1>")` can write a variable text into an HTML page.
- JavaScript can react to events - A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element.
- JavaScript can read and write HTML elements - A JavaScript can read and change the content of an HTML element.
- JavaScript can be used to validate data - A JavaScript can be used to validate form data before it is submitted to a server, this will save the server from extra processing.

3.2 JavaScript How TO?

The HTML `<script>` tag is used to insert a JavaScript into an HTML page.

3.2.1 How to Put a JavaScript Into an HTML Page

```
<script language ="JavaScript">  
document.write("Hello World!")  
</script>
```

The code above will produce this output on an HTML page: Hello World!

To insert a script in an HTML page, we use the `<script>` tag. Use the `language` attribute to define the scripting language `<script language=" JavaScript">`

Then the JavaScript starts: The JavaScript command for writing some output to a page is `document.write`. `document.write("Hello World!")`

Then the `<script>` tag has to be closed. `</script>`

3.2.2 How to Handle Older Browsers

Browsers that do not support scripts will display the script as page content. To prevent them from doing this, we may use the HTML comment tag:

```
<script language="J avaScript">  
<!--  
    some statements  
!-->  
</script>
```

The two forward slashes at the end of comment line (//) are a JavaScript comment symbol. This prevents the JavaScript compiler from compiling the line.

Note: You cannot put // in front of the first comment line (like //<!), because older browsers will display it.

3.3 JavaScript Where To

- Scripts in the body section will be executed WHILE the page loads.
- Scripts in the head section will be executed when CALLED.

3.3.1 Where to Put the JavaScript

Scripts in a page will be executed immediately while the page loads into the browser. This is not always what we want. Sometimes we want to execute a script when a page loads, other times when a user triggers an event.

3.3.1.1 Scripts in the Head Section:

Scripts to be executed when they are called, or when an event is triggered, go in the head section. When you place a script in the head section, you will ensure that the script is loaded before anyone uses it.

```
<head>  
<script language="J avaScript">  
    Some statements  
</script>  
</head>
```

3.3.1.2 Scripts in the Body Section:

Scripts to be executed when the page loads go in the body section. When you place a script in the body section it generates the content of the page.

```
<body>
<script language="]avaScript">
    Some statements
</script>
<!body>
```

3.3.1.3 Scripts in Both the Body and the Head Section:

We can place an unlimited number of scripts in your document, so you can have scripts in both the body and the head section.

```
<head>
<script language="]avaScript">
    some statements
</script>
<!head>
<body>
<script language="]avaScript">
    some statements
</script>
</body>
```

3.4 JavaScript Variables

A variable is a "container" for information you want to store. A variable's value can change during the script. You can refer to a variable by name to see its value or to change its value.

Rules For Variable Names:

- Variable names are case sensitive.
- They must begin with a letter or the underscore character.

3.4.1 Declare a Variable

- We can create a variable with the var statement: `var stname = some value`
- We can also create a variable without the var statement: `stname = some value`

3.4.2 Assign a Value to a Variable

We can assign a value to a variable like this: `var c = 1;`

The variable name is on the left side of the expression and the value you want to assign to the variable is on the right. Now the variable "c" has the value "1".

3.5 JavaScript Functions

A function contains some code that will be executed by an event or a call to that function. A function is a set of statements. You can reuse functions within the same script, or in other documents. You define functions at the beginning of a file (in the head section), and call them later in the document.

3.5.1 How to Define a Function

To create a function you define its name, any values ("arguments"), and some statements:

```
function myfunction(argument], argument2, etc)
```

```
some statements
```

A function with no arguments must include the parentheses:

```
function myfunction()
```

```
some statements
```

Arguments are variables used in the function. The variable values are values passed on by the function call. By placing functions in the head section of the

document, you make sure that all the code in the function has been loaded before the function is called.

Some functions return a value to the calling expression

```
function formcontrol()

if(document.message. TxtName.value=="")

window.alert ("Please Enter your name and surname");
document.message. TxtName.focus()
return false;
}}
```

3.5.2 How to Call a Function

A function is not executed before it is called.

We can call a function containing arguments:

`myfunction(argument1,argument2,etc)`

Or without arguments: `myfunction()`

3.5.3 The Return Statement

Functions that will return a result must use the "return" statement. This statement specifies the value, which will be returned to where the function was called from.

3.6 JavaScript Conditional Statements

Conditional statements in JavaScript are used to perform different actions based on different conditions.

In JavaScript, we have two conditional statements:

- if statement - use this statement if you want to execute a set of code when a condition is true.
- if...else statement - use this statement if you want to select one of two sets of lines to execute.

3.6.1 If and If...Else Statement

3.6.1.1 Without ELSE

You should use if statement if you want to execute some code if a condition is true.

```
if (condition)
```

```
code to be executed if condition is true
```

```
}
```

Example:

There is no... else... in this example. Just tell the code to execute some code if the condition is true

```
function control(){  
  if(document.message.txt1.value=="someone")  
  
    window.alert ("Please Enter your Name");  
    document.message.txt1.focus()  
  return false;  
}}
```

3.6.1.2 With ELSE

If you want to execute some code if a condition is true and another code if a condition is false, use if... else statement.

```
if (condition)
```

```
code to be executed if condition is true
```

```
else
```

```
code to be executed if condition is false
```

```
}
```

Example:

```

if (-seed < msg.length) {
out+=msg.substring(-seed,msg.length);
seed--;
window.status=out;
cmd="scrollit("+seed+)";
timerTwo=window.setTimeout(cmd,100);
}
else {
window.status=" ";
timerTwo=window.setTimeout("scrollit(100)",75);
}

```

3.6.2 Conditional Operator

JavaScript also contains a conditional operator that assigns a value to a variable based on some condition.

```
variablename=(condition)?value1 :value2
```

Example:

```
X=(document.layers)?e.pageX:document.body.scrollLeft+event.clientX;
```

3.7 JavaScript Looping

Looping statements in JavaScript are used to execute the same block of code a specified number of times.

In JavaScript we have the following looping statements:

- For - run statements a specified number of times

3.7.1 For Loop

The for statement will execute a block of code a specified number of times



for (*initialization; condition; increment*)

code to be executed

Example:

```
for (var i=message.length-1; i>=0; i--) {  
    xpos[i]=xpos[i-1 ]+step;  
    ypos[i]=ypos[i-1 ]+step;  
}
```

CHAPTER 4. DATABASE. MICROSOFT ACCESS DATABASE.

4.1 Introduction

The database is an organized collection of data. A database management system (DBMS) such as Access, FileMaker Pro, Oracle or SQL Server provides you with the software tools you need to organize that data in a flexible manner. It includes facilities to add, modify or delete data from the database, ask questions (or queries) about the data stored in the database and produce reports summarizing selected contents.

4.1.1 What is a Database?

The database one or more, large structured sets of persistent data. Usually associated with software to update and query the data. A simple database might be a single file containing many records, each of which contains the same set of fields, where each field is a certain fixed width. A database is one component of a database management system.

4.1.2 Database Management System (DBMS)

The DBMS is in charge of access, security, storage and a host of other functions for the database system. It does this through a selection of computer programs. This allows it to manage the large, structured sets of persistent data, which make up the database, and provide access to the data for multiple, concurrent users whilst maintaining the integrity of the data.

The DBMS provides security facilities in a variety of forms, both to prevent unauthorized access and to prevent authorized users from accessing data at the same time as each other or overwriting information, which they should not. As a first line of security to prevent unauthorized users from accessing the system, it uses user names and passwords to identify operators, programs and individual machines and sets of privileges assigned to them. These privileges can include the ability to read, write and update data in the database.

The DBMS controls who is able to read and write data through the use locks. Locks are used for read and write to specific rows of data in relational systems, or

objects in object oriented ones, which are currently being used. By doing this only one user at a time can alter data.

DBMS's typically run on special hardware, for example servers which have been specially designed to only run databases and often only with specific database systems in mind. This allows the hardware designers to increase the number and speed of its network connections, use multiple processors and discs to speed up searched for information, increase the amount of memory and cache, as well as a host of other features not found on standard hardware. This also allows the programmers to take advantage of special features in the hardware to get the best possible performance from the system.

4.1.3 Database Design

Database Design is the process of taking the requirements for the database, i.e. what information is to be stored, who can access it, how many people may be accessing it simultaneously etc. and designing a system which will achieve this.

This initial stage is carried out by either a database design specialist, or the Database Administrators and Software Analysts. At the end of it, a schema is produced which defines what data is stored, how it relates to other data, and who can access, add and modify data. This schema is broken down into several sub-schemas which define individual tables and relationships between the tables and the data contained within.

4.2 Microsoft Access

Microsoft Access is a popular relational database management system for creating desktop and client/server database applications that run under the Windows operating system. This database is easy to use.

Access stores an entire database application within a single file. An Access *.mdb* file can contain data objects, like tables, indexes and queries, as well as application objects like forms, reports, macros, and visual basic code.

In this application, we used Microsoft Access as our database system. The reason to use Access was that we did not have many tables and queries and therefore it would be inefficient if we were to use Oracle or any other very-high Database system.

Microsoft Access gives us to create tables and do any manipulations on them and to create queries and retrieve data from any combination of pre-defined user tables.

4.3 Relations Between Tables

Some times, you need to make relationships between your tables to improve these tables.

Firstly, you must determine the primary key from the first table and which fields you want to make connection between it.

Secondly, make a connection. However, the data in this field must be same in whole tables.

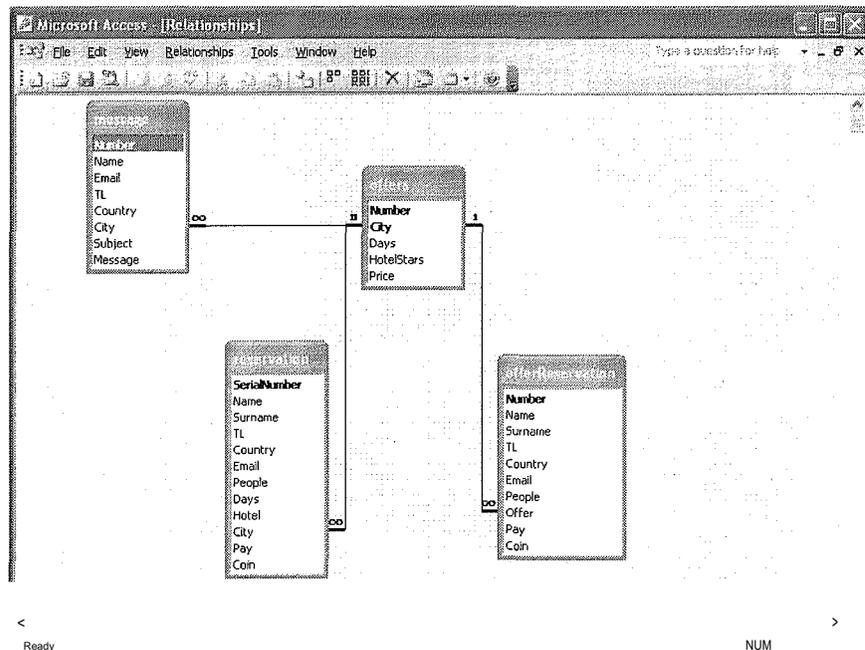


Figure 4.1 Relationships Between Tables

The diagram above shows the relationships between four tables as you see, the tables is message, reservation, offers and offerReservation.

4.4 Tables

4.4.1 Message

This is the first table in my database it is contain eight fields:

Field 1 needs lobe called 'Number and has the data type of 'AutoNumber Also set this field as the primary key.

Tourist Company Web Page Design

Field 2 needs to be called 'Name' and has the data type of text.

Field 3 needs to be called 'Email' and has the data type of text.

Field 4 needs to be called 'TL' and has the data type of number. However, here you have to change the Field size to Double.

Field 5 needs to be called 'Country' and has the data type of text.

Field 6 needs to be called 'City' and has the data type of text.

Field 7 needs to be called 'Subject' and has the data type of text.

Field 8 needs to be called 'Message' and has the data type of text. Here you must change the Field size to 255 or maximum size.

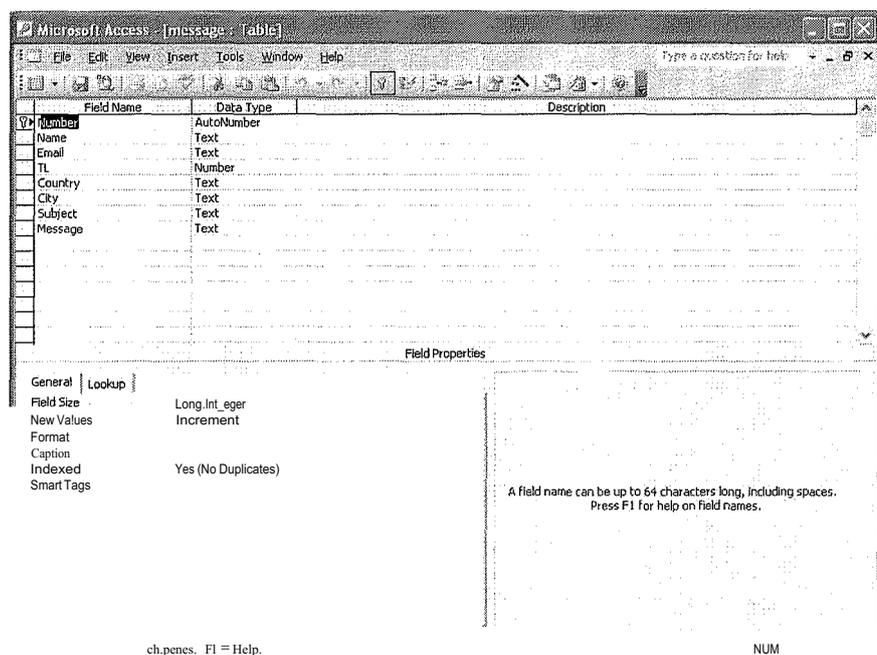


Figure 4.2 Designing The Message Table

This table makes a relation with offers table between cities as you see in Figure 4.1 and it receive the data from the web site when the user send message to the company and it will save these data in the fields

Number	Name	Email	TL	Country	City	Subject	Message
1	mohammad	alsheyab_mohammad@	0428848094	Cyprus	Agaba	project	some
2	samer	ad@yahoo.com	777247177	Australia	Jerash	see	help
3	hashem	hasash@yahoo.com	76204201	Other Country	Jerash	visit	i want to ask u
4	ammar alsheyab	ammar@yahoo.com	7-11331	Jordan	Petra	tourism	i want to get me
5	ahmad kassim	e.ahmad_kassim@yah	703701	Cyprus	Maaen	hi	hi man kefak
6	sultan alsmadi	sultan@hotmail	4743	Jordan	Jerash	tourism	see you

Figure 4.3 The Data Entry In Message Table

4.4.2 Reservation

This is the first table in my database it is contain 12 fields:

Field 1 needs lobe called 'SerialNumber and has the data type of 'AutoNumber also set this field as the primary key.

Field 2 needs to be called 'Name' and has the data type of text.

Field 3 needs to be called 'Surname' and has the data type of text.

Field 4 needs to be called 'TL' and has the data type of number. However, here you have to change the Field size to Double.

Field 5 needs to be called 'Country' and has the data type of text.

Field 6 needs to be called 'Email' and has the data type of text and it also primary key.

Field 7 needs to be called 'People' and has the data type of number.

Field 8 needs to be called 'Days' and has the data type of number.

Field 9 needs to be called 'Hotel' and has the data type of number.

Field 10 needs to be called 'City' and has the data type of text.

Field 11 needs to be called 'Pay' and has the data type of text.

Field 12 needs to be called 'Coin' and has the data type of text.

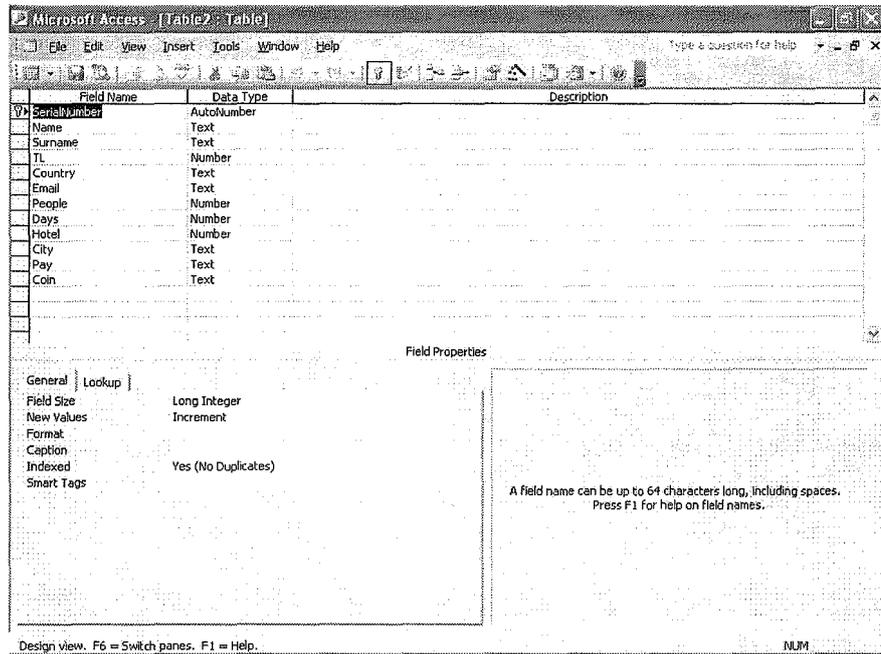


Figure 4.4 Designing The Reservation Table

This table makes a relation with offer table between cities as you see in Figure 4.1 and it receive the data from the web site when the user or customer reserve to visit some tourist places and keep it in the fields.

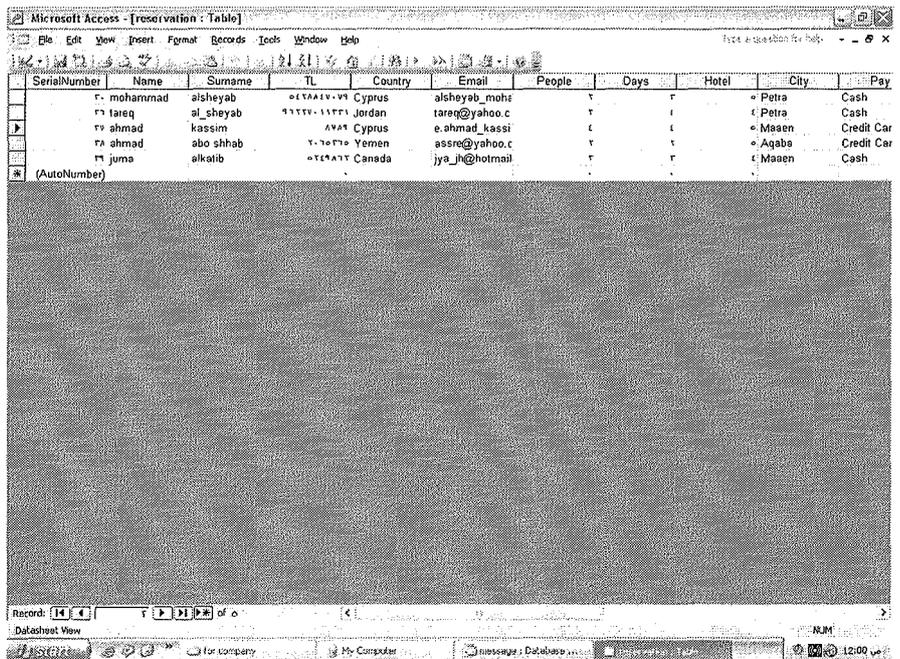


Figure 4.5 The Data Entry In Reservation Table

4.4.3 Offers

This is the first table in my database it is contain five fields:

Field 1 needs lobe called 'Number and has the data type of 'AutoNumber also set this field as the primary key.

Field 2 needs to be called 'City' and has the data type of text and it is a primary key also.

Field 3 needs to be called 'Days' and has the data type of number.

Field 4 needs to be called 'HotelStars' and has the data type of number.

Field 5 needs to be called 'Price' and has the data type of number.

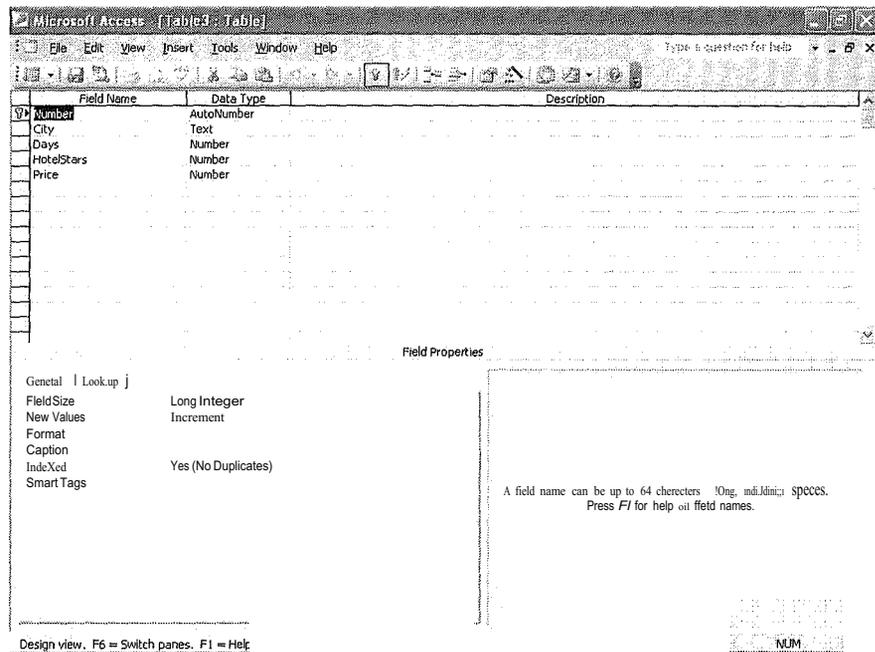


Figure 4.6 Designing The Offers Table

This table makes a relation with all tables:

1. With reservation table between cities.
2. With message table between cities.
3. With offerReservation table between city in the offers table and offer in the offerReservatton table.

As you see in Figure 4.1, it sends data to the web site when the user or customer need to see the offers, and also the manager can change the data in this table.

Number	City	Days	HotelStars	Price
1	Aqaba	4	4	200
2	Jerash	5	4	230
3	Maaen	3	5	220
4	Madaba	5	5	250
5	Petra	3	4	150
6	Salt	4	3	180
(AutoNumber)		0	0	0

Figure 4.7 The Data Entry In Offers Table

4.4.4 OffezReservation

This is the first table in my database it is contain 10 fields:

Field 1 needs lobe called 'SerialNumber' and has the data type of 'AutoNumber' also set this field as the primary key.

Field 2 needs to be called 'Name' and has the data type of text.

Field 3 needs to be called 'Surname' and has the data type of text.

Field 4 needs to be called 'TL' and has the data type of number. However, here you have to change the Field size to Double.

Field 5 needs to be called 'Country' and has the data type of text.

Field 6 needs to be called 'Email' and has the data type of text.

Field 7 needs to be called 'People' and has the data type of number.

Field 8 needs to be called 'Offer' and has the data type of text.

Field 9 needs to be called 'Pay' and has the data type of text.

Field 10 needs to be called 'Coin' and has the data type of text.

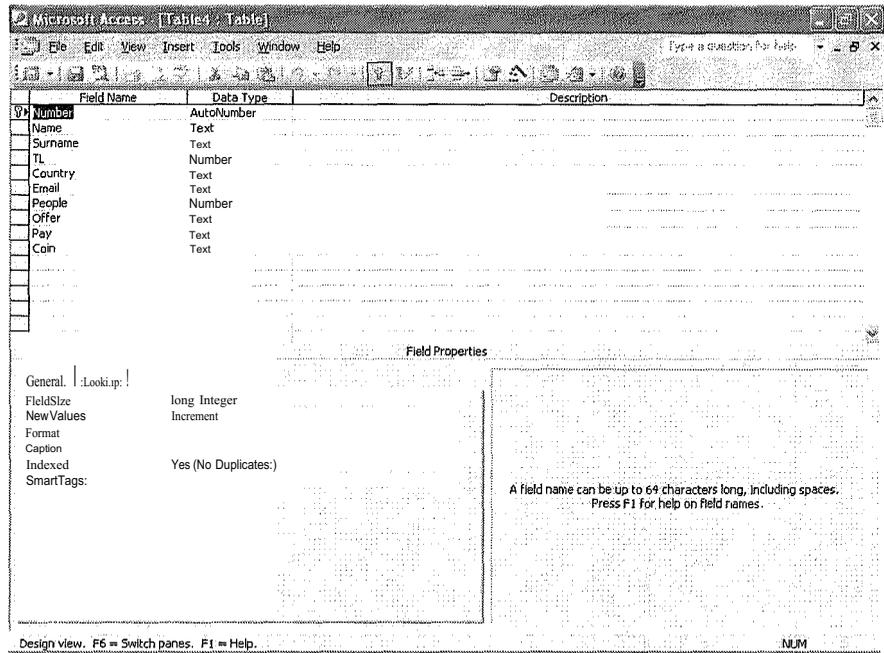


Figure 4.8 Designing The OfferReservation Table

This table makes a relation with offer table between city in the offers table and offer in the offerReservation table as you see in Figure 4.1, it receive data from the web site when the user or customer need to reserve in offers, and also it will keep the data in the fields.

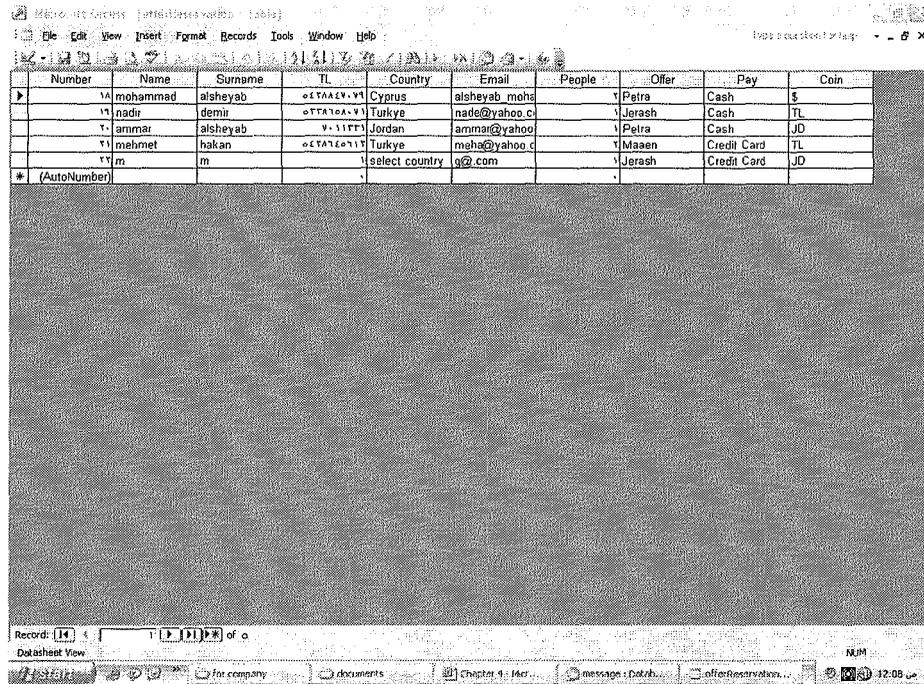


Figure 4.9 The Data Entry In OfferReservation Table

CHAPTER 5. TOURIST COPMANY WEB PAGE DESIGN

5.1 Introduction

Firstly my project contains around 31 pages some pages connecting with other pages and the other one connecting with database.

Secondly the relationships between the files or pages as the bottom figure 5.1:

First page connecting with 11 pages its search.asp, exhibit.asp, register.asp, Jordan.asp, user.asp and the 6 cities.

Every pages from this connecting with other as I said before the search.asp connecting with activate.asp, the exhibit.asp connect with offer.asp and offers.asp, the register.asp with received.asp, Jordan.asp with 7 pages and user.asp with select.asp.

The select.asp connecting with price.asp, the price.asp with num.asp and find.asp, the find.asp with delete.asp and modify.asp and the modify.sap connecting with mod.asp.

In the figure the user sue it to take some information about the history places or about Jordan, some times the user need to send message to the company for inquire about any thing, also the user can see the company offers and register in any offer if he like, there is also a reservation if the user desired to register with the company to visit any city he like, and in the last there is button for the company the user can not use this page because it is for manager and there is user name, ID number and password .

5.2 First Page

5.2.1 Project. Html

My project in HTML file represent in this figure:

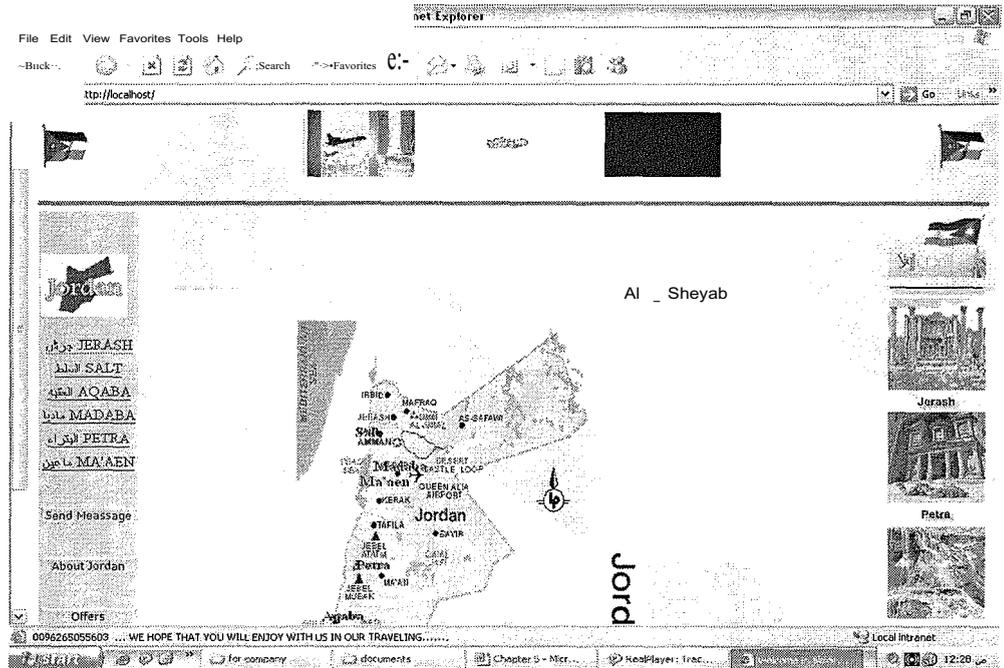


Figure 5.2 First Page In The Project. Project.HTML

So, our user arrives at the site and is presented with project.html.

In this page (project.html) I am using some techniques for example six <form> tags chapter 1, first form has three radio buttons and also the other forms have one button for each one, it will goes to the destination page by using Action="URL" to connect with the page .

```
<form method=post action="search/search.asp">
<input type=submit value="Send Meassage" style="width= 110px">
</form>
<form method= post action='about jordan/jordan.asp">
<input type="submit" value="About Jordan" style='width> 110px">
<!form>
<form method= post action="for company/exhibit.asp":>>
<input type="submit" value="Offers" style="width= 110px">
-oform>
```

Tourist Company Web Page Design

```
<form method= post action="resirvation/register.asp">
  <input type='tsubmit" value='Reservation" style='width= 110px">
</form>

<form method= post action="for company/user.asp">
<input type="submit" value="For Company" stylewidth= 110px">
-oform»
```

The Connecting Code In First Page

Then the users will see these buttons:

Choose what you

want to do

Send Message

..... ~ -

About Jordan

Offers

Reservation

For Corpany

Figure 5.3 The Connection Button In First Page

And also I am using `<a>` tag in this page to connect with the other pages.ximg> tag to connect with the picture, `<map>` tag to connect with the map and `<area>` tag this tag it doing like the `<a>` tag but the different it is that the `<area>` tag you can use it inside the map but the `<a>` you can not use it in the map.

```
<map name="FPMapO">
  <area href="tourism%20place/jerash.asp"shape="rect"coords="50,99,112,125">
  <area href="tourism%20place/salt.asp"shape="rect"coords="42,116,104,142">
  <area href="tourism%20place/ma'adeba.asp"shape="rect"coords="76,155,138,181 ">
  <area href="tourism%20place/ma'aen.asp"shape="rect"coords="60,173,122,199">
```

```
<area href="tourism%20place/petra.asp"shape="rect"coords="48,269,110,295">  
<area href="tourism%20place/aqaba.asp"shape="rect"coords="18,327,80,353">  
<!map.
```

Mapping And Connecting Code

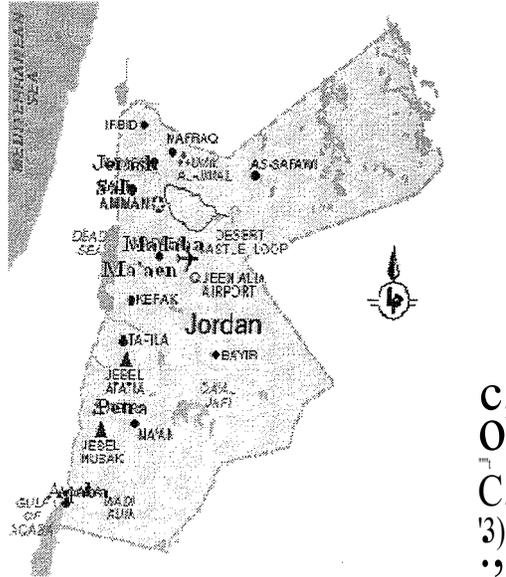


Figure 5.4 The Mapping Of Jordan

5.3 User Message

5.3.1 Search. Asp

This page open when you are click send message form the project.html page here you have to write your name, surname, Email , telephone number, country, subject and the message.

If you forget to write the name and telephone number the page it will give you a message to write your name and telephone number.

After filling every thing press send if you want to send this message or reset if you want to cancel this operation. Now when you send a message this page connection with activate. asp.

```
<table>  
<form method="post" action="activate.asp" name="message" onsubmit="return  
formcontrol(this)">
```

```

<tr><td><u><lb>Name and Surname.c/td>>
<tdc-cinput type="text" class="t'gonder" name="TxtName" size="50"
maxlength="50"></td>
</tr><tr><td><u><lb>E-Mail Address :</td>
<tdc-cinput type="text" class="gonder" name="TxtEmail" size="50" maxlength="50"
value="@"></td>
</tr><tr><td><u><lb> TL No.c/td>>
<tdc-cinput type="text" class="t'gonder" name="TxtNo" size="50" maxlength="50"
value="00 "></td></tr><tr><td><u>Country</b> :</td><td>
<select name="TxtCity" class="gonder" o>
<optiorc-vselect country-ve/option>
<optiorc-J ordan-c/option>
<option> Turkye-c/option>
<optionc-Cyprus-c/option>
<optioic-Uk-c/option>
<optiorc-U'S.Ac/option>
<zselect>
</td></tr><tr><td><u>S ubject</b> :</td>
<tdc-cinput type="text" class="gonder" name="TxtSubject" size="61"
maxlength="255"></td>
</tr><tr><td valign=vt'op'c-cic-Message-c/b> :</td>
<tdc-ctextarea name="Txtlvmessage" class="gonder" rows="10"
cols="60"></textarea><br></td>
</tr><tr><td colspan="2"><hr size="D" color="#000000"></td></tr>
<tr><td colspan="2"><center><input type="submit" value="Send" style='width:
200px;'> <input type="reset" value="Reset" style='width: 200px;'><br><br></td>
</tr>
<input type="hidden" name="act" value="send">
</oforrcc-c/table>>

```

The Message Code

5.3.2 Activate. Asp

The Activate. Asp receive the data from search. Asp by using request as we say in chapter 2, now this receive the data it will save this information in the Tablet in Database called message.mdb and give the user message when we received the data.

```

<%@ LANGUAGE="VBScript"%>
<%dim name,email,tl,country,subject,message
dim objConn, dbpath, ressel, result, strProv, myfile
name=request.form("TxtName")
email=request.form("TxtEmail ")
tl=request.form("TxtNo")
country=request.formf"TxtCity")
subject=request.form("TxtSubject")
message=request.formf"TxtMessage")
dbpath="D:\my project\for company\message.mdb"
strProv="Driver={Microsoft Access Driver (*.mdb)}; DBQ="& dbpath
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv
ressel="Insert          Into          Table1(Name,Email,TL,Country,Subject,Message)
          values("&name&","&email&","&tl&","&country&","&subject&","&message&")"
set myfile = objConn.Execute(ressel)
objConn.close
set objConn=Nothing%>
<h3>
<p align=center dir="Ttr">Thank you&nbsp;<%=name%> your detail are successful and we will
answer by the near time </p></h3>
<p align=centerc-cform method-epost action="search.asp">
<input type=submit value="OK"></form>

```

Active Message Code

5.4 Collecting Registration Details

In my project I have tow reservation one for the user doesn't need to register in the offers, so he registers in this page. The other one it is for the offers reservation when the user wants one of these offers.

5.4.1 Register. Asp

The purpose of this page, the user he can register himself and choose the city and days etc. as he like and then this page will send the data to the receive. asp.

```

<form method=post" action="receive.asp" name="message" onsubmit="return
control(this)">
<table align="center" border="0" bgcolor="#COCOCO">
<tr><td><h3>
<p align="center"><font color="#FF3300">Please Fill The Important
information</font></p></td>
<td><tr><td>
<p dir="ltr"> Your Name: <input type="text" name="txt1" value="someone"
size="40 "></td></tr>
<td>
<p dir="ltr"> Your Surname: <input type="text" name="txt2" value="someone"
size="40 "></td></tr>
<tr><td>
<p dir="ltr">TL No: <input type="text" name="txt3" value="OO" size="25"></td>
</tr><tr><td dir="ltr">
<p dir='ltr'>c-Country : <select name="txt4" class="gonder" o>
<option>c-select country-c/>
<option>c-Jordan-c/>
<option> Turkye-c/>
<option>c-Cyprus-c/>
<option>c-Uk-c/>
<option>c-Uô Ac/>
<option>c-Syria-c/>
<option>c-Saudi Arabia-c/>
<option>c-Other Country-c/>
</select>
</td></tr><tr><td>
<p dir="ltr">Email: <input type="text" name="txt5" value="@ " size="40"></td>

```

```

</tr><tr><td>
<p dir='Ttr'c-No of People: <input type="text" name="txt8" value="1" size="10"
maxlength=" 1 "></td></tr><tr><td>
<p dir="ltr">Days: input type="text" name="txt6" value="3" size="25"></td>
</tr><tr><td>
<p dir="ltr">Hotel Stars : <input type="text" name="txt7" value="5"
size=" 20 "></p><td></tr><tr>
<td dire Ttr'c-chrc-Choose The City You Want-c/td-c/tr>
<tr><td dir="ltr"><p diree'Ttr'c-City :
<select name="txt9" class="gonder" o>
-coptiorc-select city-c/option>
<optionc-Aqaba-c/option>
<option>J erash-c/option>
<optiorc-Maaen-c/option>
-coptiorc-Madaba-c/option>
<optiorc-Petra-c/option>
-coptiorc-Salt-c/option>
</select></p><td></tr><tr><td></tr>
<tr><td dir="ltr"><p dir='ltr"> The Currency :
<select name="txtl 1" class="gonder">
<optionc-select the currency-c/option>
<optiorc-Ifrc/option>
<optionc-Sc/option>
<option> Tl.x/option>
<optionc-CYl'<option>
<optiorc-Ec/option>
<optionc-Cc/option>
</select>
</p><td></tr>
<tr><td></td></tr><tr><td dir="ltr"><p direTtr'c-The Pay:
<select name="txtlO" class="gonder">
<optiore-select how you can pay-c/option>
<optiorc-Credit Card-c/option>

```

Tourist Company Web Page Design

```
-coptiorc-Cash-c/opüorc-  
-coptiorc-Chequec/option»  
<Zselectsc/p>  
<hr><tr><td align="center"><hr><p dir="ltr">  
<input type="submit" value="Send" style='width= 200Px"><input type="reset"  
style="width= 200Px "></td></tr></table></form>
```

The Reservation Code

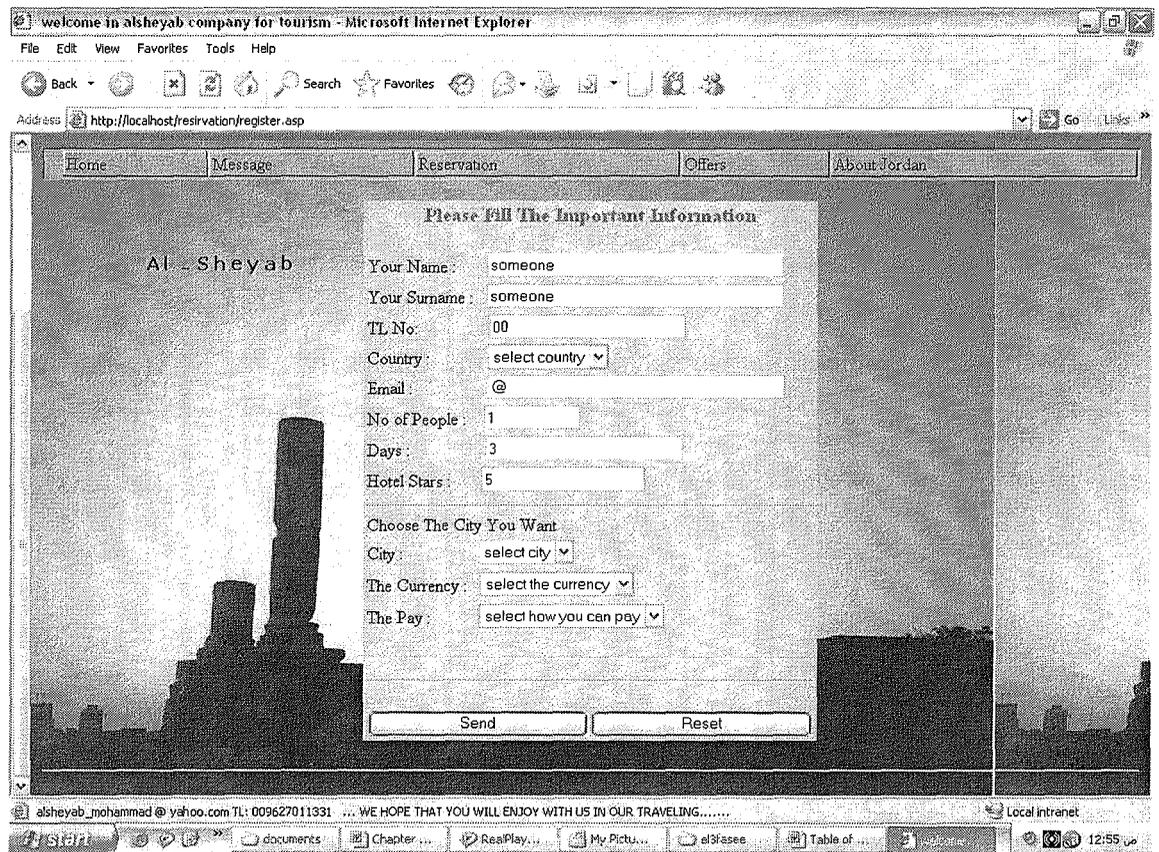


Figure 5.5 Normal Reservation

5.4.2 Function of JavaScript (VerifyData)

When the user forget to fill some information he will see a message tell him to fill the important information this message is written by the JavaScript you can see the function code here:

```
function control()

    if(document.message.txt1. value=="someone")

        window.alert ("Please Enter your Name");
            document.message.txt1 .focus()
        return false;

    if(document.message.txt2. value=="someone")

        window.alert ("Please Enter your Surname");
            document.message.txt2.focus()
        return false;

    if(document.message.txt3. value=="00")

        window.alert ("Please Enter your TL No");
            document.message.txt3 .focus()
        return false;

    if(document.message.txt6. value-c"l ")

        window.alert ("Please Enter Period of Visiting");
            document.message.txt6.focus()
        return false;
```

```
}
```

```
if(document.message.txt7. value >"5")
```

```
    window.alert ("Please Enter The Hotel Stars at Most 5");
```

```
        document.message.txt7 .focus()
```

```
return false;
```

```
if(document.message.txt8.value <"1 ")
```

```
    window.alert ("Please Enter How Mony Persons");
```

```
        document.message. txt8.focus()
```

```
return false;
```

The Testing Reservation Code Of JavaScript

5.4.3 Receive.Asp

This page receive the data from the Register.asp and store all the information in Table2 inside the Database called message.mdb then give a message to the user tell him the registers are successful.

```
<%@ LANGUAGE="VBScript"%>
<%dim name,suniname,tl,email,country,people,days,hotel,city,pay,coin
dim objConn, dbpath, SQL, result, strProv, myfile
        name=request.form("txt1 ")
        suniname=request.form("txt2 ")
        tl=request.formf "txt3 ")
        email =request.form( "txt5")
        country=request.form("txt4")
        people=request.form("txt8")
        days=request.form("txt6")
        hotel=request.form("txt7")
        city=request.form("txt9")
        pay=request.form("txtl O")
```

```

        coin=request.form("txtl 1 ")
dbpath="D:\my project\for company\message.mdb"
strProv="Driver={ Microsoft Access Driver (*.mdb)}; DBQ=" & dbpath
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv
SQL="Insert Into
Table2(Name,Sumame,TL,Country,Email,People,Days,Hotel,City,Pay,Coin)
Values("&name&"","&surname&"","&tl&"","&country&"","&email&"","&people&"",
        "&days&"","&hotel&"","&city&"","&pay&"","&coin&")"
set myfile= objConn.Execute(SQL)
objConn.close
set objConn=Nothing%>
<br><br><br><h3>
<p align=center dir="ltr"><font color="#FFFFFF"> Thank you&nbsp;<%=sumame%>
your register are successful and you are Welcome in Jordan-cbr> We hope to enjoy
with us </font></p><lh3>
<p align=centerc-cform methodepost action= "register.asp">
<input type=submit value="OK"></form>

```

Active Reservation Code

5.4.4 Exhibit. Asp

This is the second page for register especially for offers register here the user he can determine which offer he like and fill all important information and send it to offer.asp to do activate for data and store it in database, also he can see the offers from offers.asp when he press on the name of offer.

```








<lp><p align="center"><table><tr><td dir="ltr">

```

```

<a href="Offers.asp?id=1" style="color: #0000FF; text-decoration: none"><h3><font
color="#FFFFFF"><u>
Aqaba Offer</c/></c/></font></h3></a> </td>
<td dir="ltr">
<a href="Offers.asp?id=2" style="color: #0000FF; text-decoration: none"><h3><font
color="#FFFFFF"><u>
Jerash Offer</u></font></h3></a> </td>
<td dir="ltr">
<a href="Offers.asp?id=6" style="color: #0000FF; text-decoration: none"><h3><font
color="#FFFFFF"><u>
Salt Offer</c/></font></h3></a> </td>
<td dir="ltr">
<a href="Offers.asp?id=4" style="color: #0000FF; text-decoration: none"><h3><font
color="#FFFFFF"><u>
Madaba Offer</u></font></h3> </td>
<td dir="ltr">
<a href="Offers.asp?id=3" style="color: #0000FF; text-decoration: none"><h3><font
color="#FFFFFF"><u>
Ma'aen Offer</u></font></h3></a> </td>
<td dir="ltr">
<a href="Offers.asp?id=5" style="color: #0000FF; text-decoration: none"><h3><font
color="#FFFFFF"><u>
Petra Offer</u> </font></h3></a> </td></tr></table></p>
<form method="post" action="toffer.asp" name="message" onsubmit="return
control( this)">
<table border="1" cellpadding="5" cellspacing="5"><tr><td>
<p dir="ltr">
<br></td>
<td><h3 dir="ltr">
<p align="center" dir="ltr"><font color="#000000"><u>Please Fill The Important
Information</u></font></p>
<h3 dir="ltr"></h3>
<h1 dir="ltr"><font color="#FFFFFF">

```

```

<p dir="ltr"> Your Name <input type="text" name="txt1" value="someone"
size="40"></p>
<p dir="ltr"> Your Surname <input type="text" name="txt2" value="someone"
size="40"></p>
<p dir="ltr"> No input type="text" name="txt3" value="OO" size="25"></p>
<p dir="ltr">Country :
<select name="txt4" class="gonder" o>
    <option value="Jordan">Jordan</option>
    <option value="Turkey">Turkey</option>
    <option value="Cyprus">Cyprus</option>
    <option value="HK">HK</option>
    <option value="UAE">UAE</option>
    <option value="Syria">Syria</option>
    <option value="Saudi Arabia">Saudi Arabia</option>
    <option value="Other Country">Other Country</option>
</select>
<p dir="ltr">Email: <input type="text" name="txt5" value="@ " size="40"></p>
<p dir="ltr">No of People : <input type="text" name="txt6" value="1" size="10"
maxlength="10"></p>
<hr dir="ltr">
<p dir="ltr">Choose The offer You Want
</p><p dir="ltr">Offer City :
<select name="txt7" class="gonder" o>
    <option value="Aqaba">Aqaba</option>
    <option value="Jerash">Jerash</option>
    <option value="Ma'aen">Ma'aen</option>
    <option value="Madaba">Madaba</option>
    <option value="Petra">Petra</option>
    <option value="Salt">Salt</option>
</select>
<p dir="ltr">The Currency:

```

```

        <select name="txt9" class="gonder">
        -coptiorc-select the currency-c/option>
        <option value="TL" >TL</option>
        <option value="CYP" >CYP</option>
        -coptiorc-Ec/option>
        <option value="Cc" >Cc</option>
        </select></p>
<p dir="ltr"> The Pay :
        <select name="txt8" class="tgonder">
        -coptiorc-select how you can paye/option>
        -coptiorc-Credit Card-c/option>
        -coptionc-Cash-c/option>
        -coptiorc-Chequec/option>
        </select></p>
-ofonr>
<h4 dir="ltr"></h4>
<p dir="ltr">
<input type="submit" value="Send" style="width= 180Px"><input type="reset"
style="width= 180Px ">
</p></td></tr>
<p dir="ltr">
-cbre-cimg src=" ../turism_image/ma2.jpg" width=" 150" height=" 150"></td></tr>
<tr-ctd>
<p dir="ltr"></td>
-ctdc-cp dir="ltr"></td>
-ctd dir="ltr"></td></tr>
-c/tablec-oform>

```

The Offer Reservation And Selecting Code

Now you can see the output of this page:

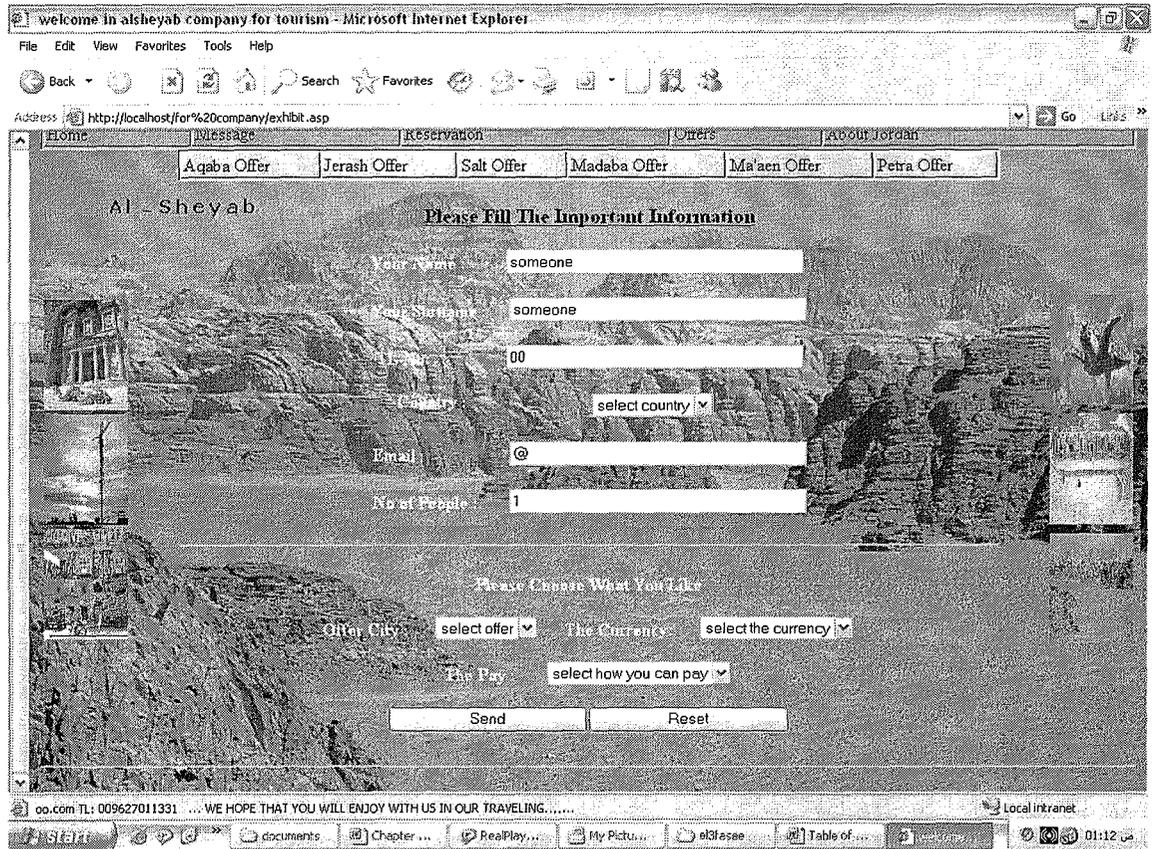


Figure 5.6 Offer Reservation

5.4.5 Offer. Asp

This is the active page for Exhibit.asp it is receive the data from that page and store it in Table4 inside the database called message.mdb as we say in chapter 4 and give a message to user tell him the registers are successful.

```
<%@ LANGUAGE="VBScript"%>
```

```
<%dim name,sumame,tl,email,country,people,city,pay,coin
```

```
dim objConn, dbpath, SQL, result, strProv, myfile
```

```
name=request.form("txt1 ")
```

```
sumame=request.form("txt2 ")
```

```
tl=request.form("txt3 ")
```

```
email=request.form("txt5")
```

```
country=request.form("txt4 ")
```

```
people=request.form("txt6 ")
```

```
city=request.form("txt7")
```

```

pay=request.form( "txt8 ")
coin=request.form("txt9")

dbpath="D:\rny project\for company\rmessage.mdb"
strProv="Driver={ Microsoft Access Driver (*.mdb)}; DBQ=" & dbpath
    set objConn = server.createobject("ADODB.Connection")
    objConn.Open strProv
    SQL="InsertInto Table4(Name,Surname,TL,Country,Email,People,Offer,Pay,Coin)
Values("&name&" , "&surname&" , "&tl&" , "&country&" , "&email&" ,
"&people&" , "&city&" , "&pay&" , "&coin&")"
set myfile= objConn.Execute(SQL)
objConn.close
set objConn=Nothing%>
<h3>
<p align=center dir="ltr"><font color="#FFFFFF">Thank you;<%=surname%> your
register are successful and you are Welcome in Jordan-cbr> We hope to enjoy with
us </font><lp><lh3>
<p align=centerc-cform method=post action="exhibit.asp">
<input type-submit value="OK"></form>

```

Offer Page Coding

5.4.6 Offers. Asp

When the user connect to see the offers he will go to this page here this page bring the data from Table3 from the database called message.mdb as we say in chapter 4 and show it in screen.

```

<%@ LANGUAGE="VBScript"%>
<%dim id
dim objConn, dbpath, SQL, result, strProv, myfile
id=request("id")
dbpath="D:\rny project\for company\rmessage.mdb"
strProv="Driver={Microsoft Access Driver (*.mdb)}; DBQ="& dbpath
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv

```

```

SQL="select* from Table3 where Number="&id
set myfile= objConn.Execute(SQL)%<h3></font><lp>
<center>
<table border="O" cellpadding="O" cellspacing="O" style="border-collapse: collapse"
bordercolor="#1 1111" width="100%" id="AutoNumber1 ">
    <%while not myfile.eof%>
        <tr><td width="71 %" bgcolor="#FFFFFF"><p>
<font color="#3399FF"><strong><em>City:</em></strong></font><strong><em>
<font color="#000033 "><%=myfile("City")%></font>
        <lem><lstrong> <lp>
<pc-cfont color="#FF0033"><strong><em>Days: <lem><lstrong><lfont>
<fontcolor="#330033" face="Courier New, Courier, mono" ><%=myfile("Days")%>
        <lfont><lp>
<p><font color="#FF0033 "c-cstrongi-cemc-Hotel Stars.c/emc-c/strongc-c/font>
<fontcolore "#333333 "><strong><em><%=myfile( "HotelS tars")%>&nbsp;
Stars-c/emc-c/ strong><lfont><lp>
<p><font color="#FF00FF"><em><strong>Price: <lstrong><lem><lfont><font
color="#9966FF"><em><strong><%=myfile("price")%>&nbsp;
$</strong><lem><lfont><lp>
        <p><p></p></td>
        <td><% if id=1 then%><% End if%></td>
        <td><% if id=2 then%><% End if%></td>
        <td><% if id=3 then%><% End if%></td>
        <td><% if id=4 then%><% End if%></td>
        <td><% if id=5 then%><% End if%></td>
        <td><% if id=6 then%><% End if%></td>
<td width="29% "><td><tr>

```



```
function formcontrol()
{
    if(document.message.TxtName.value!="MOH")

    window.alert ("Please Enter The Correct Name");
        document.message.TxtName.focus()
    return false;

    if(document.message.NO.value!="20034267")

    window.alert ("Please Enter The Correct ID Number");
        document.message.NO.focus()
    return false;

    if(document.message.password.value!="7011331")

    window.alert("Please Enter The Correct Password");
        document.message.password.focus()
    return false;
```

Active User Name And Password Coding Of JavaScript

5.5.2 Select. Asp

Here the manager selects which reservations want the normal reservation or offer reservation. When he chooses the reservation he will connect to price.asp page.

5.5.3 Price. Asp

In this page the manager can do search for the name from the tables in the database if he select both reservation, also he do search for whose reserve in any city as he choose and see the total of people reserves in this city but he can do this when he


```

<tr><td><%=myfile("People")%></td>
<td><%=myfile("Surname")%></td>
<td><%=myfile("Name")%></td>
<td><%=myfile("SerialNumber")%><1td>
</tr> <% myfile.movenext
wend
myfile.close
set myfile=Nothing
objConn.close
set objConn=Nothing%></table><br>
<%set conn= server.createobject("ADODB.Connection")
conn.provider= "Microsoft.JET.OLEDB.4.0"
conn.connectionstring = Server.mappath("message.mdb")
conn.open
SQL="SELECT sum(people) as asdf from Table2 WHERE City="&name&""
set rs= conn.execute(sql)
while not rs.eof%>
<table align="center" border="2" bgcolor="#CCCCCC" cellpadding="5"
cellspacing="5 ">
<tr><td></td><td><%=name%></td><td></td></tr>
<tr><td><%=rs("asdf")%></td><td></td><td>Total<ztde-c/tr>
<:%rs.movenext
wend%></table>

```

Searching Code By City

5.5.4.2 Find. Asp

This page appears the names in table when the manager does search about them for both tables of reservation.

Now if the manager selects normal reservation he can do delete for this name or update as he like.

```

<%@ LANGUAGE="VBScript"%>
<%dim name,c

```

```

dim objConn, dbpath, SQL, result, strProv, myfile
name=request.form( "search")
c=request.form("searchl ")
if(c="Resirvation")then
dbpath="D:\my project\for company\message.mdb"
strProv="Driver={ Microsoft Access Driver(* .mdb)}; DBQ=" & dbpath
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv
SQL="select* from Table2 where Name=""&name&""
set myfile= objConn.Execute(SQL)%><1h3></font><lp>
<table align="center" border="2" bgcolor="#CCCCCC" cellpadding="5"
cellspacing="5">
<tr><th> <p dir="ltr"> The Currency <lp><lth>
<th> <p dir='Ttr'c-The Pay <lp></th>
<th> <p dir="ltr">City <lp><lth>
<th><p dir=Ttr'c-Days-c/pc-c/th>
<th><p dir="ltr">Hotel Stars-c/pc-c/th>
<th><p dir='Ttr'c-No of People-c/pc-c/th>
<th><p dire'Ttr'c-Email-c/pc-c/th>
<th><p dir='ltr'c-Country-c/pc-c/th>
<th><p dir=Ttr"> TL No-c/pc-c/th>
<th><p dir=Ttr'c-Surnamec/pc-c/th>
<th><p dir='ltr'c-Name-c/pc-c/th>
<th><p dir=Ttr'c-Serial Number-c/pc-c/th>
</tr><% while not myfile.eof%>
<tr> <td><%=myfile("Coin")%></td>
<td><%=myfile( "Pay ")%><td>
<td><%=myfile("City")%></td>
<td><%=myfile( "Hotel ")%><td>
<td><%=myfile( "Days ")%><td>
<td><%=myfile( "People ")%><td>
<td><%=myfile( "Email ")%><td>
<td><%=myfile("Country")%><td>

```

```

<td><%=myfile("TL")%></td>
<td><%=myfile( "Surname ")%></td>
<td><%=myfile("Name")%></td>
<td><%=myfile("SerialNumber")%></td>
</tr> <% myfile.movenext
wend
myfile.close
set myfile=Nothing
objConn.close
set objConn=Nothing%>
-otablec-dirc-eform method="post" action="delete.asp" name="message"
onsubmit="return control(this)">
<h3 align="center" dir="ltr"><font color="#FFFFFF">Enter the Serial
Number-cinput type='text' name="txtNo">
<br-cbr-cinput type='submit' value="Delete"></font></h3></form>
<form method="post" action="modify.asp" name="mod">
<p align="center"><input type=submit" value="Modify"></p></form>
<%else%>
<%dbpath="D:\my project\for company\message.mdb"
strProv="Driver={Microsoft Access Driver (*.mdb)}; DBQ=" & dbpath
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv
SQL="select* from Table4 where Name=" & name& ""
set myfile= objConn.Execute(SQL)%><h3></font></p>
<table align="center" border="2" bgcolor="#CCCCCC" cellpadding="5"
cellspacing="5"><tr><th> <p dir="ltr">The Currency </p></th>
<th> <p dir="ltr">The Pay -c/pe-c/th>
<th-cp dir='Ttr'c-Offer City-c/pc-c/th>
<th><p dir="ltr">No of People-c/pc-c/th>
<th><p dir="ltr">Emaik/p></th>
<th><p dire'Ttr'c-Country-c/pc-c/th>
<th><p dir="ltr">TL No-c/pe-c/th>
<th><p dire-'Ttr'c-Surnarne-c/pc-c/th>

```

```

<:th><p dir='Ttr'c-Narnec/pc-c/th>
<th><p dir=Ttr'c-Serial Number-c/pc-c/th>
</tr><% while not myfile.eof%>
<tr> <td><%=myfile("Coin")%><td>
<td><%=myfile("Pay")%></td>
<td><%=myfile("Offer")%></td>
<td><%=myfile("People")%></td>
<td><%=myfile("Email")%><td>
<td><%=myfile("Country")%></td>
<td><%=myfile("TL")%></td>
<td><%=myfile("Surname")%><td>
<td><%=myfile("Name")%><td>
<td><%=myfile("Number")%></td>
<ztr><% myfile.movenext
wend
myfile.close
set myfile=Nothing
objConn.close
set objConn=Nothing%></table><hr><%End if%>

```

Searching Code By Name

5.5.5 Deleting Procedure

5.5.5.1 Delete. Asp

This page makes delete for the name from the table when the manager determined the name by the serial number and give him message tell him the delete is successful.

```

<%@ LANGUAGE="VBScript"%>
<%dim t
dim objConn, dbpath, SQL, result, strProv, myfile
t=request.form("txtNo")
dbpath="D:\my project\for company\message.mdb"
strProv="Driver={Microsoft Access Driver (*.mdb)}; DBQ=" & dbpath

```

```
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv
SQL="Delete* from Table2 where SerialNumber ="&t
set myfile= objConn.Execute(SQL)
set myfile=Nothing
objConn.close
set objConn=Nothing%><h3></font><lp>
<h3><p align=center dir="ltr"> Your delete are successful <lp>
<h3><p align=center><form method=post action="user.asp">
<input type=submit value="OK"></form>
```

The Deleting Code

5.5.6 Updating Procedure

5.5.6.1 Modify. Asp

This page allow to manager to insert the data to do update in table inside the database when he fill the information it will connect to the mod.asp.

5.5.6.2 Mod. Asp

This page receive the data from modify.asp and make update in the table, also give a message to the manager tell him the update is successful.

```
<%@ LANGUAGE="VBScript"%>
<%dim number ,name,surname, tl,country ,email, people,days,hotel ,city ,pay ,coin
dim objConn, dbpath, SQL, result, strProv, myfile
number=request.form("txtNo")
name=request.form("t1 ")
sumameerequest.formt "t2")
tl=request.form("t3 ")
countryerequest.formt "t4")
emailerequest.formt "t5")
people=request.form("t6")
days=request.form( "t7")
hotel=request.formt "t8")
city=request.forrm "t9")
```

```
pay=request.form( "t1O")
coin=request.fom("tl 1")
dbpath="D:\my project for company\message.mdb"
strProv="Driver={Microsoft Access Driver (*.mdb)}; DBQ="& dbpath
set objConn = server.createobject("ADODB.Connection")
objConn.Open strProv
SQL="update          Table2          set          Name="&name&",
Sumame="&sumame&",TL="&tl&",Country="&country&",Email="&email&",
People="&people&",Days="&days&",Hotel="&hotel&",City="&city&",
Pay="&pay&",Coin="&coin&"where SerialNumber="&number
set myfile= objConn.Execute(SQL)
set myfileeNothing
objConn.close
set objConn=Nothing%><h3></font></p>
<h3><p align=center dir=Ttr">Your update are successful </p>
<lh3><p align=centerc-cinput type=submit value="OK"><lp><lform><lp>
```

The Updating Code

5.6 About Jordan

Here I am talking about Jordan and the user can connect to the Jordan.asp page from the main page and then connect to the seventh pages talks about Jordan.

5.7 Cities

Here I am talking about the history cities and the user can connect to any city from the main page and he will see some information about these cities.

So he will connect to Aqaba.asp, Jerash.asp, Madaba.asp, Ma'aen.asp, Petra.asp and Salt.asp.

CONCLUSION

The Internet allows any computer in the world to exchange data with any other computer in the world. During the point at the end of the project that using clever programming techniques and simple design, we can make a very well organized web page so that the user can be interact easily. There are many techniques for the web designing, that technique used is Active Server Page (ASP) and JavaScript.

The advantage of ASP file can contain text, HTML tags and scripts. Scripts in an ASP file are executed on the server. In addition, we can do to execute the connection with the server and send only report to the browser.

The advantage of JavaScript, we can call simple functions in head section and body section to make our web site like clock, date and animation, etc.

From the Database, ADO can be used to access databases from our web pages. The ADO Connection Object is used to create an open connection to a data source. Through this connection, we can access and manipulate a database. We can save the memory, that is we use simple database, that using many information to making connection. We come across the use of database in web site.

REFERENCES

- [1] *Internet and World Wide web, How To program*. Third edition, Deitel, H. M
- [2] *Database Design and programming, with Access, SQL, and ASP, Second edition*, JOHN CARTER. published by JOHN WILEY & SON Ltd. in 1995.
- [3] *Guide to Microsoft Access 2000 program*
- [4] *Microsoft Encarta Encyclopedia 2002*
- [5] *Microsoft Active Server Page* by Manas Tungare
- [6] *JavaScript for Beginners*, Copyright Martin Baier
- [7] *Data Manipulation Language (DML)* for KSA <http://www.phpvillage.com>
- [8] <http://www.visualbuilder.com/asp/tutorial>
- [9] <http://www.w3.org/TR/REC-html40/intro/intro.html>
- [10] <http://www.utexas.edu/its/windows/database/access/start.html>
- [11] <http://wwwdb.web.cern.ch/wwwdb/aboutdbs>
- [12] <http://www.JordanPhoto.com/photo>