

WEB DESIGNING USING HTML

HTML:

- HTML stands for Hyper Text Markup Language.
- It is the basic language used to create web pages.
- An HTML document is a text file which can be created using editor like Notepad2, Wordpad .
- This text file has to be saved with the extension .html or .htm.

BASIC ELEMENTS OF HTML :

- All HTML documents must start with a document type declaration: `<!DOCTYPE html>`.
- The HTML document itself begins with `<html>` and ends with `</html>`.
- The visible part of the HTML document is between `<body>` and `</body>`

GENERAL STRUCTURE OF HTML:

```
<html>

<head>

<title> </title>

</head>

<body>

</body>

</html>
```

EXAMPLE 1:

```
<!DOCTYPE html>
<html>
<head>
<title>PageTitle</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

Example Explained:

- The `<!DOCTYPE html>` declaration defines that this document is an HTML document
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the HTML page
- The `<title>` element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)
- The `<body>` element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

HTML ELEMENTS:

- An HTML element is defined by a start tag, some content, and an end tag:

<tagname>Content goes here...</tagname>

- The HTML **element** is everything from the start tag to the end tag:

<h1>My First Heading</h1>

<p>My first paragraph.</p>

HTML ATTRIBUTES:

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

INSERTING TEXT IN A WEB PAGE :

- Open the text editor

<html>

<head>

<title> My First Page </title>

</head>

<body>

.....

..... Contents.....

</body>

</html>

- Save the file with the extension .html or .htm
- Open the browser and run the saved file

EXAMPLE 2:

```
<!DOCTYPE html>
<html>

  <head>
    <title>This is document title</title>
  </head>

  <body>
    <h1>This is a heading</h1>
    <p>Document content goes
here.....</p>
  </body>

</html>
```

OUTPUT:

This is a heading

Document content goes here.....

HTML TEXT FORMATTING:

- HTML contains several elements for defining text with a special meaning.

This text is bold

This text is italic

This is _{subscript} and ^{superscript}

HTML Formatting Elements

Formatting elements were designed to display special types of text:

- - Bold text
- - Important text
- <i> - Italic text
- - Emphasized text
- <mark> - Marked text
- <small> - Smaller text
- - Deleted text
- <ins> - Inserted text
- <sub> - Subscript text
- <sup> - Superscript text

- **CLOSE () FUNCTION:**

```
stream_object.close ()
```

- **PROGRAM:**

**The following program to
Create a data file
Display a data file
Adding a new record
Modify the existing record**

```
# include
class student
{
char name [30];
int rn;
public: void getdata ();
void putdata ();
};
void student :: getdata ()
{ cout <> name; cout << "Enter roll
number"; cin >> rn;
}

void student :: putdata ()
{
cout << "Student name" << name << "\n";
cout << "Student roll number" << rn <<
"\n";
}
void main ()
{
fstream file;
file . open ( "ABC", ios::in | ios::out |
ios::binary);
student st;
int i, n;
cout << "How many records to enter";
cin >> n;
for (i = 1; i <= n, i ++ )
{
st. getdata ();
file . write ((char*) & st, sizeof st);
} // Display a data file
file . seekg ( 0, ios::beg);
while (file . read ((char*) & st, sizeof st))
{
st. putdata ();
```

```
}
file . clear (); // To make the end of file
mark false
// To append record
st . getdata ();
file . write ((char*) & st, sizeof st);
// To modify a record
file.clear ();
cout << "Enter record number";
cin >> n;
file . seekp ((n - 1)* sizeof st, ios::beg);
st. getdata ();
file.write ((char*) & st, sizeof st);
// To close a file
file . close ();
}
```

CHECK YOURSELF

1. Which header file is required to use file I/O operations?
A) <ifstream>
B) <ostream>
C) <fstream>
D) <iostream>
2. Which of the following is not used as a file opening mode?
A) ios::trunc
B) ios::binary
C) ios::in
D) ios::ate
3. By default, all the files in C++ are opened in _____ mode.
A) Text
B) Binary
C) ISCI
D) VTC
4. What is the return type open() method?
A) int
B) char
C) bool
D) float is a pointer to a string

5. Which operator is used to insert the data into file?

- A) >>
- B) <<
- C) <
- D) None of the above

ANSWERS

Answers to Check Yourself:

- 1. C
- 2. B
- 3. A
- 4. C
- 5. B

STRETCH YOURSELF

- 1. Write a program to write and read from file?
- 2. Write a program using open() to read and write a file.
- 3. Write a program to enter student details in the file and display the output.