

# PicoMite W network course

## Chapter 7

### Some basic HTML codes

To bring the files in the Pico's memory in pace with these lessons let's give the Basic programs and the HTML files more significant names.

The last Basic program was called server02.bas Load that program in the editor. It should look like this:

```
Dim buff%(512)
WEB ntp 1
PRINT
PRINT "Your IP address is : " + MM.Info(ip address)
PRINT

WEB tcp interrupt serverstart

Do
  randomnum = Int(Rnd(100)*100)
Loop

Sub serverstart
Local p%, t%
For a%=1 To MM.Info(MAX connections)
  LongString CLEAR buff%()
  WEB tcp READ a%,buff%()edit

  p%=LInStr(buff%(),"GET")
  t%=LInStr(buff%(),"HTTP")
  s$=""
  If (p%<>0) And (t%<>0) And (t%>p%) Then
    s%=LGetStr$(buff%(),p%,t%-p%+4)
    Print "String is ",s$
  EndIf
  If Instr(s$,"HTTP") Then
    Print "sending page"
    WEB transmit PAGE a%,"third.html"
  EndIf
Next a%
End Sub
```

Please change one line. Change:

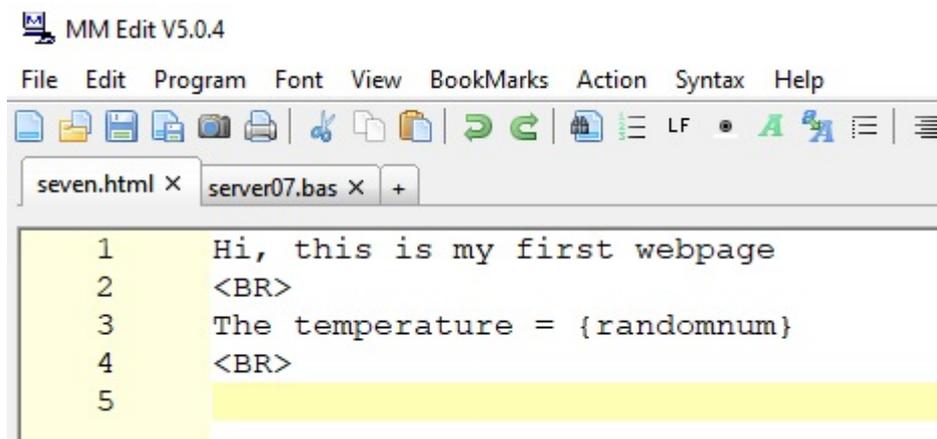
```
WEB transmit PAGE a%,"third.html"
```

Into

```
WEB transmit PAGE a%,"seven.html"
```

And save this program with the name **server07.bas** on your computer and then transfer it to the Pico.

In the previous chapters we made a text that was displayed in our webbrowser as a webpage. It works. However it is not really how the HTML code for a webpage should look. There are some conventions we need to follow.



```
Hi, this is my first webpage
<BR>
The temperature = {randomnum}
<BR>
```

This is how our HTML page looks now. Our browser fully accepts this without any problem. That is because HTML code in some ways is very forgiving. Let us make this however in the way it should be.

This is the file "third.html". Load this in your editor and save it as "seven.html"

The program names and the HTML files should now be in place in your directory on the Pico should look like this:

Maximite Control Centre

Setup Edit Action View Mode Help

picomite COM14

pico 'target port\COM14: s\picomite

Macro Macro Macro Macro Macro Macro Macro Macro Macro

X < > Send Send+CR

81 x 24 char, 895 x 436 pixels VT Terminal

```
files
A:/
  <DIR> .
  <DIR> ..
00:00 01-01-2000      4  bootcount
17:42 26-02-2023    128 first.html
21:13 02-03-2023     73 second.html
22:06 26-02-2023    506 server01.bas
16:56 05-03-2023    827 server02.bas
20:47 08-03-2023    650 server07.bas
20:48 08-03-2023     73 seven.html
19:11 05-03-2023    165 third.html
2 directories, 8 files
>
```

## HTML TAGS

Any html code should start with `<!DOCTYPE html>` and at the end of the page it ends with `</html>`. So for starters our page should look like this:

```
<!DOCTYPE html>

Hi, this is my first webpage
<BR>
The temperature = {randomnum}
<BR>

</html>
```

Please edit the file "seven.html" accordingly.

We already saw the `<BR>` code. This code only has an opening command. Normally HTML code has an opening and a closing command. We call these HTML Tags.

So the opening command is `<!DOCTYPE html>` and the closing command is `</html>` Opening and closing are distinguished by the backslash /

The next part an HTML file should have is a header. The header opens with `<header>` and closes with `</header>` In the header you can put all kinds of codes that are an introduction to the file. You can put a title in there, the name of the author etc. Many web developers are lazy and only put a pages title in there. A title opens with `<title>` and closes with `</title>`

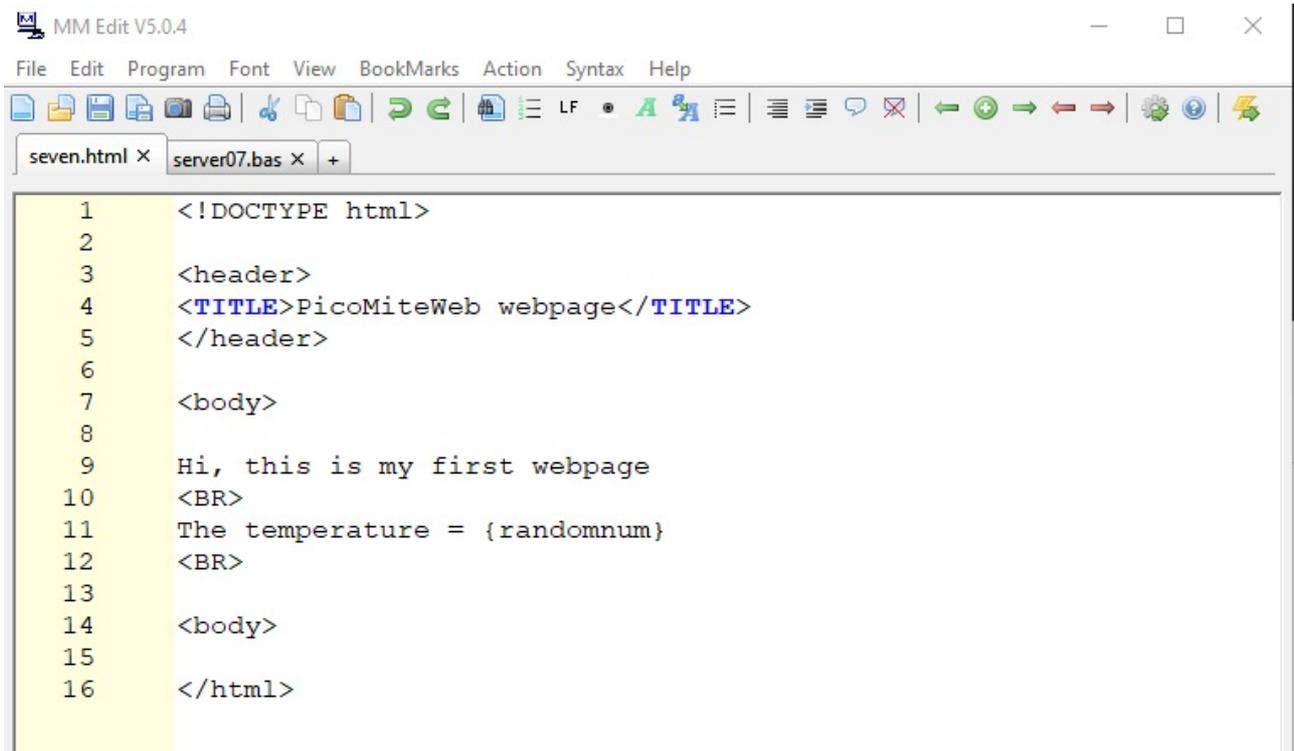
Does it do anything ?? Well a title on your webpage makes the page searchable for search machines like Google. Not that this site will be found at this moment as you are only visible from within your own network.

You can put opening tags and closing tags on new lines on your webpage or put them on the same line. HTML will not mind.

There is one last tag that officially must be included in your HTML code and that is the body tag.

After the header the webpage really starts and that is the body of your HTML code. Like almost all tags the body tag is opened with `<body>` and closed with `</body>`

The complete HTML code now looks like this:

A screenshot of the MM Edit V5.0.4 text editor. The window title is "MM Edit V5.0.4". The menu bar includes "File", "Edit", "Program", "Font", "View", "BookMarks", "Action", "Syntax", and "Help". The toolbar contains various icons for file operations and editing. The active window is "seven.html" and the code is as follows:

```
1 <!DOCTYPE html>
2
3 <header>
4 <TITLE>PicoMiteWeb webpage</TITLE>
5 </header>
6
7 <body>
8
9 Hi, this is my first webpage
10 <BR>
11 The temperature = {randomnum}
12 <BR>
13
14 <body>
15
16 </html>
```

Here is the code so you can copy it, paste it in MMEdit and save it as "seven.html".

```
<!DOCTYPE html>

<header>
<title>PicoMiteWeb webpage</title>
</header>

<body>

Hi, this is my first webpage
<BR>
The temperature = {randomnum}
<BR>

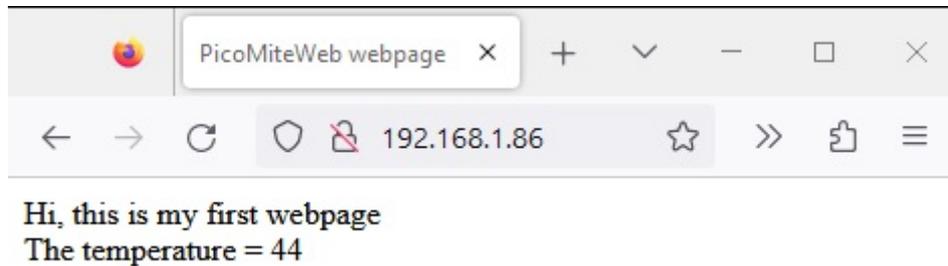
<body>

</html>
```

Like always save it on your computer and on the PicoMiteWeb. If a new version of PicoMiteWeb is introduced and installed on your Pico all files on the Pico may get

erased. So save them also on your computer.

Now run server07.bas then open your browser and put the IP number of the Picomiteweb in the URL bar and there is the webpage again.



Hey, look at the top of the webpage. It says PicoMiteWeb Webpage. That is why you put the title in the header. It neatly shows the name of the webpage in your browser.

So here are the HTML commands we have covered so far:

```
<!DOCTYPE html> </html>
<header> </header>
<title> </title>
<body> </body>
<br>
```

**<br>** is the only tag that has no closing tag. That speaks for itself as there is no closing for a new line.

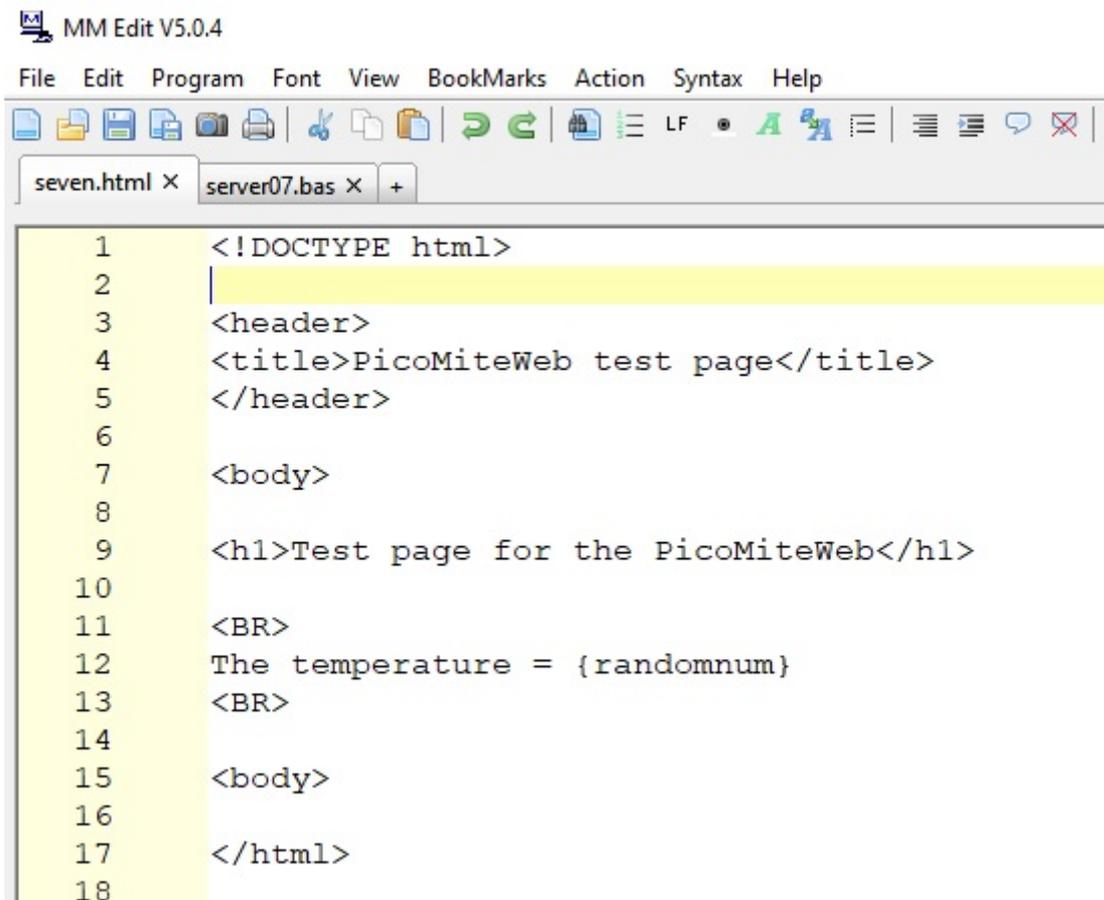
All html commands are made like this: **<commandname>** and everything else is just plain text. This is the way HTML knows that you give it a command. The closing tag **</commandname>** tells HTML that the command stops there.

Commands can be expanded like we are going to see next.

## Different font sizes

In HTML we can give a command to show a line of text in a larger font as normal. The command is `<h1>` and closes with `</h1>`. We can use this for emphasizing certain texts on our webpage.

Lets see what happens if we alter the HTML file as follows:



```
MM Edit V5.0.4
File Edit Program Font View BookMarks Action Syntax Help
seven.html x server07.bas x +
1 <!DOCTYPE html>
2
3 <header>
4 <title>PicoMiteWeb test page</title>
5 </header>
6
7 <body>
8
9 <h1>Test page for the PicoMiteWeb</h1>
10
11 <BR>
12 The temperature = {randomnum}
13 <BR>
14
15 <body>
16
17 </html>
18
```

Here is the code so you can copy it and save it again as "seven.html"

```
<!DOCTYPE html>

<header>
<title>PicoMiteWeb test page</title>
</header>

<body>

<h1>Test page for the PicoMiteWeb</h1>

<BR>
The temperature = {randomnum}
<BR>

<body>

</html>
```

And this is what it does with your webpage:



As you can see the first line is much bigger as the second line. That is what the **<h1>** text **</h1>** commands do.

**<h1>** is just one of the tags for font sizes (called heading not to confuse with header). You can use:

```
<h1> ..... </h1>
<h2> ..... </h2>
```

```
<h3> ..... </h3>
<h4> ..... </h4>
<h5> ..... </h5>
<h6> ..... </h6>
```

**<h1>** makes the largest text **<h6>** the smallest.

You can put the commands on a new line or on the same line where the text is. You also do not have to do this for each line.

```
<h1>
this is a line <br>
this is the next line<br>
this is the third line<br>
</h1>
```

Would show three lines in the biggest font size.

## Adding color to a page

Now we have a webpage with some edited text in different font sizes. But it has a rather dull color: white.

Well of course we can change every element on the page and we can certainly change the pages color. This is easily done with the style command.

What are we changing ?

Well we are changing the body of the webpage. So we need to change the <body> part.

Change the line <body> in your html file into:

```
<body style = "background-color: PowderBlue">
```

Your HTML file should now look like this:

```
<!DOCTYPE html>

<header>
<title>PicoMiteWeb test page</title>
</header>

<body style = "background-color: PowderBlue">

<h1>Test page for the PicoMiteWeb</h1>

<BR>
The temperature = {randomnum}
<BR>

<body>

</html>
```

If you edit the file manually please make sure that you spell the name of the color exact so: PowderBlue If you make an error the background color of the page will remain white.

Save it again as "**seven.html**"

And here is the result:



I like this color but I can imagine that you want something more lively (or dull). Well to each its own. There are 140 standard color names that HTML accepts

**[https://www.w3schools.com/colors/colors\\_names.asp](https://www.w3schools.com/colors/colors_names.asp)**

This page shows all the 140 standard colors with their names. So go wild and chose the background color you like.

There is another option you can try:

```
<body style="background-color:rgb(176,224,230)">
```

This gives the same result because 176,224,230 are the RGB values for the color PowderBlue.

But this gives us a lot more options. Just look take the following webpage:

**<https://htmlcolorcodes.com/>**

Here is a color picker. With the slider choose a color and then move the dot choose the exact color you need. You are not limited to the 140 standard colors anymore.



HEX #B0E0E6 RGB 176, 224, 230 HSL 187, 23%, 80%



The image shows a color picker interface. On the left is a large square with a vertical gradient from light blue at the top to black at the bottom, with a small white circle in the upper left. To its right is a vertical color slider with a white handle, showing a rainbow spectrum. On the far right is a white panel with a light blue header and a list of color values:

#	B0E0E6
R	176
G	224
B	<input type="text" value="230"/>
H	187
S	23
L	80
C	23
M	3
Y	0
K	10

And you can even use hex values if you like. Use them like this:

```
<body style="background-color: #b0e0e6">
```

**#b0e0e6** is the hex value which you can find in the color picker on the mentioned website. This is again the value for the color PowderBlue.

On the right side of the color slider you can see the hex values and the RGB values. You can change those numbers and immediately see the new color on the webpage.

## Adding color to an element.

Just like you can change the color of the complete webpage you can also change the color of an individual element. This can be done in the same way as the color of the body is changed.

Here is an example in which the color of the large text is changed.

```
<h1 style = "color: red">Test page for the  
PicoMiteWeb</h1>
```

Note that we change the color of the element, which is the text, in red. And indeed you can change the color using the 140 defined colornames or by using the RGB values or the hex values.



And of course we can also change the background color of just the text. To do so use the same style command as we did in the body but use a different color. In the next line we combine a different foreground and background color for the same text:

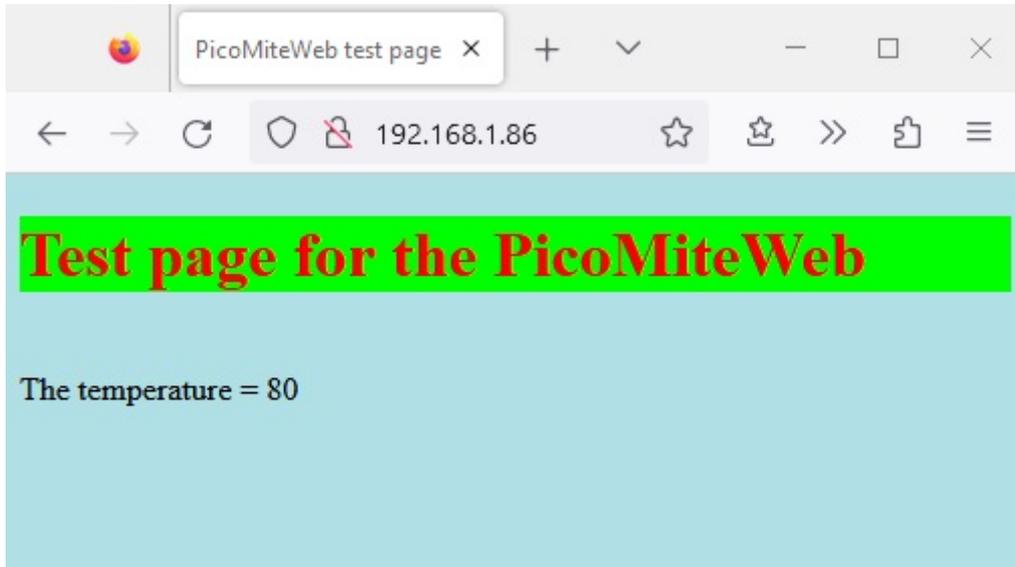
```
<h1 style = "color: red; background-color: green">Test  
page for the PicoMiteWeb</h1>
```

In the `<h1>` tag `style` is used once. After `style` the color parameters are supplied separated by a semicolon like this `color: red; background-color: green`

Like stated before you can replace the names by the RGB values like this:

```
<h1 style = "color: rgb(255,0,0); background-color:  
rgb(0,255,0)">Test page for the PicoMiteWeb</h1>
```

Or you can replace the color names by hex codes like shown above.



And here is the result.....

Play with this. Try color schemes that work for you. Build a webpage in sophisticated grey scales or a black background with green letters (like an old computer monitor) or make a complete carnival. It is all up to you.

For the future chapters I keep the PowderBlue background with the red letters, but no background for the letters.

The HTML code therefore looks like this:

```
<!DOCTYPE html>

<header>
<title>PicoMiteWeb test page</title>
</header>

<body style="background-color: PowderBlue">

<h1 style = "color: red">Test page for the
PicoMiteWeb</h1>

<BR>
The temperature = {randomnum}
<BR>

<body>

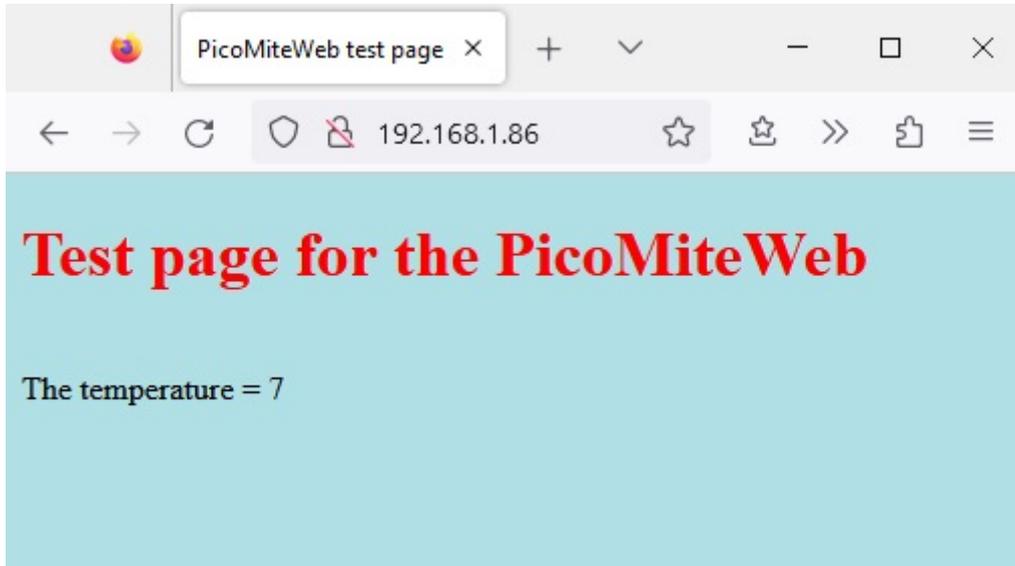
</html>
```

## The viewport

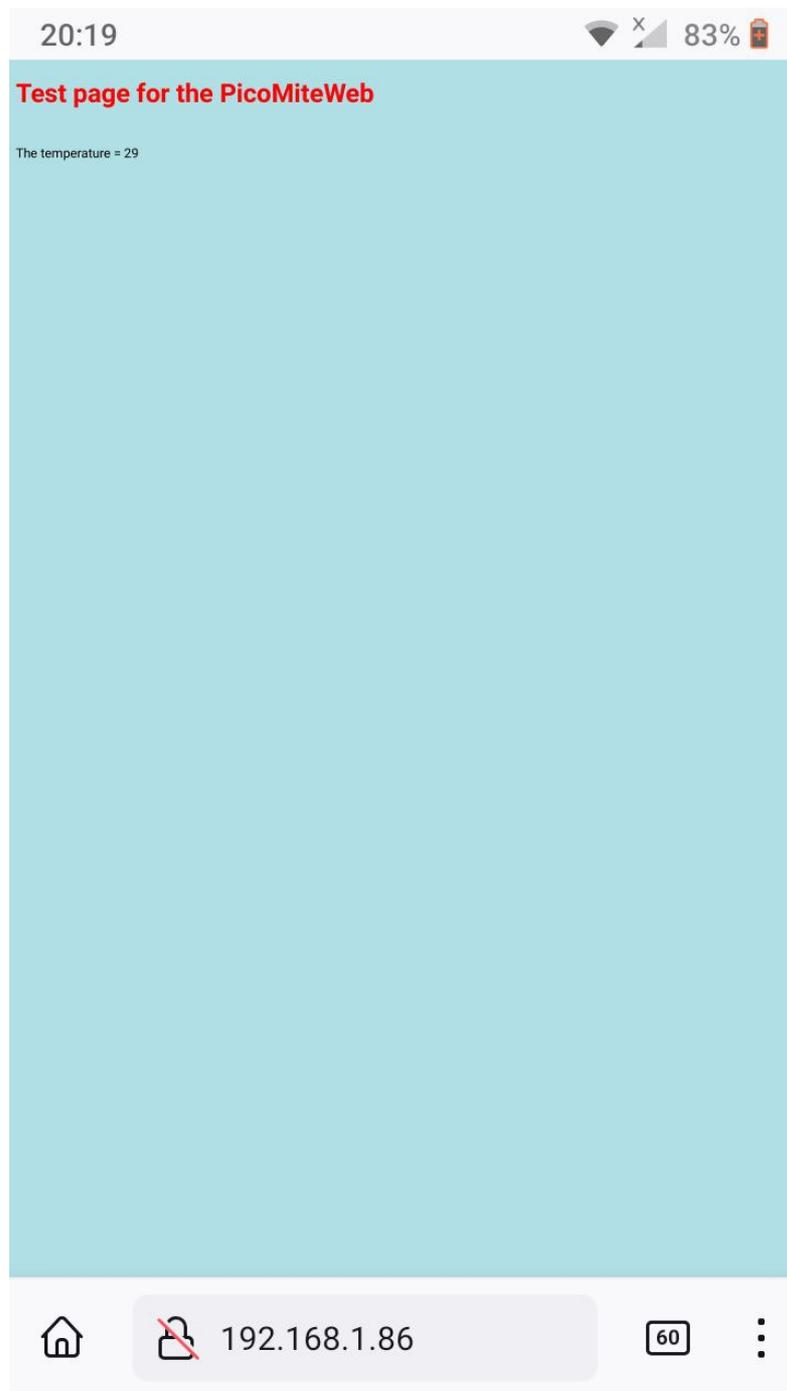
Hey what is that ???

Well actually this is a very important HTML code.

Let us look what happens if you open the webpage on your computer.



And how does it look if you open that webpage on your phone or tablet:



The difference is obvious. The letters on the phone are so small that you need to enlarge the webpage to be able to read them.

This is due to the fact that the resolution of your computer screen is totally different as that of your phone.

HTML works with pixels. And each device has a different pixel width and height. Simplified we can say that a picture of 50 pixels is displayed totally different on a screen with a width of 800 pixels or on a screen with a width of 1200 pixels.

Luckily HTML has a command to solve this problem and that is called the viewport.

This is how it is used:

```
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">
```

The command uses the devices width, which is different on your pc and on your phone, and scales the page to 1.

Let us put this in our HTML code.

```
<!DOCTYPE html>
```

```
<header>
```

```
<title>PicoMiteWeb test page</title>
```

```
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">
```

```
</header>
```

```
<body style="background-color: PowderBlue">
```

```
<h1 style = "color: red">Test page for the  
PicoMiteWeb</h1>
```

```
<BR>
```

```
The temperature = {randomnum}
```

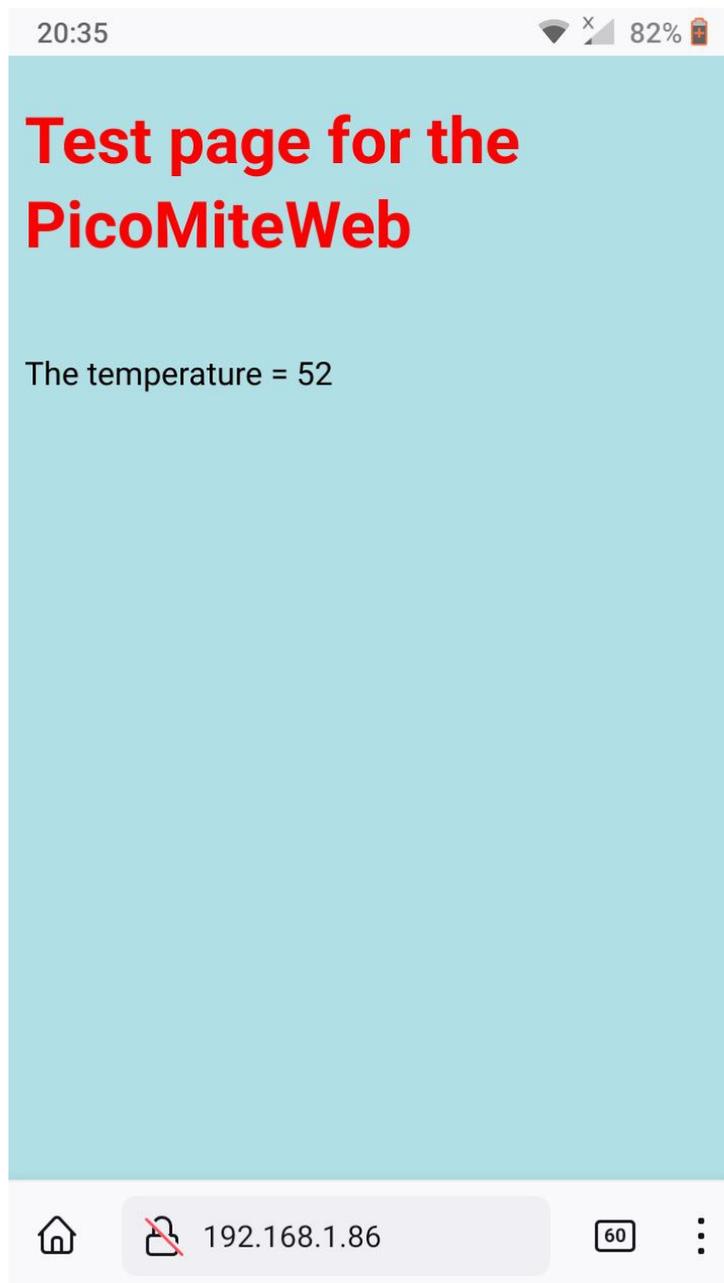
```
<BR>
```

```
<body>
```

```
</html>
```

Save this again as "seven.html" both on your PC and on the PicoMiteWeb. Run the good old server07.bas program again and now the page looks like it should on each of your devices.

Here is an example from my phone.



I really urge you to put the viewport command on every webpage that you are building.

The above HTML commands give you a lot to play with and now you can start building your webpages containing sensor readings in a nicer look.