

WEEK 5

Lab Exercises:

1. Write a simple C program (HelloWorld.c) that takes a number as an input and print the same number as an output. Now create a new git repository (check the contents of .git directory).

```
//HelloWorld.c
#include<stdio.h>
int main(){
int a;
printf("Enter a number: ");
scanf("%d", &a);
printf("The number entered is : %d", a);
return 0;}
```

```
student@dslab:~/190905513$ gedit HelloWorld.c
student@dslab:~/190905513$ git init
Initialized empty Git repository in /home/student/190905513/.git/
student@dslab:~/190905513$ git --version
git version 2.17.1
student@dslab:~/190905513$ git add HelloWorld.c
student@dslab:~/190905513$ git commit -m 'Initial project version'

*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'student@dslab.(none)')
student@dslab:~/190905513$ git config --global user.email "mohammad.eqbal@learner.manipal.edu"
student@dslab:~/190905513$ git config --global user.name "Mohammad Danish"
error: invalid key: user.nameMohammad Danish
student@dslab:~/190905513$ git config --global user.name "Mohammad Danish"
student@dslab:~/190905513$ git config -l
user.name=Mohammad Danish
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
student@dslab:~/190905513$ git config --global user.email "mohammad.eqbal@learner.manipal.edu"
student@dslab:~/190905513$ git config -l
user.name=Mohammad Danish
user.email=mohammad.eqbal@learner.manipal.edu
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
```

```
core.bare=false
core.logallrefupdates=true
student@dslab:~/190905513$ git config --global user.email "mohammad.eqbal@learner.manipal.edu"
student@dslab:~/190905513$ git config -l
user.name=Mohammad Danish
user.email=mohammad.eqbal@learner.manipal.edu
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
student@dslab:~/190905513$ gcc -c HelloWorld.c
student@dslab:~/190905513$ gcc -o HelloWorld HelloWorld.o
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 17

The number entered is : 17
student@dslab:~/190905513$ git init
Reinitialized existing Git repository in /home/student/190905513/.git/
student@dslab:~/190905513$ git status
On branch master

No commits yet

Changes to be committed:
(use "git rm --cached <file>..." to unstage)

    new file:   HelloWorld.c

Untracked files:
(use "git add <file>..." to include in what will be committed)

    .~lock.190905513_Week_5&6.docx#
    190905513_Week_5&6.docx
    HelloWorld
    HelloWorld.o
    OST_Lab_Manual_20-21.pdf

student@dslab:~/190905513$
```

2. Check the status of the repository. Add the file and commit the changes.

(Note: You can add all the changes file for commit using \$ git add -A)

```
student@dslab:~/190905513$ git add HelloWorld.o
student@dslab:~/190905513$ git status
On branch master

No commits yet

Changes to be committed:
(use "git rm --cached <file>..." to unstage)

    new file:   HelloWorld.c
    new file:   HelloWorld.o

Untracked files:
(use "git add <file>..." to include in what will be committed)

    .~lock.190905513_Week_5&6.docx#
    190905513_Week_5&6.docx
    HelloWorld
    OST_Lab_Manual_20-21.pdf

student@dslab:~/190905513$
```

3. Modify your C program (HelloWorld.c) such that it takes a number as input and prints a new number after adding 1 to the input. Check the changes that you have made using \$ git diff

```
//Modified HelloWorld.c
#include<stdio.h>
int main(){
int a;
printf("Enter a number: ");
scanf("%d", &a); a++;
printf("The number entered +1 is : %d", a);
return 0;
}
```

```
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 5
The number entered is : 5student@dslab:~/190905513$ git init
Reinitialized existing Git repository in /home/student/190905513/.git/
student@dslab:~/190905513$ ls
'190905513_Week_5&6.docx'  HelloWorld  HelloWorld.c  HelloWorld.o  OST_Lab_Manual_20-21.pdf
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 5
The number entered is : 5student@dslab:~/190905513$ gedit HelloWorld.c
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 6
The number entered is : 6student@dslab:~/190905513$ gcc -c HelloWorld.c
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 5
The number entered is : 5student@dslab:~/190905513$ gedit HelloWorld.c
student@dslab:~/190905513$ gcc -c HelloWorld.c
student@dslab:~/190905513$ gcc -o HelloWorld HelloWorld.o
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 5
The number entered +1 is : 6student@dslab:~/190905513$ git commit HelloWorld.c -m "New commit"
[master 835fb80] New commit
 1 file changed, 4 insertions(+), 3 deletions(-)
 mode change 100644 => 100755 HelloWorld.c
student@dslab:~/190905513$ git diff
diff --git a/HelloWorld.o b/HelloWorld.o
index c030ca0..5869eb4 100644
Binary files a/HelloWorld.o and b/HelloWorld.o differ
student@dslab:~/190905513$ 
```

4. Add this file to your git repository and commit. Check the logs of git commit using

```
$ git log
```

```
student@dslab:~/190905513$ git init
Reinitialized existing Git repository in /home/student/190905513/.git/
student@dslab:~/190905513$ git log
commit 835fb80df2ca38f1b20ab8561cef0cb58bb98bb6 (HEAD -> master)
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:44:35 2021 +0530

    New commit

commit ceffbc85c3ad40cd5b757f3e7ed0e94646e07a3f
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:31:13 2021 +0530

    New commit
student@dslab:~/190905513$ gedit HelloWorld1.c
student@dslab:~/190905513$ git add HelloWorld1.c
student@dslab:~/190905513$ git log
commit 835fb80df2ca38f1b20ab8561cef0cb58bb98bb6 (HEAD -> master)
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:44:35 2021 +0530

    New commit

commit ceffbc85c3ad40cd5b757f3e7ed0e94646e07a3f
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:31:13 2021 +0530

    New commit
student@dslab:~/190905513$ □
```

5. Unmodifying a Modified File:

- Again, modify your C program (HelloWorld.c) such that now it adds 2 to the input before printing it. Check the git status.

```
//Modified HelloWorld.c
#include<stdio.h>
int main(){
int a;
printf("Enter a number: ");
scanf("%d", &a);
a=a+2;
printf("The number entered +2 is : %d", a);
return 0;
}
```

```
student@dslab:~/190905513$ gedit HelloWorld.c
student@dslab:~/190905513$ gcc -c HelloWorld.c
student@dslab:~/190905513$ gcc -o HelloWorld HelloWorld.o
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 9
The number entered +2 is : 11student@dslab:~/190905513$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   HelloWorld
    modified:   HelloWorld.c
    new file:   HelloWorld.o
    new file:   HelloWorld1.c

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   HelloWorld
    modified:   HelloWorld.o

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .~lock.190905513_Week_5&6.docx#
    190905513_Week_5&6.docx
    OST_Lab_Manual_20-21.pdf

student@dslab:~/190905513$ 
```

- b. Later you can unmodify the program by \$ git checkout -- HelloWorld.c

(CAREFUL this deletes the local changes in the tracked file)

```
student@dslab:~/190905513$ gcc -c HelloWorld.c
student@dslab:~/190905513$ gcc -o HelloWorld HelloWorld.o
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 55
The number entered +2 is : 57student@dslab:~/190905513$ git checkout HelloWorld.c
student@dslab:~/190905513$ gcc -o HelloWorld HelloWorld.o
student@dslab:~/190905513$ gcc -c HelloWorld.c
student@dslab:~/190905513$ gcc -o HelloWorld HelloWorld.o
student@dslab:~/190905513$ ./HelloWorld
Enter a number: 41
The number entered is : 41student@dslab:~/190905513$ 
```

- c. Check git status again.

```
student@dslab:~/190905513$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   HelloWorld
    modified:   HelloWorld.c
    new file:   HelloWorld.o
    new file:   HelloWorld1.c

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    deleted:    HelloWorld1.c

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .~lock.190905513_Week_5&6.docx#
    190905513_Week_5&6.docx
    OST_Lab_Manual_20-21.pdf

student@dslab:~/190905513$ 
```

6. Checking old commit:

- Each commit operation generates a commit ID (you can see using git log)

```
commit 7973e54895bdfb7c226952bd612fec81055305e6
```

```
Author: cseuser10 <cseuser10@param-cse.(none)>
```

```
Date: Mon May 6 17:08:24 2019 +0530
```

```
student@dslab:~/190905513$ git log
commit 648bd0fd142be1e07e33e69010d64fb5ee9de875 (HEAD -> master)
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 14:23:09 2021 +0530

    New Commit

commit 835fb80df2ca38f1b20ab8561cef0cb58bb98bb6
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:44:35 2021 +0530

    New commit

commit ceffbc85c3ad40cd5b757f3e7ed0e94646e07a3f
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:31:13 2021 +0530

    New commit
student@dslab:~/190905513$ 
```

- b. Run \$ git show
7973e54895bdfb7c226952bd612fec81055305e6 (Your commit ID).

```
student@dslab:~/190905513$ git show 835fb80df2ca38f1b20ab8561cef0cb58bb98bb6
commit 835fb80df2ca38f1b20ab8561cef0cb58bb98bb6
Author: Mohammad Danish <mohammad.eqbal@learner.manipal.edu>
Date:   Mon Feb 15 13:44:35 2021 +0530

    New commit

diff --git a/HelloWorld.c b/HelloWorld.c
old mode 100644
new mode 100755
index 8bdbbc07..7f04e47
--- a/HelloWorld.c
+++ b/HelloWorld.c
@@ -1,9 +1,10 @@
 #include<stdio.h>
-int main(){
+int main()
+{
    int a;
    printf("Enter a number: ");
    scanf("%d", &a);
-printf("\nThe number entered is : %d\n", a);
+a++;
+printf("The number entered +1 is : %d", a);
    return 0;
}

student@dslab:~/190905513$ 
```

WEEK 6

Lab Exercises:

9. Specifying headings of the content and inserting paragraphs of text content :

- Headings – h1 to h6
- Paragraphs and line breaks
- Text formatting
- Preformatted text
- Marking deleted and inserted text
- Setting text direction of an element
- Usage of div and span tags

Code:

```
<html>

<head>OST Lab 190905513</head>

<title>HTML</title>

<style>

.myDiv

{

text-align: center;

}

</style>

<body>

<h1>Heading 1</h1>

<h2>Heading 2</h2>

<h3>Heading 3</h3>

<h4>Heading 4</h4>

<h5>Heading 5</h5>

<h6>Heading 6</h6>

<p>This is a paragraph with <br>line breaks</br></p>

<p><b>This is bold</b></p>
```

```
<p><i>This is italic</i></p>

<pre>This is preformmated text with <br></br>line breaks</pre>

<p>This is <del>deleted</del> and <ins> inserted </ins>text</p>

<p style="direction: rtl;">This text will be from right to left</p>

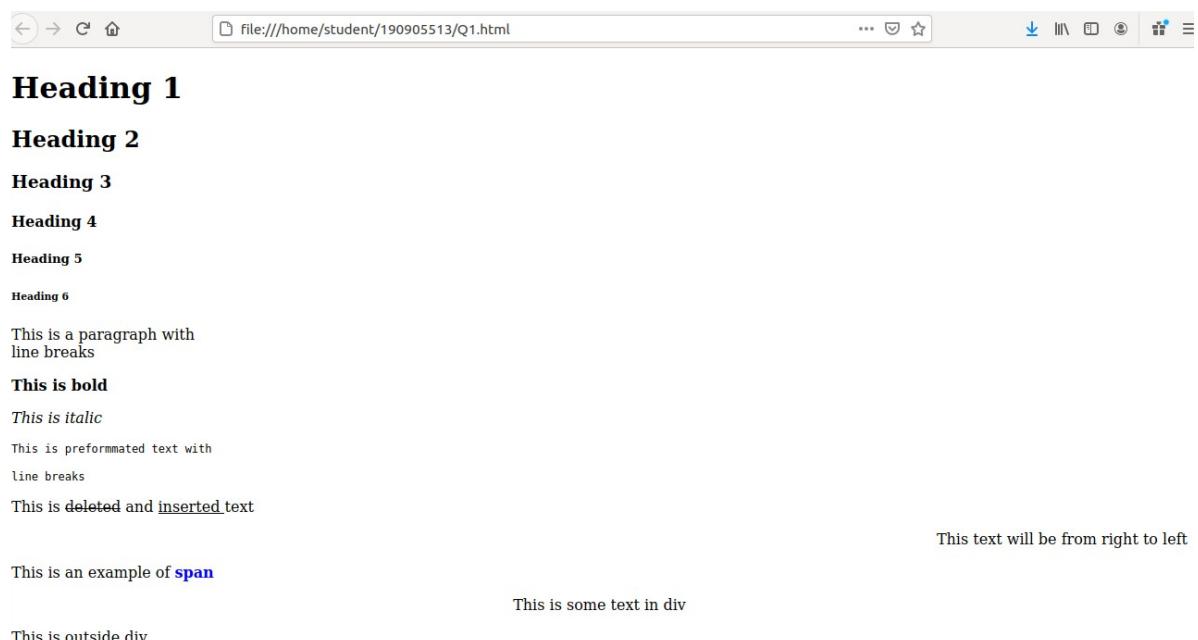
<p>This is an example of <span style="color: blue;font-weight:bold">span</span></p>

<div class="myDiv"><p>This is some text in div</p></div>

<p>This is outside div</p>

</body>

</html>
```

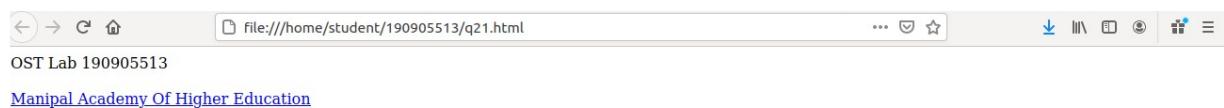


10. Creating links to other HTML documents or Web resources :

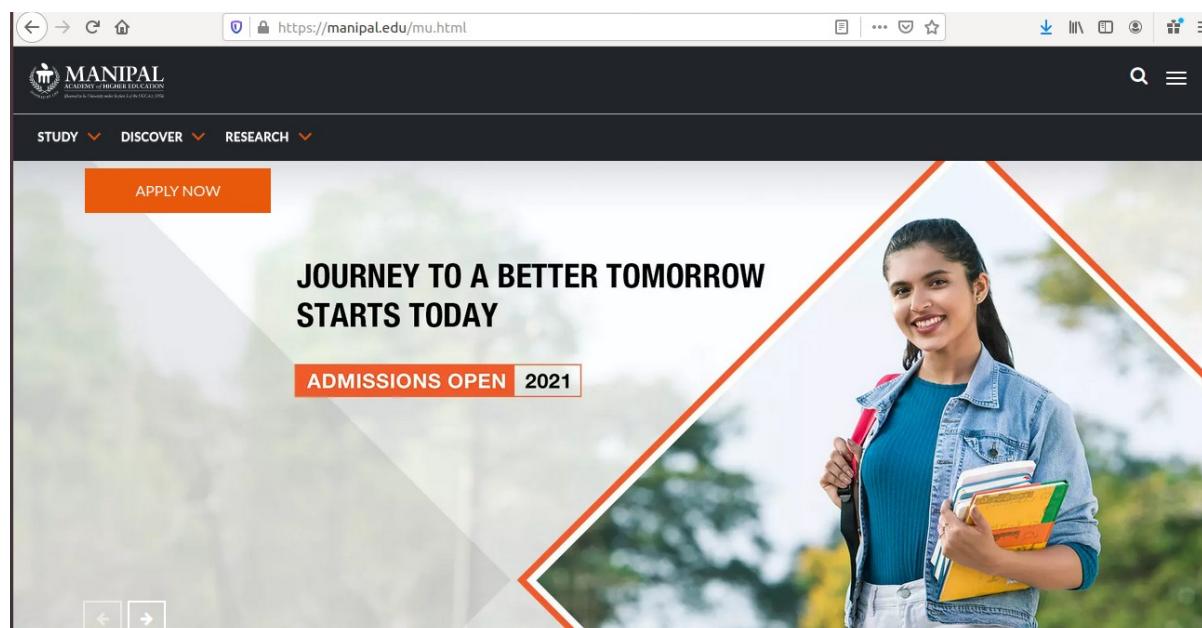
- Create hyperlinks
- Using image as a Link
- Open link in a new browser window

Creating Hyperlinks

```
<html>  
<head>OST Lab </head><br></br>  
<title>190905513</title>  
<body>  
<a href ="https://manipal.edu/mu.html">Manipal Academy of Higher Education</a>  
</body>  
</html>
```

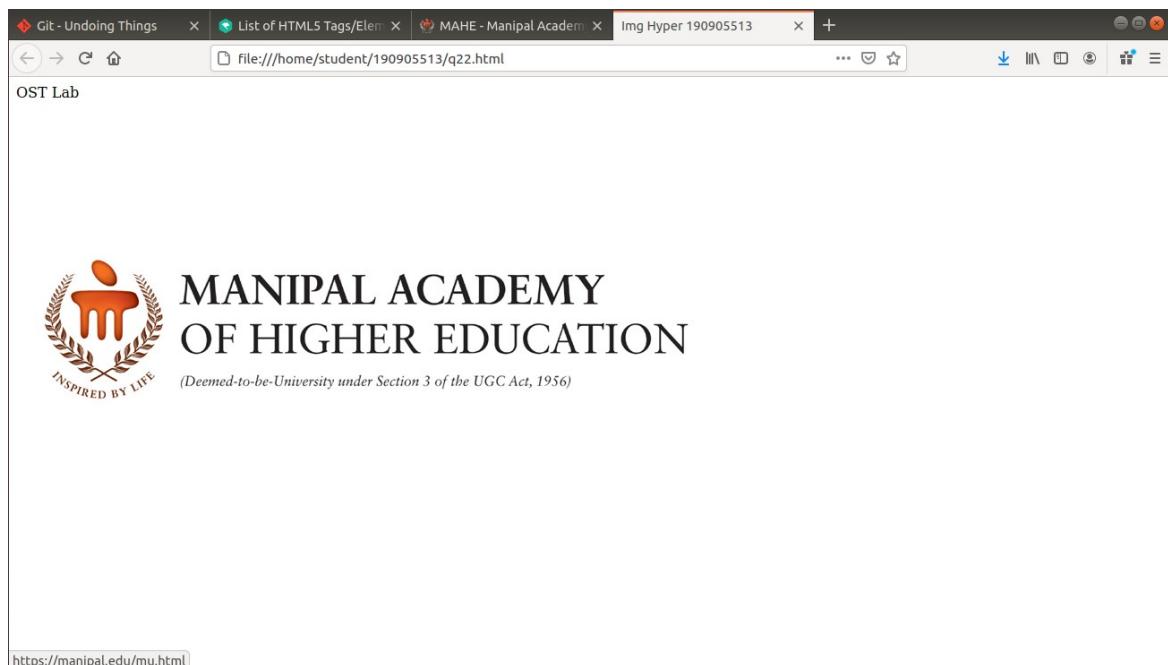


//After clicking on hyperlink text

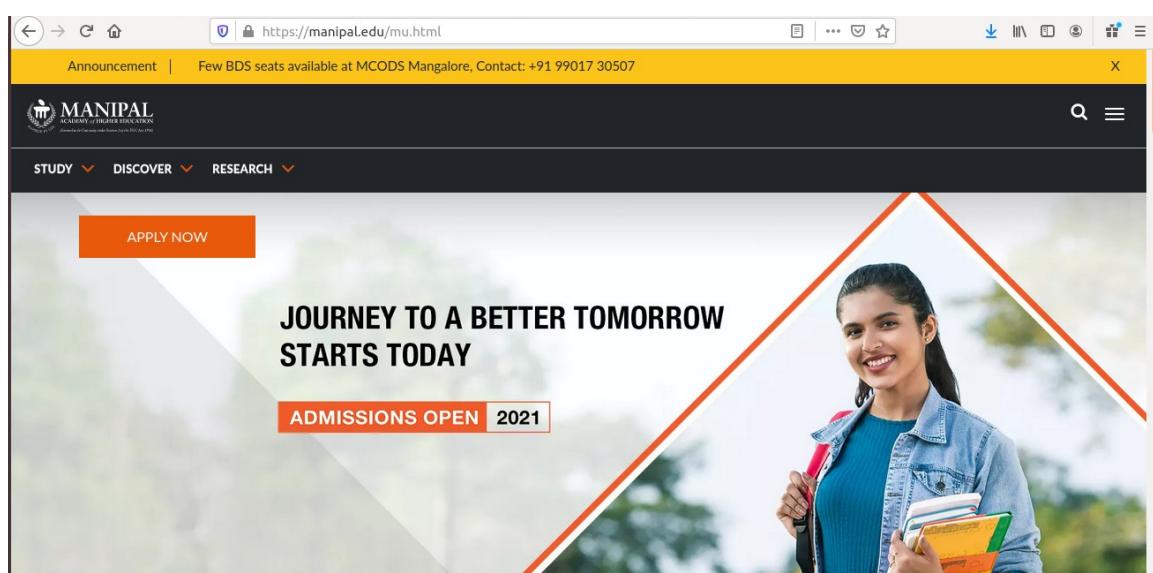


Using Image As a Link

```
<html>
<head>OST Lab</head>
<br></br>
<title>Img Hyper 190905513</title>
<body>
<a href ="https://manipal.edu/mu.html"> </a>
</body>
</html>
```



//After clicking on Hyperlink image(mahe logo)

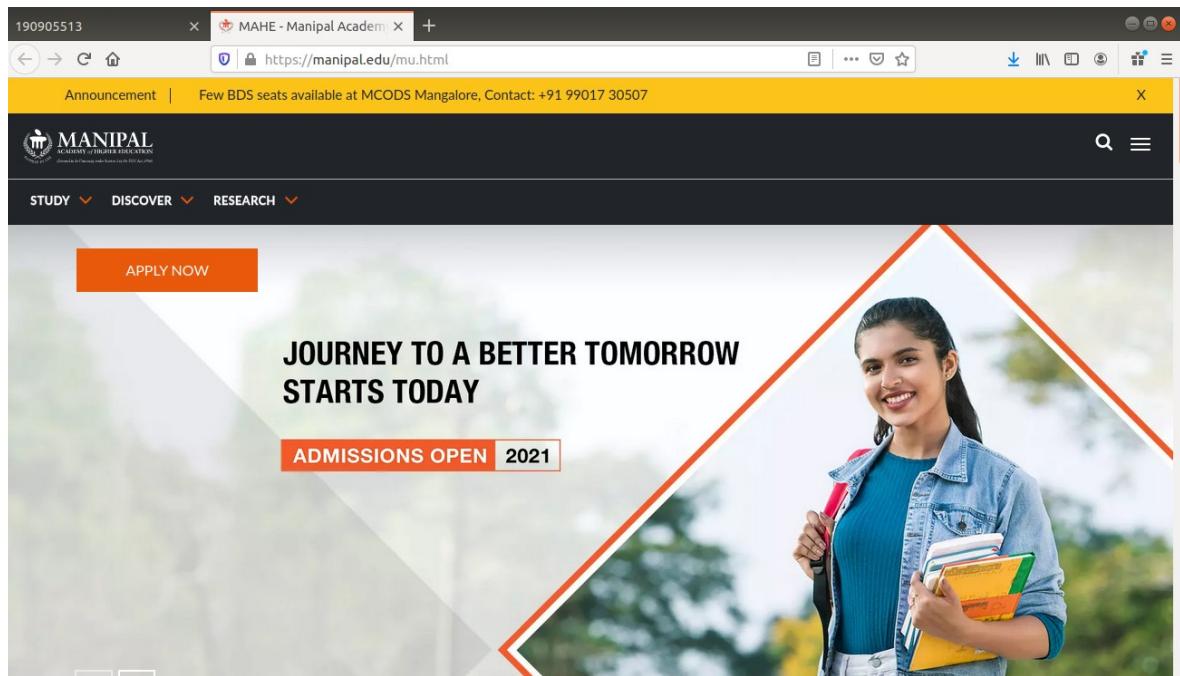


Opening Link In New Browser Window:

```
<html>
<head>OST Lab 190905513</head><br></br>
<title>190905513</title>
<body>
<a href ="https://manipal.edu/mu.html" target=_"blank">Manipal Academy Of Higher
Education</a>
</body>
</html>
```



//After clicking on Hyperlink text



11. Inserting images into the HTML document :

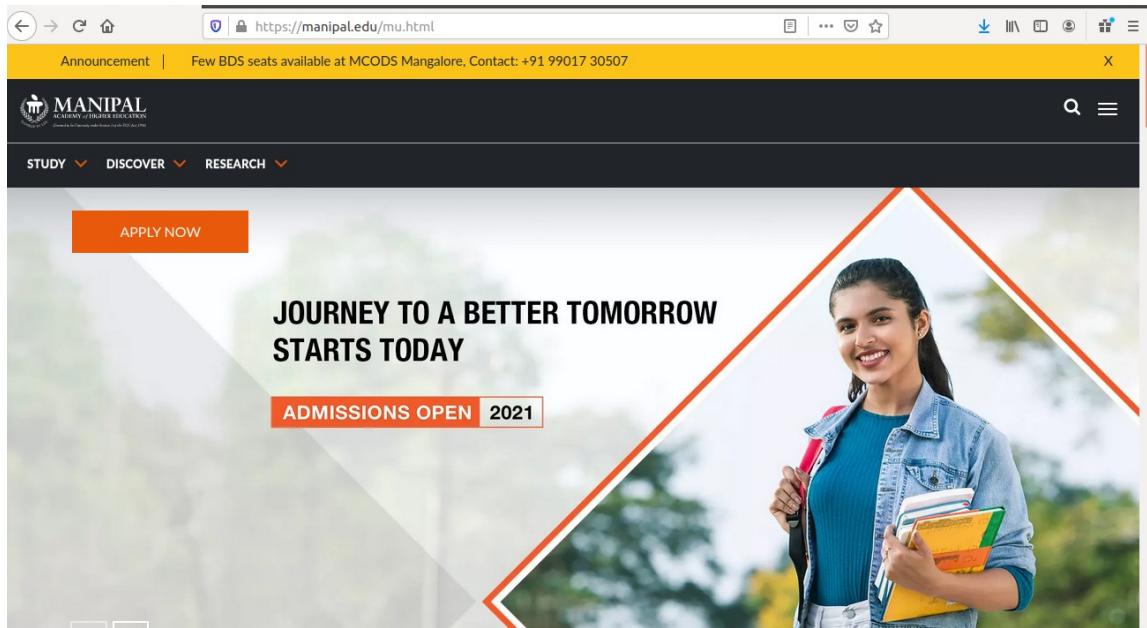
- Setting width and height of the images
- Aligning images
- Make a hyperlink of an image
- Creating an image – map (An image with clickable regions)

```
<html>
<head>OST Lab 190905513</head>
<br></br>
<title>Ost 190905513</title>
<body>
<div style="width:100%;text-align:center;">
<a href ="https://manipal.edu/mu.html">Manipal Academy of Higher Education</a>
</div>
</body>
</html>
```



[Manipal Academy of Higher Education](https://manipal.edu/mu.html)

//After clicking on Hyperlink



12. Creating a simple table :

- Setting the dimension of a table
- Specify the table caption
- Tables with borders
- Using cell padding, cell spacing, row span, column span attributes

```
<html>
<head>OST Lab2</head>
<style>table,th,td
{
border: 1px solid black;
}
th,td
{
padding=15px;
}
</style>
<br><br>
<title>HTML Table</title>
<body>
```

```

<table style="width:100%">
<caption> Details </caption>
<tr>
<th>Father's Name</th>
<td>Dr Parwez</td>
</tr>
<tr>
<th rowspan = "3">Children</th>
<td>Mohammad Danish</td>
</tr>
<tr>
<td>Eram Fatima</td>
</tr>
<tr>
<td>Mohammad Bilal</td>
</tr>
</body>
</html>

```

OST Lab2

Details

Father's Name	Dr Parwez
Children	Mohammad Danish
	Eram Fatima
	Mohammad Bilal

13. Making your own lists :

- Unordered, Ordered and definition list

```
<html>
```

```
  <head>OST Lab 190905513</head>
```

```
  <title>Dan Lists</title>
```

```
  <body>
```

```
    <ol>
```

```
      <li>Chocolate</li>
```

```
      <li>Juice</li>
```

```
      <li>Cake</li>
```

```
    </ol>
```

```
    <ul>
```

```
      <li>Chocolate</li>
```

```
      <li>Juice</li>
```

```
      <li>Cake</li>
```

```
    </ul>
```

```
    <dl>
```

```
      <li>Desert</li>
```

```
        <dd>Cake</dd>
```

```
      <li>Soft Drink</li>
```

```
        <dd>Coke</dd>
```

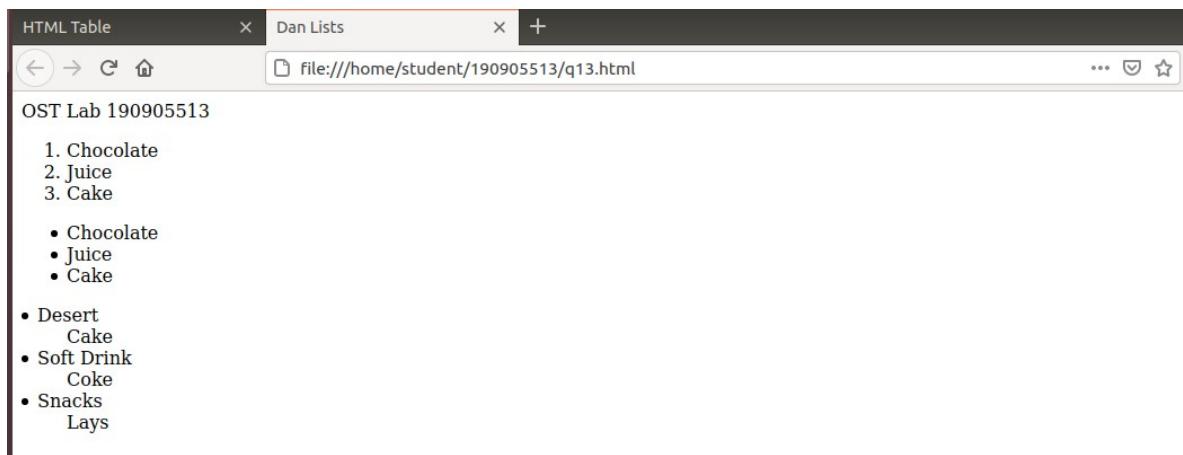
```
      <li>Snacks</li>
```

```
        <dd>Lays</dd>
```

```
    </dl>
```

```
  </body>
```

```
</html>
```



14. Creating a simple form :

- Using Text input field
- Creating password input field
- Checkboxes and radio buttons
- Drop down list
- Select box with preselected value
- Text area
- Creating buttons (Show the usage of submit and reset buttons)
- Grouping similar form fields

```
<html>
<head>OST Lab Dan</head>
<title>Form</title>
<body>
<form>
<fieldset>
<label for="fname">First Name:</label><br>
<input type="text" id ="fname" name="fname"><br>
<label for="lname">Last Name:</label><br>
<input type="text" id ="lname" name="lname"><br>
<label for="pwd">Password:</label><br>
<input type="password" id ="pwd" name="pwd"><br>
<input type="radio" id ="male" name="gender" value="male">
```

```

<label for="male">Male</label><br>
<input type="radio" id ="female" name="gender" value="female">
<label for="female">Female</label><br>
<input type="radio" id ="others" name="gender" value="others">
<label for="others">Others</label><br>
<input type="checkbox" id ="vehicle1" name="vehicle1" value="Bike">
<label for="vehicle1">I have a Bike</label><br>
<input type="checkbox" id ="vehicle2" name="vehicle2" value="Car">
<label for="vehicle2">I have a Car</label><br>
<input type="checkbox" id ="vehicle3" name="vehicle3" value="Plane">
<label for="vehicle3">I have a Plane</label><br>
<label for="Food">Choose Food:</label><select name="Food" if="Food">
<option value="Cake">Cake</option>
<option value="Coke">Coke</option>
<option value="Lays">Lays</option>
</select>
<br><br>
<button type="Submit" value="Submit">Submit</button>
<button type="Reset" value="Reset">Reset</button>
</fieldset>
</form>
</body>
</html>

```

The screenshot shows a web browser window with the following details:

- Title Bar:** file:///home/student/190905513/q14.html
- Form Title:** OST Lab Dan
- Text Fields:**
 - First Name: Danish
 - Last Name: Eqbal
 - Password: (Redacted)
- Radio Buttons (Gender):**
 - Male (Selected)
 - Female
 - Others
- Checkboxes (Vehicle Ownership):**
 - I have a Bike (Selected)
 - I have a Car (Selected)
 - I have a Plane (Not Selected)
- Select Box (Food Choice):** Choose Food: Lays
- Buttons:** Submit, Reset

15. Making the HTML5 page more stylish using CSS

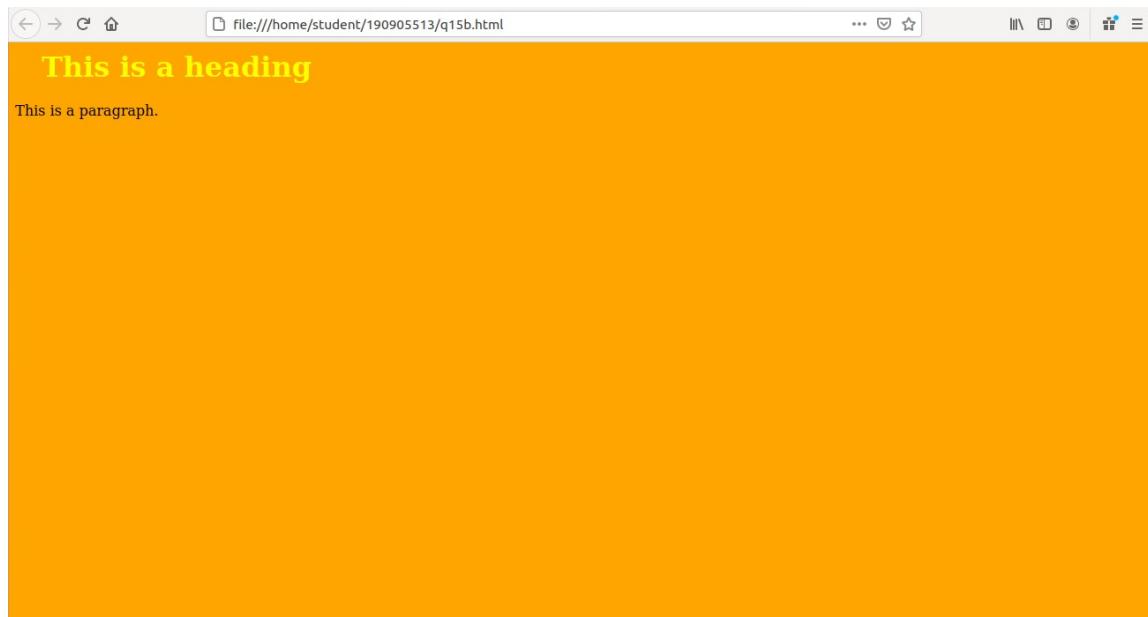
- Using inline style - style attribute

```
<html>
<head>OST Lab</head>
<title>Forms</title>
<body>
<h1 style="color:blue;text-align: center;">Heading</h1>
<p style="color:red;">Paragraph</p>
<p style="color: red; font-size: 20px;">This is my first paragraph.</p>
<p>This is my second paragraph.</p>
</body>
</html>
```



- Using embedded style

```
<html>
<head>
<style media = "all">
body {
background-color: orange;
}
h1 {
color: yellow;
margin-left: 30px;
}
</style>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```



- Link to an external style sheet – Link element

```
student@dslab:~/190905513$ gedit mystyle.css
student@dslab:~/190905513$ gedit q15c.html
student@dslab:~/190905513$ █
```

//mystyle.css

```
body {
    background-color: lightblue;
}
```

h1 {

```
    color: navy;
    margin-left: 20px;
}
```

//HTML file q15c.html

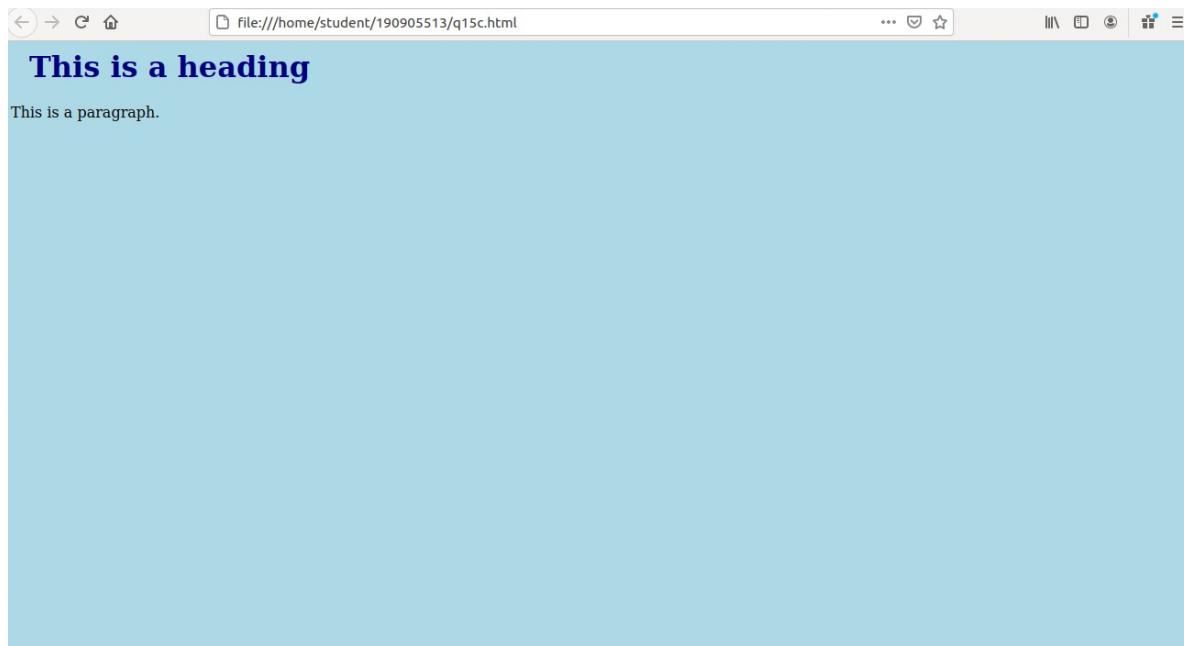
```
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>
```

<h1>This is a heading</h1>

<p>This is a paragraph.</p>

</body>

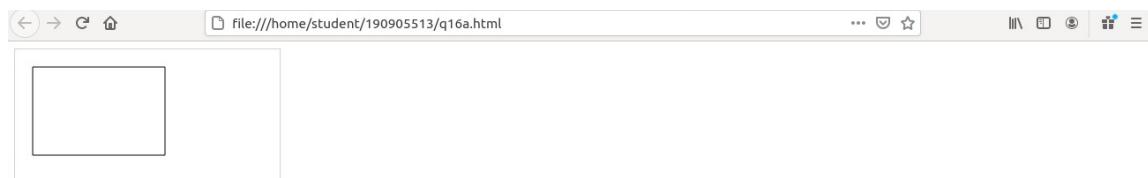
</html>



16. HTML5 Canvas

- Embedding canvas into an HTML document
- Drawing a line, circle, rectangle, arc onto the canvas
- Filling the colour into shapes on canvas

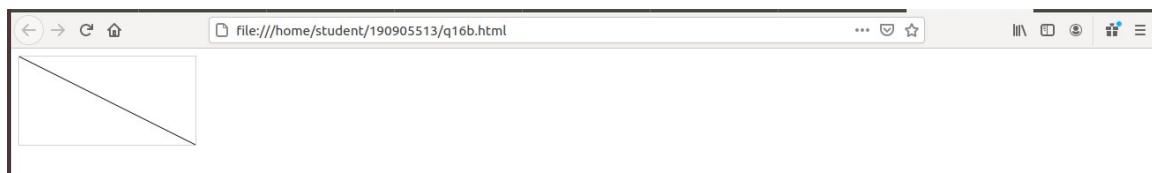
```
<html>
<body>
<canvas id="myCanvas" width="300" height="150" style="border:1px solid #d3d3d3;">
Your browser does not support the HTML5 canvas tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.rect(20, 20, 150, 100);
ctx.stroke();</script>
</body>
</html>
```



```

<html>
<body>
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">
Your browser does not support the HTML canvas tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0,0);
ctx.lineTo(200,100);
ctx.stroke();
</script></body>
</html>

```



```

<html> <body>
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">
Your browser does not support the HTML canvas tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.arc(95,50,40,0,2*Math.PI);
ctx.stroke();
</script> </body> </html>

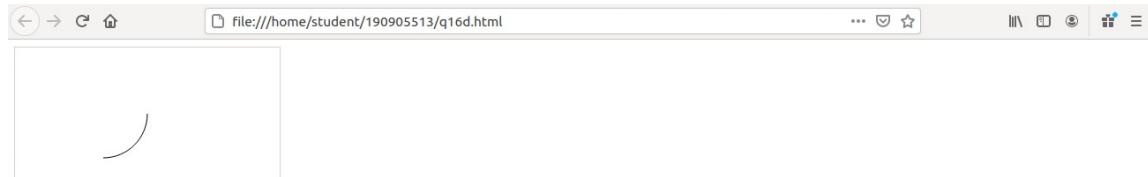
```



```

<html>
<body>
<canvas id="myCanvas" width="300" height="150" style="border:1px solid #d3d3d3;">
Your browser does not support the HTML5 canvas tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.arc(100, 75, 50, 0, 0.5 * Math.PI);
ctx.stroke();
</script>
</body>
</html>

```



```

<html>
<body>
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;">
Your browser does not support the HTML5 canvas tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
var grd = ctx.createLinearGradient(0,0,200,0);
grd.addColorStop(0,"red");
grd.addColorStop(1,"white");
ctx.fillStyle = grd;
ctx.fillRect(10,10,150,80);
</script>

```

```
</script>
```

```
</body>
```

```
</html>
```

