W1. Create a simple Mobile Website using jQuery Mobile for our college Pune Institute of Computer Technology(PICT).

C1) Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

W2. Create a simple Mobile Website using jQuery Mobile for our college Pune Institute of Computer Technology(PICT).

C2) Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

W3) Write a JavaScript Program to get the user registration data and push to array/local storage and data list in the new page.

C3) Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

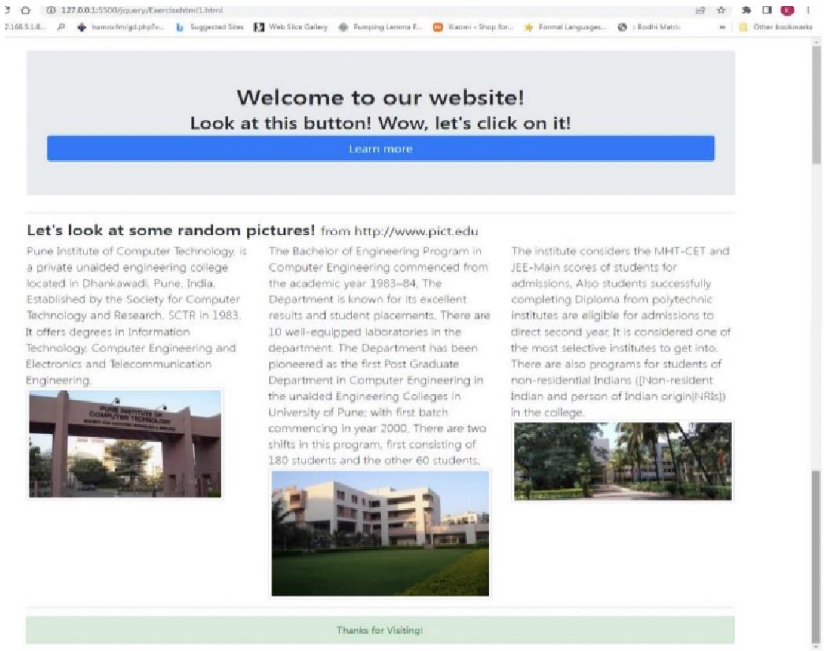
W4) Write a JavaScript Program to get the user registration data and push to array/local storage and data list in the new page.

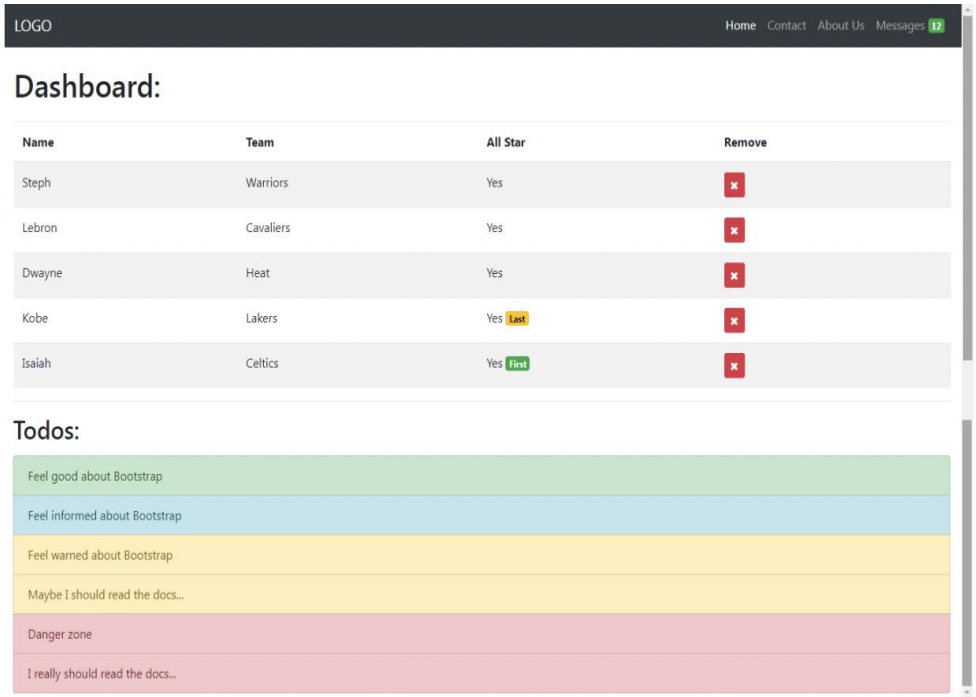
C4) Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

W5. Write a JavaScript Program to get the user registration data and push to array/local storage and data list in the new page.

C5. Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

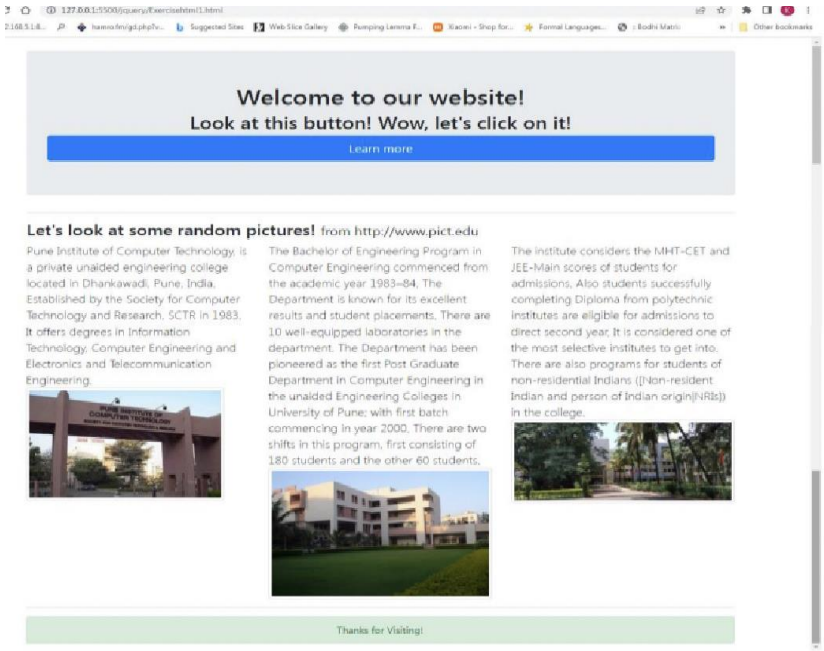
W6) Design the following Responsive websites using HTML, CSS and Bootstrap.

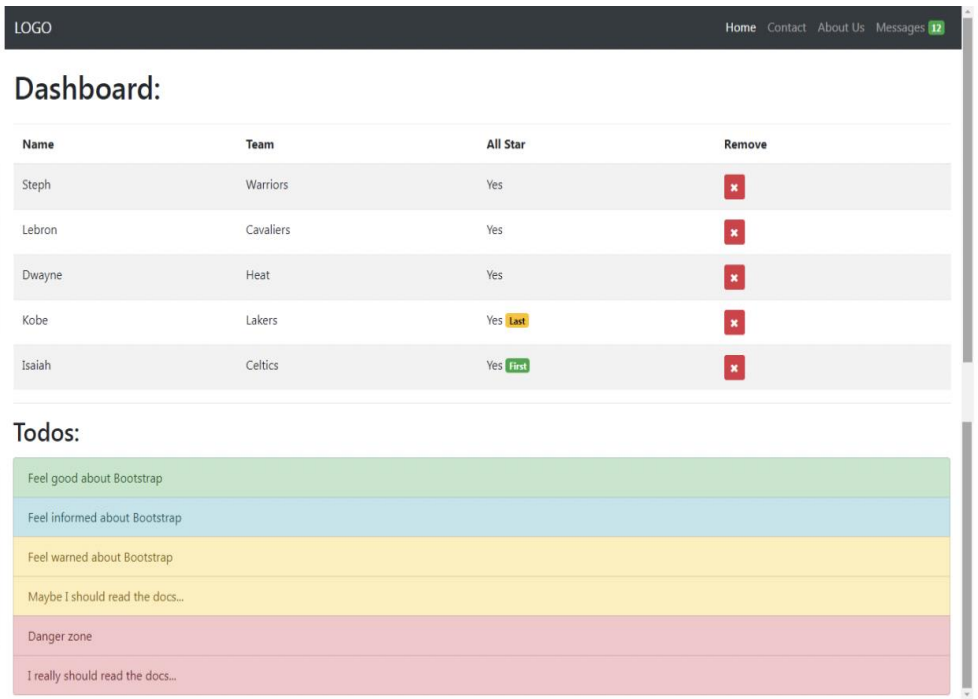




C6) Design and develop custom Application using Sales-force Cloud.

W7) Design the following Responsive websites using HTML, CSS and Bootstrap.





C7) Design and develop custom Application using Sales-force Cloud.

W8) Perform following tasks using nodejs, Expressjs and MongoDB. Following operation should be performed in Nodejs and Expressjs only.

a) Create a Database called music

b) Create a collection called song details

c) Insert array of 5 song documents in above Collection. [Document should have following field: Songname, Film, Music_director, singer]

d) Display total count of documents and List all the documents in the browser.

e) List specified Music Director songs.

f) List specified Music Director songs sung by specified Singer

g) Delete the song which you don't like.

h) Add a new song which is your favorite.

i) List Songs sung by Specified Singer from specified films.

j) Update the document by adding Actor and Actress name.

k) Display the above data in Browser in tabular format.

Song Name	Film Name	Music Director	Singer	Actor	Actress
ABC	DEF	GHI	JKL	MNO	PQR

C8. Design and develop Student Management System using Salesforce Cloud.

W9. Perform following tasks using nodejs, Expressjs and MongoDB. Following operation should be performed in Nodejs and Expressjs only.

l) Create a Database called music

m) Create a collection called song details

n) Insert array of 5 song documents in above Collection. [Document should have following field: Songname, Film, Music_director, singer]

o) Display total count of documents and List all the documents in the browser.

p) List specified Music Director songs.

q) List specified Music Director songs sung by specified Singer

r) Delete the song which you don't like.

s) Add a new song which is your favorite.

t) List Songs sung by Specified Singer from specified films.

u) Update the document by adding Actor and Actress name.

v) Display the above data in Browser in tabular format.

Song Name	Film Name	Music Director	Singer	Actor	Actress
ABC	DEF	GHI	JKL	MNO	PQR

C9. Design and develop Student Management System using Salesforce Cloud.

W10. Perform following tasks using nodejs, Expressjs and MongoDB.
Following operation should be performed in Nodejs and Expressjs only.

w) Create a Database called music
x) Create a collection called song details
y) Insert array of 5 song documents in above Collection. [Document should have following field: Songname, Film, Music_director, singer]
z) Display total count of documents and List all the documents in the browser.
aa) List specified Music Director songs.
bb) List specified Music Director songs sung by specified Singer
cc) Delete the song which you don't like.
dd) Add a new song which is your favorite.
ee) List Songs sung by Specified Singer from specified films.
ff) Update the document by adding Actor and Actress name.
gg) Display the above data in Browser in tabular format.

Song Name	Film Name	Music Director	Singer	Actor	Actress
ABC	DEF	GHI	JKL	MNO	PQR

C10. Design and develop Student Management System using Salesforce Cloud.

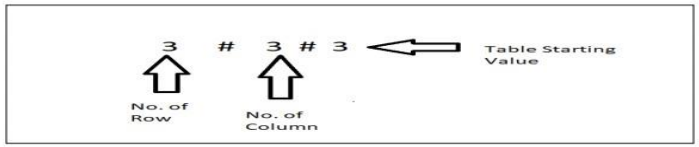
W11. Perform following tasks using nodejs, Expressjs and MongoDB. Following operation should be perform in Nodejs and Expressjs only.

a) Create a Database called student.
b) Create a collection called studentmarks
c) Insert array of documents in above Collection. [Document have following field: Name, Roll_No, WAD_Marks, CC_Marks, DSBDA_Marks,CNS_Marks,AI_marks]
d) Display total count of documents and List all the documents in browser.
e) List the names of students who got more than 20 marks in DSBDA Subject in browser.
f) Update the marks of Specified students by 10.
g) List the names who got more than 25 marks in all subjects in browser.
h) List the names who got less than 40 in both Maths and Science in browser.
i) Remove specified student document from collection.
j) Display the Students data in Browser in tabular format.

Name	**Roll No**	**WAD**	**DSBDA**	**CNS**	**CC**	**AI**
ABC	111	25	25	25	25	25

C11. Design and develop Student Management System using Salesforce Cloud.

W12. Create two input boxes and one button. In Input box you can provide input like 3#3#3 and 3#3#2



Create 2 Table having row and column mentioned in two different Input boxes
If first No. of Row is same as second No. of Row and first No. of Column is same as second No. of Column, then show third table. The last no. which is present in input value.
Start Printing table vertically from that no.

3	4	5
6	8	10
9	12	15

2	3	4
4	6	8
6	9	12

If first cell value of first table is equal to first cell value of second table, then print same no. else print multiplication of both no. in Third Table.
In third table if any cell value repeats then give same background colour else give new one

6	12	20
24	48	80
54	144	180

Example 2. In Input box you can provide input like 3#3#3 and 3#3#3

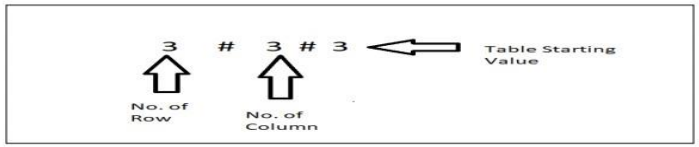
3	4	5
6	8	10
9	12	15

3	4	5
6	8	10
9	12	15

3	4	5
6	8	10
9	12	15

C12. Design and develop Student Management System using Salesforce Cloud.

W13. Create two input boxes and one button. In Input box you can provide input like 3#3#3 and 3#3#2



Create 2 Table having row and column mentioned in two different Input boxes
If first No. of Row is same as second No. of Row and first No. of Column is same as second No. of Column, then show third table. The last no. which is present in input value.
Start Printing table vertically from that no.

3	4	5
6	8	10
9	12	15

2	3	4
4	6	8
6	9	12

If first cell value of first table is equal to first cell value of second table, then print same no. else print multiplication of both no. in Third Table.
In third table if any cell value repeats then give same background colour else give new one

6	12	20
24	48	80
54	144	180

Example 2. In Input box you can provide input like 3#3#3 and 3#3#3

3	4	5
6	8	10
9	12	15

3	4	5
6	8	10
9	12	15

3	4	5
6	8	10
9	12	15

C13. Design and develop Student Management System using Salesforce Cloud.

W14. Use the Bootstrap grids classes to create this webpage with a sidebar and main content:

- Use the Bootstrap content CSS classes to add a quote, table, and image thumbnails to the page.

- Use the Bootstrap components classes to add an alert, search form to the page.

play around with adding more components or styling!

C14. Design and deploy a web application in a PaaS environment.

W15. Use the Bootstrap grids classes to create this webpage with a sidebar and main content:

- Use the Bootstrap content CSS classes to add a quote, table, and image thumbnails to the page.

- Use the Bootstrap components classes to add an alert, search form to the page.

play around with adding more components or styling!

C15. Design and deploy a web application in a PaaS environment.

W16. Use the Bootstrap grids classes to create this webpage with a sidebar and main content:

- Use the Bootstrap content CSS classes to add a quote, table, and image thumbnails to the page.

- Use the Bootstrap components classes to add an alert, search form to the page.

play around with adding more components or styling!

C16. Design and deploy a web application in a PaaS environment.

W17. Create your own webpage using HTML, CSS and Bootstrap called "index.html". It should have:

a. A title
b. Three or more types of headings
c. Some paragraphs
d. One or more ordered lists
e. One or more unordered lists
f. One or more tables
g. Some line-breaks
h. Some horizontal lines
i. Some comments
j. Various text styles: bold, italic, strike-through, underline

Make sure that the file is reasonably large: 2-3 screens worth at least. (You can add dummy content if you wish).

2. View the above HTML file on a browser, using a URL such as:
http://localhost/index.html OR double click on html file.
3. Learn to look at the HTML source from the browser window.

4. Now create some internal bookmarks and links using the "name=xyz" property and "href=#xyz". Type the URLs for these internal bookmarks directly onto the browser.
5. Now split your webpage into multiple HTML files, or equivalently create more HTML files. For example, you may have a separate HTML file describing your interests or educational background. Provide links from the main index.html to these other HTML files. Provide also some links to external websites (e.g. google).
6. Now move all the HTML files to another directory. Your links should still work (they should be relative links).
7. Embed one or more images in any of the HTML files. The images should reside within a sub-directory called "images".
8. Now include some styles within HTML tag elements. For example: table border, cell background, link color, text color, background color, list style, text font, etc.
9. Now include some of the earlier styles in the "style" tag of the HTML "head" section, instead of marking the styles in individual HTML tag elements.
10. Now have a separate css file for the above styles, so that the same can be included in multiple HTML files.
11. Insert some special characters such as &, <, >, etc. in your HTML file

C17. Design and deploy a web application in a PaaS environment.

W18. Create your own webpage using HTML, CSS and Bootstrap called "index.html". It should have:

k. A title
l. Three or more types of headings
m. Some paragraphs
n. One or more ordered lists
o. One or more unordered lists
p. One or more tables
q. Some line-breaks
r. Some horizontal lines
s. Some comments
t. Various text styles: bold, italic, strike-through, underline

Make sure that the file is reasonably large: 2-3 screens worth at least. (You can add dummy content if you wish).

12. View the above HTML file on a browser, using a URL such as: http://localhost/index.html OR double click on html file.
13. Learn to look at the HTML source from the browser window.

14. Now create some internal bookmarks and links using the "name=xyz" property and "href=#xyz". Type the URLs for these internal bookmarks directly onto the browser.
15. Now split your webpage into multiple HTML files, or equivalently create more HTML files. For example, you may have a separate HTML file describing your interests or educational background. Provide links from the main index.html to these other HTML files. Provide also some links to external websites (e.g. google).
16. Now move all the HTML files to another directory. Your links should still work (they should be relative links).
17. Embed one or more images in any of the HTML files. The images should reside within a sub-directory called "images".
18. Now include some styles within HTML tag elements. For example: table border, cell background, link color, text color, background color, list style, text font, etc.
19. Now include some of the earlier styles in the "style" tag of the HTML "head" section, instead of marking the styles in individual HTML tag elements.
20. Now have a separate css file for the above styles, so that the same can be included in multiple HTML files.
21. Insert some special characters such as &, <, >, etc. in your HTML file

C18. Design and deploy a web application in a PaaS environment.

W19. Create your own webpage using HTML, CSS and Bootstrap called "index.html". It should have:

u. A title
v. Three or more types of headings
w. Some paragraphs
x. One or more ordered lists
y. One or more unordered lists
z. One or more tables
aa. Some line-breaks
bb. Some horizontal lines
cc. Some comments
dd. Various text styles: bold, italic, strike-through, underline

Make sure that the file is reasonably large: 2-3 screens worth at least. (You can add dummy content if you wish).

22. View the above HTML file on a browser, using a URL such as: http://localhost/index.html OR double click on html file.
23. Learn to look at the HTML source from the browser window.

24. Now create some internal bookmarks and links using the "name=xyz" property and "href=#xyz". Type the URLs for these internal bookmarks directly onto the browser.
25. Now split your webpage into multiple HTML files, or equivalently create more HTML files. For example, you may have a separate HTML file describing your interests or educational background. Provide links from the main index.html to these other HTML files. Provide also some links to external websites (e.g. google).
26. Now move all the HTML files to another directory. Your links should still work (they should be relative links).
27. Embed one or more images in any of the HTML files. The images should reside within a sub-directory called "images".
28. Now include some styles within HTML tag elements. For example: table border, cell background, link color, text color, background color, list style, text font, etc.
29. Now include some of the earlier styles in the "style" tag of the HTML "head" section, instead of marking the styles in individual HTML tag elements.
30. Now have a separate css file for the above styles, so that the same can be included in multiple HTML files.
31. Insert some special characters such as &, <, >, etc. in your HTML file

C19. Design and deploy a web application in a PaaS environment.

W20. Create Following form and Facebook webpage:

C20. Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

W21. Create Following form and Facebook webpage:

C21. Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

W22. Create Following form and Facebook webpage:

C22. Find a procedure to transfer two text files from one virtual machine to another virtual machine. Also, add some text in the file and display it before and after transfer on both the VMs.

W23. Create a Node.JS Application which serves a static website for applications like **Art Gallery (pinterest) or restaurant** or any other application.

C23. Design and deploy a web application in a PaaS environment.

W24. Create a Node.JS Application which serves a static website for applications like **Art Gallery (pinterest) or restaurant** or any other application.

C24. Design and deploy a web application in a PaaS environment.

W25. Create an Angular application for **Course Enrollment System** where the process begins with Registration of User followed by User Login. Once the process is done User Data is displayed on Profile Component

C25. Design and develop custom Application using Sales-force Cloud.