COMS 106: Comprehensive Program # 1 Assignment

Personal Homepage Design

Begin: Week 5.

Deadline: Week 12, Sunday, 11pm. (Please see the detailed due date information on the 'Course Schedule'. (Click the 'Course Schedule' within the 'Week 1 Module' within COMS 106 Canvas course.)

This is a mandatory project.

A) There are two versions: Students only need to work on the basic version project. However, if you like to work on advanced version project, you will earn additional 50 points.

* About the advanced version:

1) The advanced version includes JavaScript and JQuery which we will cover them in the chapter 14. I have produced streaming lectures to assist you in these two areas. They will be posted within the weekly modules for you to learn the skills and materials in advance. However, if you choose not to work on the advanced project, that's fine because it won't hurt your grade.

2) Under the 'Family' tag, you need to create two sub links. One is called: My Family. Please introduce your family on this page. Please see the sample advanced version project.

Here are the samples of these two versions:

1) Basic version project: 100 points
http://www.cs.iastate.edu/~cs106hp1/basic/new5/homepage.html

2) Advanced version project: 100 points + 50 additional extra credit points.
http://www.cs.iastate.edu/~cs106hp1/advance/new5/homepage.html

To apply what you have learned in this course to design and build your personal homepage.

What can I have on my web page?

* You can share your resume and educational background.

* You can post personal information and pictures of your family or friends.

* You can share your hobbies.

* You can embed favorite music links or any useful video clips.

* You can share your special talents etc..

* This list is not limited and you are free to add anything relevant to the personal home page.

I have produced the 'Streaming Lectures' to show you how to build the sample Personal Homepage Project. I will release some units to you every week. They are posted under the "Homepage Design Project Streaming Lectures (Unit #)" in the "Week # Module". (e.g. For the week 6, the "Homepage Design Project Streaming Lectures (Units #1,2,3) have been posted within the "Week 6 Module" under the "Modules" on the course menu).

I will add more new units of the Homepage Design Project every week. Please watch the streaming lectures and continue to build your own Personal Homepage project.

The 'Homepage Design Project Streaming Lectures' will be posted in the following weeks:

* Weeks 5 and 6: "Homepage Design Project Streaming Lectures (Units 1,2,3)

* Midterm Week: "Homepage Design Project Streaming Lectures (Units 4,5)

* Week 7: "Homepage Design Project Streaming Lectures (Units 6, 7)

* Week 8: "Homepage Design Project Streaming Lectures (Unites 8, 9)

* Weeks 9, 10,11, 12: "Homepage Design Project Streaming Lectures (All Units)

Showcase your creativity and programming skills to design this project.

Enjoy this fun project!!

A) Homepage project design requirements:

Here are the suggested outlines of your homepage: Your project should include a welcome page (Home) and at least 4 sub pages, such as:
* Educational background

* About your family or friends

* Your hobbies or your special talents, if you have any

* Favorite links (e.g. favorite music links, cartoon links, movie links etc.)
* Your contact information

* Others
   Note: You are welcome to add more pages with content relevant to personal homepage.

1) Your webpages should include HTML5 elements, such as :
   a) Links (navigations, hyperlinks, etc.)
   * Link to another Web page in the same Web site
   * Link to external web site
   * Navigation Bar
   b) Images (image maps, image links, etc.)
   c) Tables (at least 2 tables with different styles)
   d) Use embedded style sheets, inline styles, and classes to change the format of text, links, images, and headings.
   e) Use a bullet list. (either order or unorder bulleted list with one of the bullet styles).
   f) Use external style sheets
   e) Use Multimedia Content

2) CSS should be reflected in your homepage especially in font, background style, division size and positions.

3) In addition, you may include more fancy elements such as audio, video, JavaScript, jQuery, PHP, SQL and MySQL functions etc..

4) Students should design and code their own homepage programs. Students should not use website templates from other companies (such as: wix.com etc.)

1. All links should be working fine.

2. All images are displayed on the page.

3. Content in each page is neatly organized.

Most important criteria for the project: ***"Your webpages should be clear and user-friendly".***

==D) Browser compatibility:==

Students must test your web pages through different browsers (e.g. Microsoft Edge, Firefox, Chrome, Safari etc.) before you submit your project.

==D) Copyright:==

Make sure you do not include any photos, music or videos that do not belong to you (no professional photos or videos with copy marks etc.) or inappropriate photos or videos. Photos or videos including illegal acts, nudity etc. will disqualify your web page from eligibility for credit and will be removed from this project.

Please read the weekly memos for more detail information.

Sample Homepage Design Project Web Page: Here is a sample Homepage Design Project Web Page for your reference.

http://www.cs.iastate.edu/~cs106hp1/basic/new5/homepage.html

Help for students:

If you have any questions, contact your group TA through Canvas email or come to the face to face help desk in B17 Atanasoff Hall (Basement).
Please see the 'Help Desk Schedule' within the 'Week 1 Module' in COMS 106 Canvas course.

If you are not available during these time slots, you also can make additional appointments with your group TA.

Below are the instructions regarding the project submission.

I. How to Submit the Homepage Design Project via 'Assignment Drop Box'?

You need to submit your project via the 'Assignments' drop box: (You must complete the following two steps in order to be able to earn the credit.)

1) Upload all of files to the CS CourseWeb server:
a) After you upload all of your homepage files to the CS CourseWeb server, then it will create a link for your homepage.

For example:

https://courseweb.cs.iastate.edu/~jsmith/homepage.html

b) After that, you will need to create a new Word document, post your homepage link on the Word document and save it as:
'G#_Yourfirstlastname_Homepagelink'.

In the Word document, you need to type in your group #, first name, last name and your homepage link.

For example:

You type in the following contents on the Word document:

"G1_JenniferSimth

Homepage link:

https://courseweb.cs.iastate.edu/~jsmith/homepage.html

Best regards

Jennifer Smith"

(Note: Check 3 times to ensure your homework link is a workable link and all of the pages are working fine.)

c) Save the file as:"G1_JenniferSmith_Homepagelink". Include this file when you submit all of your homepage files via the assignment drop box for grading.
Important Note: You must follow this file convention when you save your Word document which contains your homepage link. Otherwise, your group TA may not be able to fine your file.
('G#_Yourfirstlastname_Homepagelink')

2) Create a new folder and put all of your homepage files along with your homepage link file into this new folder. Zip all of your homepage files and your homepage link file (word document) together, save it as 'Your group #_Firstname+LastnameCP1.zip'. (e.g. 'G1_JenniferSmithCP1.zip.')

c) Submit the zip file via the 'CP1' assignment drop box. (Click the 'Assignments' on the COMS 106 course menu, click the 'CP1' drop box to submit the zip file.

    * Please check and test it at least 3 times to ensure the link works fine. e.g.

    https://courseweb.cs.iastate.edu/~NETID/homepage.html

Note: You need to check your link. It depends on how you name your first page. For example:
a) If your first page named: home.html, then the link should be:

    https://courseweb.cs.iastate.edu/~jsmith/home.html

b) If your first page named: jsmithhome.html, then the link should be:

    https://courseweb.cs.iastate.edu/~jsmith/jsmithhome.html

Important Notes:

1) Please ZIP all of the files (including your homepage link file) together before you submit your homework via Assignment drop box.

2) DO NOT submit the PDF file format when you submit your homework.

3) Please DO NOT WAIT until the last minutes to submit the homework because the system will be very slow and you may miss the deadline.

4) Please submit your homework before Sunday, 11pm of the due date. The assignment drop box will be closed after 11pm.

5) You must submit the project via 'Assignment' drop box (Click the 'Assignment' on the Canvas course menu, click the 'CP1' link) in Canvas before the deadline. Please don't submit it to the instructor (Dr. Chang).

6) Please keep a backup copy of all of your files in your local machine as well.

II. How to Configure Home Drive Access for CourseWeb

Configuring Home Drive Access for CourseWeb

The Computer Science student website for class course work is now located at:
http://courseweb.cs.iastate.edu/~NETID

This site uses a folder called CSWEB on the root of your Home Drive (\\my.files.iastate.edu\users\NETID)

Notes:
1) This process will prepare your home drive for use and **MUST** be done via a Windows computer.
2) This process must be done while connected to the campus network (VPN).
3) The process is applicable for windows machine only. If you use a Mac or Linux machine, please use a windows machine on campus or use a remote connection to a windows machine.
4) **Please keep backup copies for all of your files because the department of computer science only can facilitate your web pages for one semester.**

Configuration of Home Drive Folder:
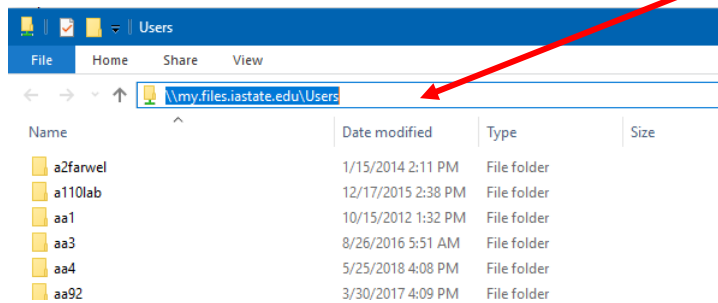
1. In Windows, using Windows Explorer (or Click Start then, Run (Windows + R))
    (Note: If you are off campus, please connect to ISU VPN first.)

    Type in

    \\my.files.iastate.edu\users

    > Type in
    >
    > **\\my.files.iastate.edu\users**

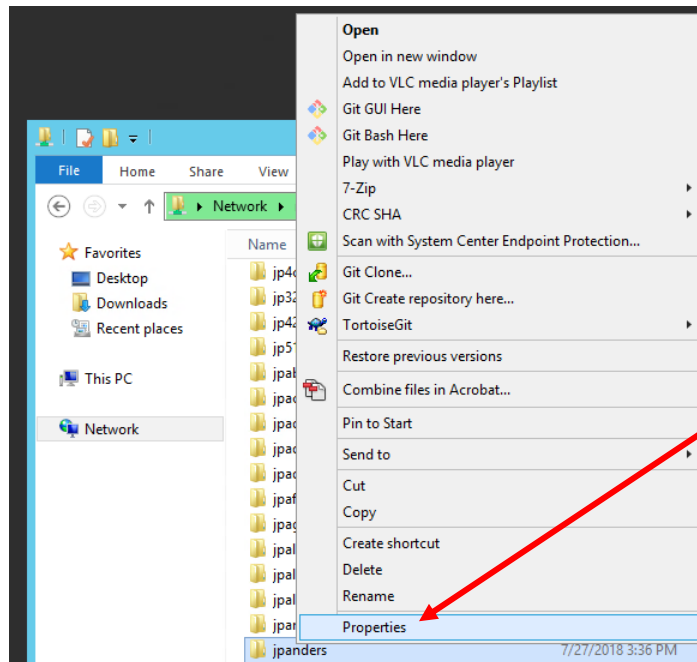 | Name | Date modified | Type | Size |
    |------|---------------|------|------|
    | a2farwel | 1/15/2014 2:11 PM | File folder | |
    | a110lab | 12/17/2015 2:38 PM | File folder | |
    | aa1 | 10/15/2012 1:32 PM | File folder | |
    | aa3 | 8/26/2016 5:51 AM | File folder | |
    | aa4 | 5/25/2018 4:08 PM | File folder | |
    | aa92 | 3/30/2017 4:09 PM | File folder | |

    \\my.files.iastate.edu\Users

2. Scroll down and find your NETID (or use the search bar in the top right).

    > **To find your NETID, type in your NETID in the search bar in the top right hand side of the screen**

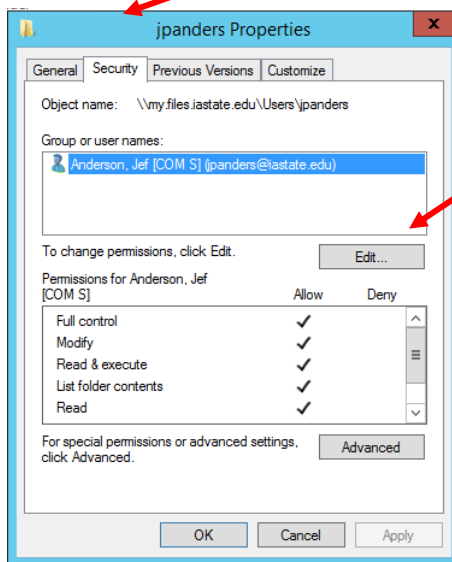3. Right click your NETID and choose 'Properties' at the bottom.

Open
Open in new window
Add to VLC media player's Playlist
Git GUI Here
Git Bash Here
Play with VLC media player
7-Zip
CRC SHA
Scan with System Center Endpoint Protection...
Git Clone...
Git Create repository here...
TortoiseGit
Restore previous versions
Combine files in Acrobat...
Pin to Start
Send to
Cut
Copy
Create shortcut
Delete
Rename
Properties

Right click your NETID and choose 'Properties'.

You will see your NETID here

4. In the Properties window, click the 'Security' tab, then click 'Edit'.

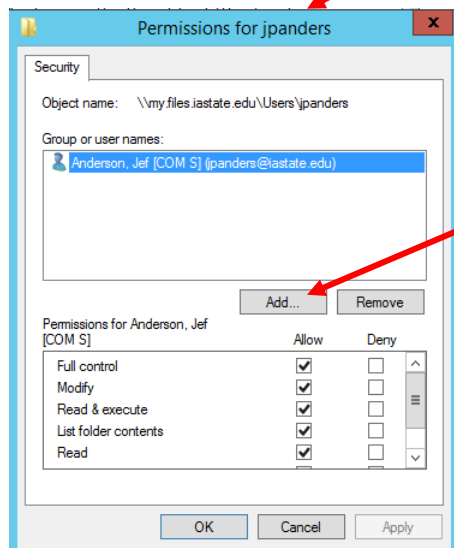**Click 'Security' tab**

Click 'Edit'

jpanders Properties

General | Security | Previous Versions | Customize

Object name: \\my.files.iastate.edu\Users\jpanders

Group or user names:

Anderson, Jef [COM S] (jpanders@iastate.edu)

To change permissions, click Edit.   Edit...

Permissions for Anderson, Jef
[COM S]   Allow   Deny

Full control   ✓
Modify   ✓
Read & execute   ✓
List folder contents   ✓
Read   ✓

For special permissions or advanced settings,   Advanced
click Advanced.

OK   Cancel   Apply

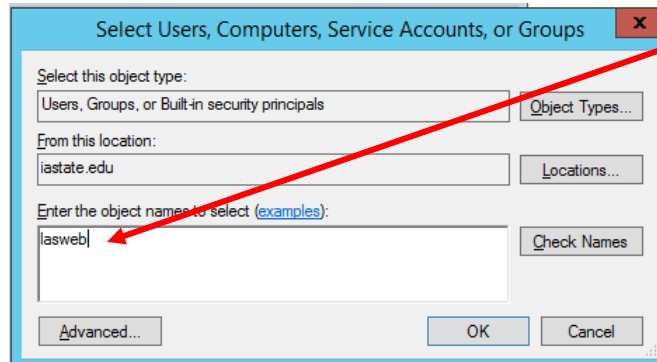5. You should now see a window that says Permissions for <NETID>
6. Click the 'Add' button.

Permission for NETID Window (your NETID).

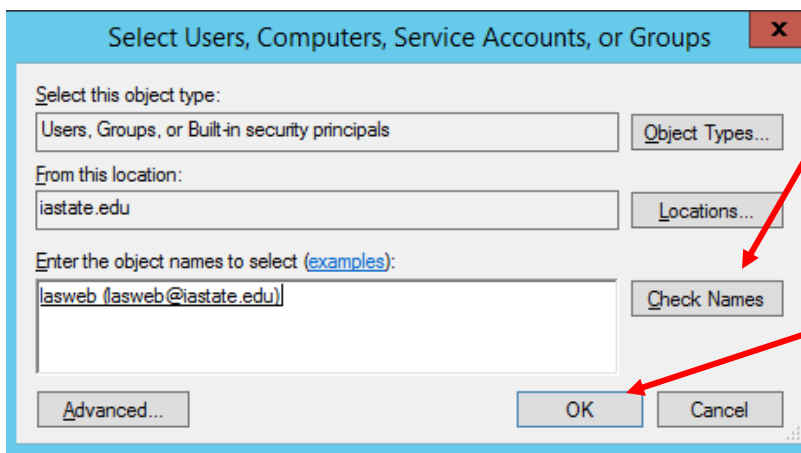**Click 'Add'**

Permissions for jpanders

Security

Object name: \\my.files.iastate.edu\Users\jpanders

Group or user names:

Anderson, Jef [COM S] (jpanders@iastate.edu)

Add...   Remove

Permissions for Anderson, Jef [COM S]   Allow   Deny

Full control   ☑   ☐
Modify   ☑   ☐
Read & execute   ☑   ☐
List folder contents   ☑   ☐
Read   ☑   ☐

OK   Cancel   Apply

7. In the 'Select Users, Computers, Service Accounts, or Groups' dialog box, type in **lasweb**
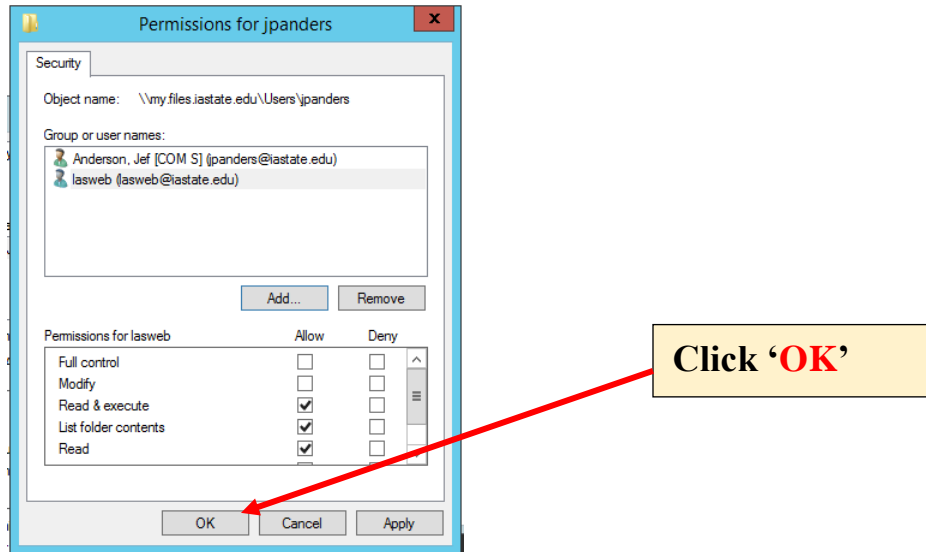
Type in lasweb

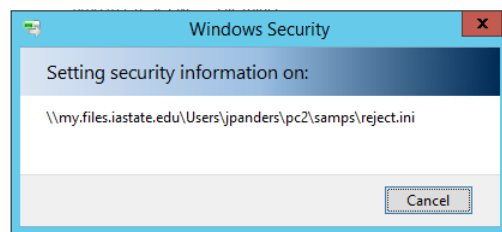8. Click 'Check Names' and the click 'OK'.

Click 'Check Name'

Click 'OK'

9. You should be back to the Permissions for <NETID> window.

Click 'OK'

10. Click 'OK'.

11. You should see a 'Windows Security' screen pop up similar to the one below:

Notes:
a) This process may take approximately 30 minutes or longer to complete. It depends how many files you have already stored on your NETID account.
b) Please do not wait until the last day to upload files to CS Course Web because it will be very slow and you may miss the deadline.

12. Once that completes, click 'OK' on the NETID Properties window.

13. Go into your home drive folder and create a new folder called '**csweb**'.

14. Place your html files, etc. in this folder and it will be viewable on the site now.

15. You can view your Homepage Design Project by opening a web browser and navigating to
https://courseweb.cs.iastate.edu/~NETID/homepage.html
where NETID is your NETID. (e.g.
https://courseweb.cs.iastate.edu/~jsmith/homepage.html

Note: You need to check your link. It depends on how you name your first page. For example:
a) If your first page named: home.html, then the link should be:
https://courseweb.cs.iastate.edu/~jsmith/home.html

b) If your first page named: jsmithhome.html, then the link should be:
https://courseweb.cs.iastate.edu/~jsmith/jsmithhome.html

Important Notes:
1) DO NOT upload a zip file. You must unzip them, then upload all of the files to the 'csweb' folder.
2) It's case sensitive. Please check your link to ensure it works. (Note: It's easier to use all lowercase.)

Congratulations! You have just uploaded your Homepage Project to the cs courseweb.

Notes:
1) You must test your web page through different browsers (e.g. Microsoft Edge, Firefox, Chrome, Safari etc.) before you submit the homepage link file. If the link doesn't work, you won't earn credit for the project.

2) When you test it, you must be sure you connect to ISU VPN. (It won't work if you do not connect to ISU VPN.)
* Please understand that it's the department's policy that all of students' homework should be protected within ISU network. I have no control over this issue.

If you have problems when you work on the projects, please ask your group TA for assistance in advance.

III. Program Hints for Sample Homepage Design Project for your reference

http://www.cs.iastate.edu/~cs106hp1/basic/new5/homepage.html

Brief introduction of this program:

This program is a homepage design project website which includes HTML5 and CSS. This is just a sample project for your reference. Your project will be different with this one.

Homepage.html

```
01:
02:
03:
04:
05:
06:
07:
08: <script>
09: function overButton(img){
10:  buttonImg = "buttons/button_"+img+"_over.png"
11:  document.getElementById(img).src = buttonImg}
12: function upButton(img) {
13:  buttonImg = "buttons/button_"+img+"_up.png"
14:  document.getElementById(img).src = buttonImg}
15: </script>
16:
17: <body>
18: <div id="container">
19:  <div id="main">
20:   <div class="header">
21:    <div class="icon">
22:     <a href="homepage.html"><img src="material/icon1.png"></a>
23:    </div>
24:   <div class="navbar" style="margin-right:50px">
25:      <table border="0" style="margin:auto; margin-right:auto";>
26:       <tr>
27:        <td><a href="homepage.html" style="cursor:pointer;" type="button"><img id="home"
src="buttons/button_home_up.png"  onMouseOver="overButton('home')"></a></td>
28:
29:
30:
31:       <td><a href="contact.html"><img id="contact"  src="buttons/button_contact_up.png"
onMouseOut="upButton('contact')" onMouseOver="overButton('contact')"></a></td>
32:
33:
34:
35:
36:
37:
38:
39:
40:
:
:
66:
67:
68:
69:
70:
71:
72:
```

overButton(img) function is used to display a new image, whose name is "buttonImg", at the same position of "img", when the mouse hovers on "img".

img is an object, therefore no quotation

upButton(img) function is used to display a new image, whose name is "buttonImg", at the same position of "img", when the mouse leaves "img".

Objects "container" and "main", classes "header" and "icon" are defined in homepage.css

"Margin: auto" lets the browser calculate the margin

The cursor is a pointer and indicates a link

The function "overButton" is called when the mouse is over the image

73: `<div align="center">`

74: You are the

75: `<img src='http://www.hit-counter-html-code.com/c.php?d=9&id=140186&s=16' border='0' height="16" width="90" title='counter'>` visitors to this site.

76: `</div>`

77:

78:

aboutme.html

01:

02:

03:

04:

05:

06: `<link rel="shortcut icon" href="material/logo.png">`

07:

:

:

16:

17:

18:

19:

20:

21:

22: `<div class="navbar" style="margin-right:50px";>`

23:

:

:

29:

30:

31:

32:

33:

34:

35: `<div class="maincontent">`
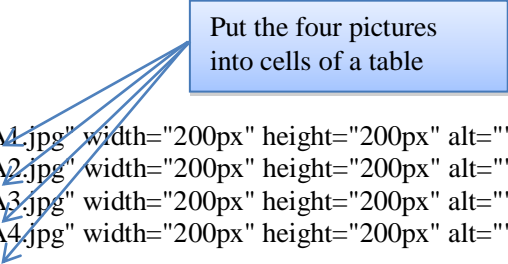
36: `<h2 class="contentTitleh2">My Profile</h2>`

37: `<div class="inner_Content">`

38: `<div class="phote" style = "margin-left:auto; margin-right:0;" ><img src="profile/profilephoto.jpg" alt="You should see my Photo Here!" width="350" height="700.11"></div>`

39:

40:

:

:

97:

98:

99:

album.html

```
01:
02:
 :
 :
30:
31:
32:
33:
34:
35:
36:
37:  <tr>
38:  <td><img src="album/California/CA4.jpg" width="200px" height="200px" alt="" /></td>
39:  <td><img src="album/California/CA2.jpg" width="200px" height="200px" alt="" /></td>
40:  <td><img src="album/California/CA3.jpg" width="200px" height="200px" alt="" /></td>
41:  <td><img src="album/California/CA4.jpg" width="200px" height="200px" alt="" /></td>
42:  <caption>California</caption>
43:  </tr>
44:
45:
46:
47:
 :
 :

65:
```

Put the four pictures
into cells of a table

contact.html

```
01:
02:
03:
04:
05:
06:
07: <script>
08:
09:
10:
11:
12:
13:
14: function sendMail(){
15:   var link = "mailto:chenya@iastate.edu"
16:         + "?subject=From " + escape(document.getElementById('name').value)+": "+escape(document.getElementById('subject').value)
17:         + "&body=" + escape(document.getElementById('comment').value);
18:   window.location.href = link;}
19: function reset(){
20:   document.getElementById('name').value="";
21:   document.getElementById('subject').value="";
22:   document.getElementById('comment').value="";}
23: </script>
24:
25:
26:
27:
28:
29:
30:
 :
 :
45:
46:
47:
48:
49:
50:
51:
52:
53: <input type="submit" name="send" value="Send" onclick="sendMail()"/>
54: <input type="submit" name="reset" value="Reset" onclick="reset()"/>
55:
56:
57:
58:
59:
```

This function is used to send an email

The function "escape" makes a string portable, so it can be transmitted across any network to any computer that supports ASCII characters.
This function encodes special characters, with the exception of: * @ - _ + . /

mailto: the receiver
subject: the subject of the email
body: the email body

Open a new window

This function is used to reset some elements on the page to be empty

The click event will call the

family.html

```
01:
:
:
55:
56:
57:
58:
59:
60:
61:
62:  <li class="regularChild">
63:  <img class="img1" style="margin-top: 5px;" src="family/family1.png" width="150" height="70"
alt="" />
64:   <ul class="secondLevelChild1">
65:       <li class="secondLevelRegularChild" >1. For Elise - Beethoven:</li>
66:       <li class="secondLevelRegularChild"><audio style="margin-left: 1px;" controls="controls" >
67:        <source src="video/elise.mp3" type="audio/mp3" />hihi</audio>
68:       </li>
69:       <br /><br />
70:       <li class="secondLevelRegularChild" >2. Moonlight - Beethoven:</li>
71:
72:
73:
:
83:
84:
85:       <source src="video/video2.mp4" type="video/mp4"/></video></li>
86:
87:
88:
89:
90:
91:
92:
93:
94:
95:
96:
97:
98:
99:
```

Classes "regularChild", "secondLevelChild1" and "secondLevelRegularChild" defined in

Put a video here. The video is in the folder "video" and with the name "video2.mp4".

special.html

```
01:
 :
 :
27:
28:
29:
30:
31:
32:
33:
34:
35:
36:
37:
38:
39:  <div class="inner_content">
40:  <ul id="hw" style="list-style: none;width: 80%;padding: 0; ">
41:  <li class="regularChild" style="background-color:transparent;">
42:     <h2 style="color: #00CCFF;">2014:</h2>
43:  </li>
44:
45:
 :
 :
87:
88:
89:
90:
91:
```

> The <ul> tag defines an unordered (bulleted) list. Use the <ul> tag together with the <li> tag to create unordered lists.

homepage.css

```
001:
002:
003: html, body, div, span, applet, object, iframe,h1, h2, h3, h4, h5, h6, p, blockquote, pre,a, abbr,
acronym, address, big, cite, code,del, dfn, em, img, ins, kbd, q, s, samp,small, strike, strong, sub, sup,
tt, var,b, u, i, center,dl, dt, dd, ol, ul, li,fieldset, form, label, legend,table, caption, tbody, tfoot, thead,
tr, th, td,article, aside, canvas, details, embed,figure, figcaption, footer, header, hgroup,menu, nav,
output, ruby, section, summary,time, mark, audio, video { margin: 0; padding: 0; border: 0; font-size:
100%; font: inherit; vertical-align: baseline; }
004:
005:
006:
007:
008:
009:
 :
 :
```

> Align the baseline of the element with the baseline of the parent element.

> Inherits the font property from its parent element.

22

```
022:
023:
024:
025:
026:
027: #main {margin-left: 0px; margin-top: 0px;overflow: hidden;width: 100%;}
028:
029:
030:
031:
032:
033:
034:
035:
036:
037:
038:
039:
040:
041:
042:
043:   .icon{
044:   float: left;
045:   margin-top: 50px;
046:   margin-left: 5%;
047:   }
048:
049:
:
:
062:
063:
064:
065:
066:
067:
068:
069: .Content1{
070: display:inline;
071: float:right;
072: font-family: Interstate,arial,helvetica,sans-serif;
073: color:black;
074: height: 400px;
075: width: 54.8784%;
076: }
077:
:
:
142:
143:
144:
145:
146:
```

The overflow is clipped, and the rest of the content will be invisible.

The element floats to the left.

Displays an element as an inline element (like <span>).

147:
148:
149: .footer{
150: font-family: Interstate,arial,helvetica,sans-serif;
151: font-size: 14px;
152: margin-top: 30px;
153: margin-bottom: 50px;
154: text-align: center;
155: width: 100%;
156: padding: 0;
157: color:black;
158: clear:both;

> No floating elements allowed on either the left or the right side.

159: }

mystyle.css

001: body, html {
002: margin: 0;
003: padding: 0;
004: height: 100%;
005: background-image: url("material/content/background1.jpg");
006: background-position-x: 100%;
007: background-position-y: 100%;
008: background-repeat:repeat;
009: font: inherit;
010: vertical-align: baseline;
011: width:100%;
012: min-width:300px;

> The min-width property is used to set the minimum width of an element.

013: }
014:
015:
016:
017:
018:
019:
020:
021:
022:
023: .header{
024: background-image: url("material/header/header3.jpg");
025: background-position-x: 50%;
026: background-position-y: 0px;
027: background-repeat:repeat-x;

> The background-position property sets the starting position of a background image.

028: background-color:#000;
029: overflow:hidden;
030: height:180px;
031: width:100%;

> The background image will be repeated both vertically and horizontally.

032: border-bottom: 2px solid gray
033: }
034:

035:
036:
:
:

226:
227:
228:
229:
230:
231:
232:
233:
234:
235:
236:
237:
238:
239:
240:
241:
242: #carousel img {display: block; opacity: .5; cursor: pointer;}
243:
244:
245:
246:

The opacity property sets the opacity level for an element. The opacity-level describes the transparency-level, where 1 is not transparent at all, 0.5 is 50% see-through, and 0 is completely transparent.

slider.css

01:
02:
03: .flex-container a:active,
04: .flexslider a:active,
05: .flex-container a:focus,
06: .flexslider a:focus {outline: none;}
07:
08:
09:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
20:
21:
22:
23:
24:

Specify the style of the link when it is active. A link becomes active when you click on it.

25:
26:
27:
28: .flexslider .slides > li {display: none; -webkit-backface-visibility: hidden;}
29:
30:
:
:
44:
45:
46:
47:
48:
49:
50:
51:
52:
53:
54: .flex-viewport { max-height: 2000px; -webkit-transition: all 1s ease; -moz-transition: all 1s ease; -o-transition: all 1s ease; transition: all 1s ease; }
55:
56:
57:
58:
59:
60:
61:
62:
63:
64:
65:
66:
67:
68:
69:
:
:
98:
99:

Hide the slides before the JS is loaded. Avoids image jumping.

Transitions are effects that let an element gradually change from one style to another.
-webkit: for chrome and safari
-moz: for firefox
-o-: for opera

VII. Program Hints for Advanced Homepage Design Project for your reference

http://www.cs.iastate.edu/~cs106hp1/advance/new5/homepage.html

Brief introduction of this program:

This program is a homepage design project website which includes HTML5, CSS, JavaScript and JQuery etc.. This is just a sample project for your reference. Your project will be different with this one.

Homepage.html

```
001:
002:
003:
004:
005:
006:
007:
008: <link rel="stylesheet" href="slider.css" type="text/css" media="screen" />
009: <script src="AlbumWWW/htmls/js/modernizr.js"></script>
010: <script src="AlbumWWW//jquery.min.js"></script>
011: <script defer src="AlbumWWW/jquery.flexslider.js"></script>
012: <script type="text/javascript" src="AlbumWWW/htmls/js/shCore.js"></script>
013: <script type="text/javascript" src="AlbumWWW/htmls/js/shBrushXml.js"></script>
014: <script type="text/javascript" src="AlbumWWW/htmls/js/shBrushJScript.js"></script>
015: <script src="AlbumWWW/htmls/js/jquery.easing.js"></script>
016: <script src="AlbumWWW/htmls/js/jquery.mousewheel.js"></script>
017: <script defer src="AlbumWWW/htmls/js/demo.js"></script>
018:
019:
020:
021:
022:
023:
024:
025:
026:
```

Load the JQuery files

```
027:
028:
029:
030:
031:
032:
033:
034:
035: <script type="text/javascript">
036: var image1 = new Image()
037: image1.src = "Pictures/homepageSlider/slide1.jpg"
038: var image2 = new Image()
039: image2.src = "Pictures/homepageSlider/slide2.jpg"
040: var image3 = new Image()
041: image3.src = "Pictures/homepageSlider/slide3.jpg"
042: var image4 = new Image()
043: image4.src = "Pictures/homepageSlider/slide4.jpg"
044: </script>
045:
046:
047:
048:
049:
050:
051:
052:
053:
054:
055:
056:
057:
058:
059:     <ul>
060:      <li class="top"><a href="aboutme.html" target="_blank">Resume</a></li>
061:      <li><a href="special.html" target="_blank">Projects</a></li>
062:     </ul>
063:
064:
065:
066:
067:
068:
069:
070:
```

The four images displayed on the homepage

target="_blank": opens the linked document in a new window or tab.

```
071:
072:
073:
074:
075:
076:  <script type="text/javascript">
077:  var step=1;
078:  function slideit()
079:  {
080:   document.images.slideNEW.src = eval("image"+step+".src");
081:   if(step<4)
082:    step++;
083:   else
084:    step=1;
085:   setTimeout("slideit()", 5000);
086:  }
087:  slideit();
088:  </script>
089:
090:
091:
092:
093:
094:
095:
096:
097:
098:
099:
100:
101: <div align="center">
102: You are the
103: <img src='http://www.hit-counter-html-code.com/c.php?d=9&id=140187&s=16' border='0'
height="16" width="90" title='counter'> visitors to this site.
104: </div>
105:
106:
```

Load a new image

Every 5000ms (5s), the function "setTimeout" will run and load a new image

Through this link the number of hits is obtained from the hit-counter site

aboutme.html

```
001:
002:
003:
004:
005:
006:
007: <script>
008: function overButton(img) {
009:  buttonImg = "buttons/button_"+img+"_over.png"
010:  document.getElementById(img).src = buttonImg
011: }
012: function upButton(img) {
013:  buttonImg = "buttons/button_"+img+"_up.png"
014:  document.getElementById(img).src = buttonImg
015: }
016: </script>
017:
018:
019:
020:
021:
:
037:
038:
039:
040:
041:
042:
043:
044:
045:
046: <div class="maincontent" id="list2">
047:  <h2 class="contentTitleh2">My Profile</h2>
048:  <div class="inner_Content">
049:   <div class="photo" style = "margin-left:auto; margin-right:0;" ><img
src="profile/profilephoto.jpg" alt="You should see my Photo Here!" width="350"
height="700.11"></div>
050:
:
108:
109:
```

overButton(img) function is used to display a new image, whose name is "buttonImg", at the same position of "img", when the mouse hovers on "img".

These classes are defined in mystyle.css

album.html

```
001:
  :
014:
015:
016:
017:
018: <style>
019: .rw-wrapper{ display: none; }
020: .rw-sentence-IE{ display: block;  }
021: </style>
022:
023:
024:
025:
026:
027:
028:
029:
  :
069:
070:
071:
072:
073:
074:
075:     <ul id="portfolio-filter">
076:      <li><a href="#" class="current" data-filter="*">All</a></li>
077:      <li><a href="#" data-filter=".California">California</a></li>
078:      <li><a href="#" data-filter=".GrandCanyon">Grand Canyon</a></li>
079:      <li><a href="#" data-filter=".Arches">Arches</a></li>
080:      <li><a href="#" data-filter=".NewYork">New York</a></li>
081:     </ul>
082:
083:     <li class="GrandCanyon">
084:      <div class="panel">
085:           <a href="album/Grand Canyon/GC_2.jpg" rel="portfolio" class="folio">
086:            <div class="front card"><img src="album/Grand Canyon/GC2.jpg" alt=""> </div>
087:            <div class="back card">
088:            <h4>Grand Canyon</h4>
089:            <p>Grand Canyon! Clear river in the canyon. </p>
090:            </div>
091:           </a>
```

The element will not be displayed at all (has no effect on layout).

Displays an element as a block element (like <p>).

Filter elements belong to class "California"

Front side

Back side

31

```
092:        </div>
093:        </li>
094:
095:
096:
:
189:
190:
191:
192:
193:
194:
```

contact.html

```
01:
:
12:
13:
14:
15: function sendMail() {
16:  var link = "mailto:chenya@iastate.edu"
17:        + "?subject=From " + escape(document.getElementById('name').value)+": "+escape(document.getElementById('subject').value)
18:        + "&body=" + escape(document.getElementById('comment').value);
19:  window.location.href = link;
20: }
21: function reset() {
22:  document.getElementById('name').value="";
23:  document.getElementById('subject').value="";
24:  document.getElementById('comment').value="";
25: }
26:
27:
:
64:
65:
66:
67:
68:  <input type="submit" name="send" value="Send" onclick="sendMail()"/>
69:  <input type="submit" name="reset" value="Reset" onclick="reset()"/>
70:
71:
```

This function is used to send an email

This function is used to reset some elements on the page to be empty

The click event will call the function.

72:
73:
74:
75:
76:
77:

family.html

001:
002:
003:
004:
005:
006:
 :
071:
072:
073:
074:    `<audio style="margin-left: 1px;"controls="controls" ><source src="video/moonlight.mp3" type="audio/mp3" />hihi</audio>`

> Put an audio here. The audio is in the folder "video" and with the name "moonlight.mp3".

075:
076:
077:
078:
079:
080:
:
084:
085:    `<video width = "320" height = "200" controls="controls" ><source src="video/video1.mp4" type="video/mp4"/><video>`

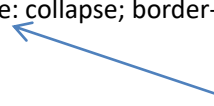> Put a video here. The audio is in the folder "video" and with the name "video1.mp4".

086:
087:
088:
089:
:
106:

special.html

001:
:
020:
021:
022:
023:
024:
025: \<div id="nav"\>
026: \<ul\>
027: \<li class="borderleft"\>\<a href="homepage.html" target="_blank"\>Home\</a\>\</li\>
028:
029:
030:
031:
:
096:
097:
098:
099:
100:
101:

The \<ul\> tag defines an unordered (bulleted) list. Use the \<ul\> tag together with the \<li\> tag to create unordered lists.

homepage.css

001:
:
006:
007:
008:
009:
010:
011: table {border-collapse: collapse; border-spacing: 0;}
012:
013:
014:
015:
016:
:
110:
111:
112:
113:

The border-collapse property sets whether the table borders are collapsed into a single border or detached.

114:

115:

116:

117: .toggle li:first-child a {-webkit-border-radius: 4px 0 0 4px; -moz-border-radius: 4px 0 0 4px; -o-border-radius: 4px 0 0 4px; border-radius: 4px 0 0 4px;}

> This property adds rounded borders to elements.

118:

119: .toggle li a.active { background: #c8e0f3; color: #325874; cursor: default; box-shadow: inset 0 0 3px rgba(0,0,0,0.4); -moz-box-shadow: inset 0 0 3px rgba(0,0,0,0.4); -webkit-box-shadow: inset 0 0 3px rgba(0,0,0,0.4); }

> RGBA color values are an extension of RGB color values with an alpha channel - which specifies the opacity of the object.

120:

121:

122:

123:

124:

125:

:

256:

257:

mystyle.css

001:

:

045:

046:

047:

048:

049:

050:

051:

052:

053: #nav ul {

054: list-style-type: none;

> The list-style-type specifies the type of list-item marker in a list.

055: }

056:

057: #nav ul li {

058: float: left;

059: position: relative;

> The element is positioned relative to its normal position, so "left:20" adds 20 pixels to the element's LEFT position.

060: }

061:

:

221:

```css
222:
223:
224:
225:
226: .picture{
227:     margin-left: 15%;
228:     border-spacing: 10px;
229: }
230:
231: caption{
232: caption-side: bottom;
233: font-weight: bold;
234: }
235:
236:
237:
238:
:
303:
304:
```

> The border-spacing property sets the distance between the borders of adjacent cells (only for the "separated borders" model).

> Specify the placement of a table caption.

slider.css

```css
001:
002:
003:
004:
005:
006:
007:
008:
009:
010: .flex-caption {
011: position: absolute;
012: top:0;
013: padding:30px;
014: padding-left: 80px;
015: height:35px;
016: color: white;
017: width:100%;
018: font-weight: bold;
```

> The element is positioned relative to its first positioned (not static) ancestor element.

```
019: opacity: 0;
020: font-family: "Arial Black", Gadget, sans-serif;
021: font-size: 50px;
022: text-transform: uppercase;
023: -webkit-transition: opacity 0.1s linear;
024: -webkit-transition-delay:0.5s;
025: }
026:
027:
028:
029:
030:
031:
032:
033:
034:
035:
036:
037:
   :
047:
048:
049:
050:
051:
052:
053: .flexslider {margin: 0; padding: 0;}
054: .flexslider .slides > li {display: none; -webkit-backface-visibility: hidden;}
055:
   :
119:
120:
121:
```

Wait 0.5 seconds before the transition effect starts.

Hide the backside of a rotated element.