Itec/MODR 2635 EN
Fall 2020

Homework 1: Submission LINK

What to submit: a PDF file with answer to questions, with LINKS to your results (web pages).

Moodle: https://eclass.yorku.ca

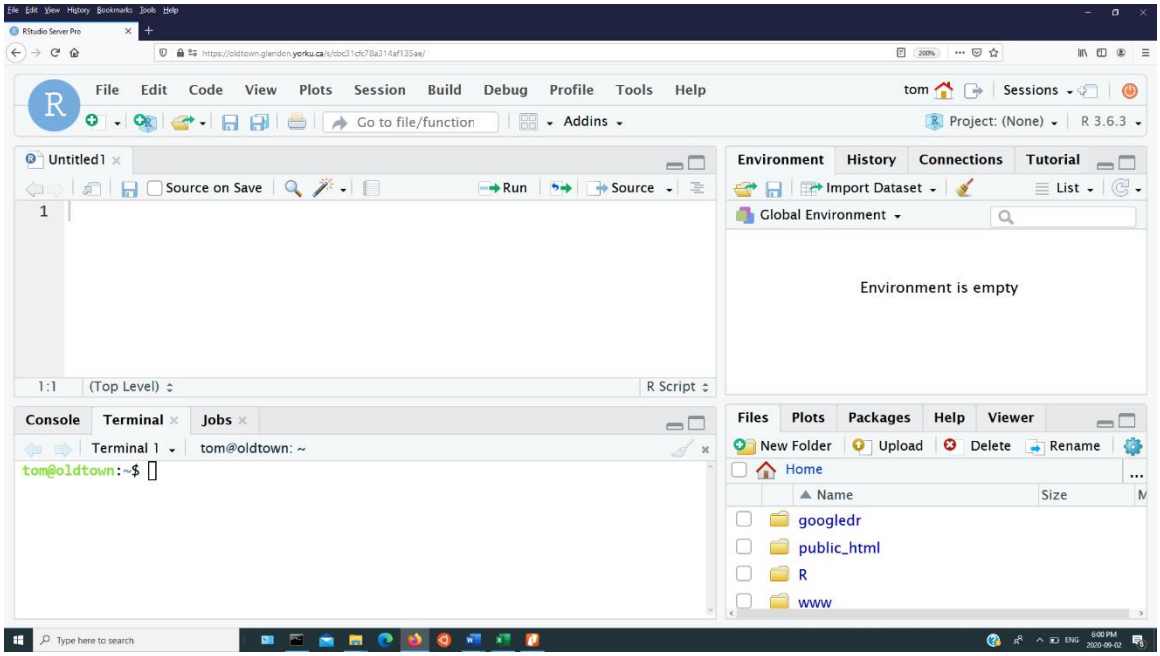
Rstudio-server: https://oldtown.glendon.yorku.ca/
Username: yourlastname (always in small letters)
Password: your_student_number
Your web pages are at: **http://oldtown.glendon.yorku.ca/~yourlastname**

Login our server at: https://oldtown.glendon.yorku.ca/
Username: yourlastname (always in small letters)
Password: your_student_number

- For example, if your full name is "Peter Toft", then you web site is http://oldtown.glendon.yorku.ca/~toft/
- If your last name is "Peter El Hare", then your web site is http://oldtown.glendon.yorku.ca/~elhare/ (last name is El Hare)
- There are 2 students D Chen and J Chen. So the web is for each student: http://oldtown.glendon.yorku.ca/~dchen/ http://oldtown.glendon.yorku.ca/~jchen/
- **Your web pages are in the directory www:** /home/yourlastname/www
- **Your home directory is /home/yourlastname**
- **~** is the abbreviation for **/home/lastname**

You will have next this graphic, or graphical user interface (GUI) below:

-



Click on "terminal" to activate that console in order to type command line (CLI), followed by "return" key. So we are giving a command to the computer (operating system) by typing a text (a keyword); more on Human–computer interaction.

Utility	Description
`ls`	"list" files and directories
`pwd`	"print working directory"
`cd`	"change (your working) directory"
`mkdir`	"make directory"
`rmdir`	"remove directory"
`cp`	"copy" a file or directory
`mv`	"move" a file or directory (i.e., rename it)
`rm`	"remove" a file (i.e., delete it)

Displaying Text Files

It is often convenient to look at the contents of a text file without having to open it in an editor. Previously in this lab, we saw that `cat` can be used for this purpose, but it is most useful for short files that can be viewed all on one screen.

GNU/Linux provides several other utilities that are useful for "paging" through text files (i.e., for viewing files one page at a time). Several of these commands are outlined in the following table.

Command	Description
`more`	move through a file screen by screen (hit space for the next page, return for one more line)

Exercise 1: 10 points

Definition (Wiki): A **command-line interface** (**CLI**) processes commands to a computer program in the form of lines of text. (Issue a command by typing text at a **terminal,** then press return.)

Try the following CLI:

(a) cd www
(b) mkdir 2021
(c) pwd
(d) cd 2021
(e) pwd
(f) mkdir images
(g) cd images
(h) convert -size 800x800 canvas:white white.png
(i) convert -size 800x800 canvas:blue blue.png
(j) Visit: http://oldtown.glendon.yorku.ca/~lastname/2021/images
(k) Download from http://oldtown.glendon.yorku.ca/~teaching/2915/l3/codes001.zip and display the content of this ZIP file at http://oldtown.glendon.yorku.ca/~lastname/2915/l3/

Exercise 2: 10 points

(a) In a terminal console, put yourself in Directory
/home/lastname/www/2021/images
using CLI (command line interface)

(b) In the GUI (graphical user interface) copy the song petit_navire.mp3 from
http://oldtown.glendon.yorku.ca/~teaching/music/mp3/petit_navire.mp3
to your directory:
/home/lastname/www/2021/images (i.e. Your PATH you made in (a)).

(c) Execute the CLI:

```
ffmpeg  -i blue.png -i petit_navire.mp3 -c:v libx264 -pix_fmt yuv420p
mymusic.mp4
```

Exercise 3: 10 points

#	Operation	What is the Command?
1	list files & directories.	
2	Put yourself at the HOME directory	
3	make a sub-directory called "**myfile**" in the home directory.	
4	Put yourself at the PATH /home/lastname/myfile	
5	Show your current PATH	
6	show the path for where you are in the directory.	
7	Make a subdirectory at /home/lastname/myfile/mp3	
8	list the contents of the home directory, no matter where you are.	
9	Put yourself at ~/myfile/mp3	
10	Go to ~ (the **HOME** directory which is /home/lastname)	
11		
12		
13		
14		
15		
16		
17		
18		
19		

20		

HTML Introduction

What is HTML?

HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of Web pages using markup
- HTML elements are the building blocks of HTML pages
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page

---

A Simple HTML Document

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

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

</body>
</html>
```

Example Explained

- The `<!DOCTYPE html>` declaration defines this document to be HTML5
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the document
- The `<title>` element specifies a title for the document
- The `<body>` element contains the visible page content
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

HTML Tags

HTML tags are element names surrounded by angle brackets:

<tagname>content goes here...</tagname>

- HTML tags normally come **in pairs** like and
- The first tag in a pair is the **start tag,** the second tag is the **end tag**
- The end tag is written like the start tag, but with a **forward slash** inserted before the tag name

Tip: The start tag is also called the **opening tag**, and the end tag the **closing tag**.

Make a web page with the html5 code given above on page 5, to be displayed at:

http://oldtown.glendon.yorku.ca/~yourlastname/examples/ex1

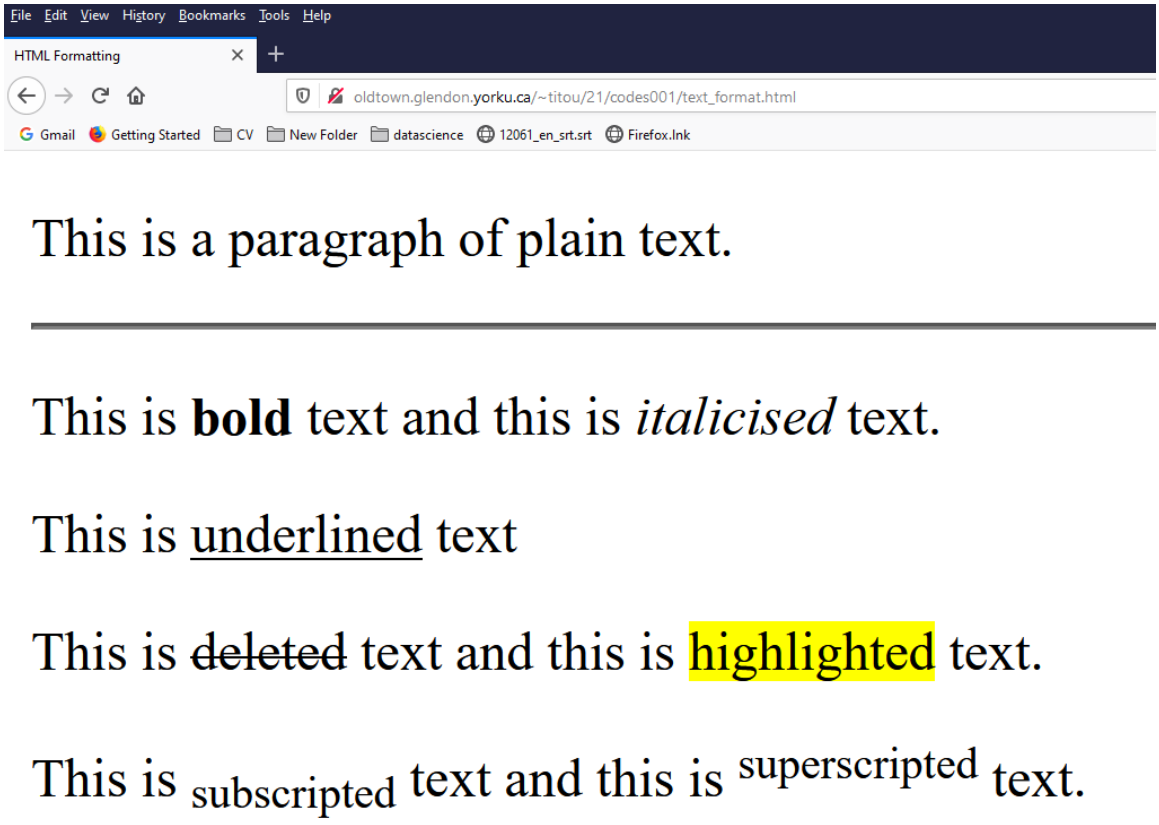
Exercise 4: 10 points

Complete the html code given below:

```
<!DOCTYPE html>
<html>
<head>
 <title>HTML Formatting</title>
 <meta charset="utf-8"/>
</head>

<body>
<p>This is a paragraph of plain text.</p>
<hr>
<p>This is strong text and this is italicised< text.</p>
<p>This is underlined text</p>
<p>This is deleted text and this is highlighted text.</p>
<p>This is subscriptedtext and this is superscripted text.</p>
<!-- this is a comment -->
</body>
</html>
```

So as to give this result below at your URL:

http://oldtown.glendon.yorku.ca/~lastname/2635/text/format/



Exercise 5: html5 10 points

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Ma passion pour les animaux</title>
  </head>

  <body>
    <h1>Titre de niveau 1</h1>
    <p>Voici mes animaux:</p>
    <ol>
      <li><strong>Dragon</strong></li>
      <li>chat</li>
      <li>giraffe</li>
    </ol>
  </body>
</html>
```

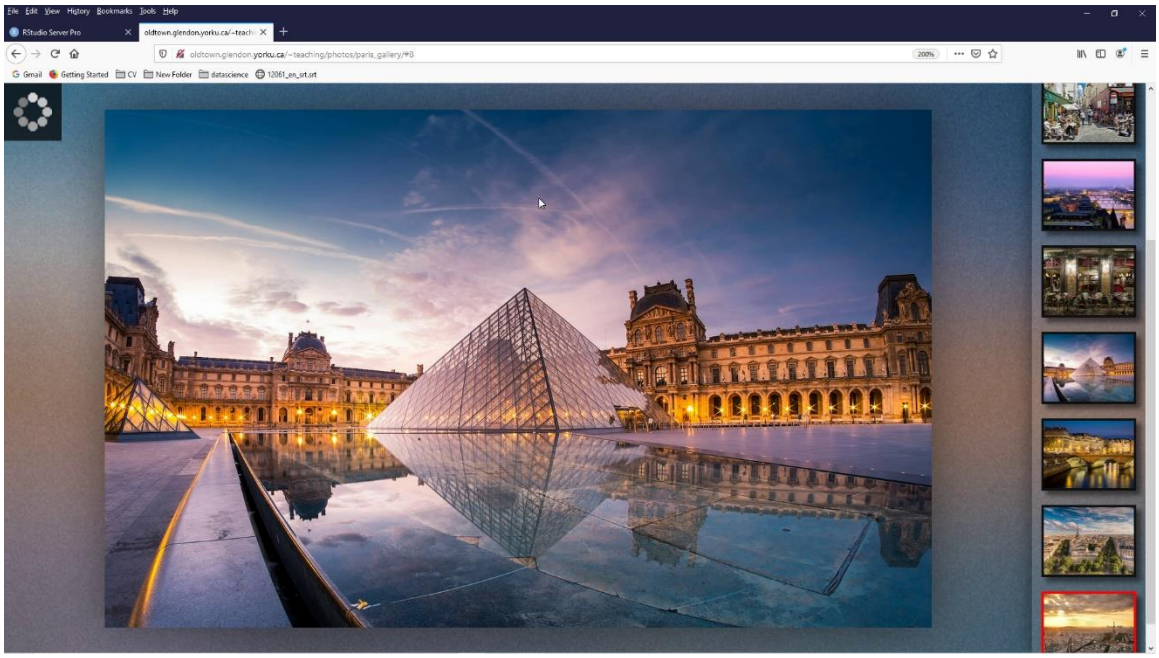
Make a web page with the html5 code given above to be displayed at:

http://oldtown.glendon.yorku.ca/~yourlastname/examples/ex212

Exercise 6: 10 points .Making a photo gallery with CLI.

(a) View, repeat and explain each of the CLI given in this video:

http://oldtown.glendon.yorku.ca/~teaching/2021/l1/ex004.mp4



(b)
The program is fgallery. The syntax (how to use it) is:

fgallery input_directory output_directory

I gave you the images paris.zip in your directory /home/lastname/www/examples/. Make a photo gallery with these images at your URL:

http://oldtown.glendon.yorku.ca/~lastname/vacances/paris

List all your steps.

Exercise 7: 10 points.

Your web page is at: **http://oldtown.glendon.yorku.ca/~yourlastname**

Visit the following links:

- http://oldtown.glendon.yorku.ca/~yourlastname/music/mp3

  - What should you do so that the URL link above be renamed as: http://oldtown.glendon.yorku.ca/~yourlastname/music/french2

  - What should you do so that that your Paris album is at this location: http://oldtown.glendon.yorku.ca/~yourlastname/paris/memory/french2

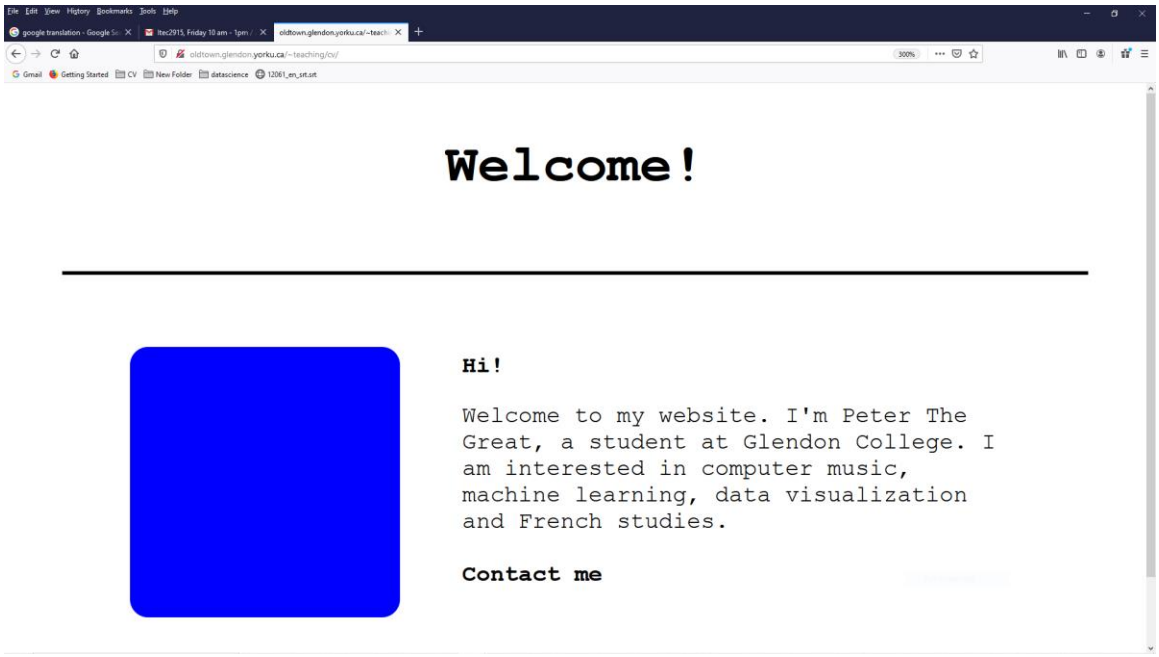
- http://oldtown.glendon.yorku.ca/~yourlastname/cv/index.html

  - How would you replace the ”blue box” with your own photo?
  - How would you replace “Welcome” by “Bienvenue!”
  - See: https://youtu.be/ZRfsCQPGiS4
  - or http://oldtown.glendon.yorku.ca/~teaching/2021/syllabus/oldtown_rstudio_server_title_cv.mp4

Hint: Edit the file index.html at /home/lastname/www/cv
Replace myphoto.png by your own image with the same name

Here is the html and css codes for the web page above:

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="css/cv.css">
<!-- Make for my students in Itec at Glendon College -->
</head>
<body>

<div id="header">
<h1> Welcome! </h1>
</div>

<h2>  </h2>
<div id="section">

    <div id="left">
    <center>
    <img src="myphoto.png" width="150px", height = "150px">
    </center>
    </div>

    <div id = "right">
    <b>Hi!</b><p>

        Welcome to my website. I'm Peter The Great, a student at
        Glendon College. I am interested in computer music, machine
        learning, data visualization and French studies.
    <br><br>
    <a href="mailto:someone@yoursite.com">Contact me</a>
    <br>

    </div>

</div>


</p>

<p>

</body>
</html>
```



CSS Basics

A basic CSS rule has the following format:

```
selector {
  property: value;
}
```

Selector
: The selector is the element that the rule will affect

Property
: The property is the actual CSS rule

Value
: The value is the value we want for the given property

Declarations and Declaration Blocks

Each property and value set are called a declaration.

```
selector {
  property: value;
}
```

You can list as many declarations as you want. A group of declarations are called a declaration block.

```
selector {
```

```
  property: value;
  property: value;
}
```

Things to note:

- Declaration blocks are always surrounded by curly braces
- Properties and values are always separated by a colon
- declarations always end with semicolons
- Spacing does not matter

CSS Comments

You type comments in to your CSS by using (`/*`) and (`*/`). Comments can span multiple lines.

```
/* A CSS comment */

/* A
Comment in
multiple lines
*/
```

Element Type Selectors

The most basic CSS selectors are **Element Type Selectors**. These are really just HTML tags.

For example, if we wanted to make all paragraphs have red text, we would use the following CSS rule:

```
p {
  color: red;
}
```

Multiple Selectors

If we wanted to make all of the ordered lists and unordered lists red, in addition to the paragraphs, we could use three rules like this:

```
p {
  color: red;
}
ol {
  color: red;
}
ul {
  color: red;
```

```
}
```

Better yet, we can **combine** these three rules, by listing the selectors, separated by commas.

```
p, ol, ul {
  color: red;
}
```

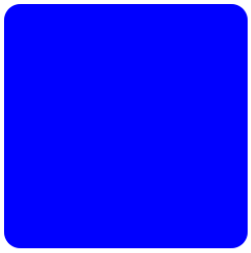
Exercise 8: 10 points

Given the code:

```
<!DOCTYPE html>
<html>
<head>
<!-- Make for my students in Itec at Glendon College -->
</head>
<body>

<div id="header">
<h1> Welcome......! </h1>
</div>

<h2>  </h2>
<div id="section">

   <div id="left">
   <center>
   <img src="myphoto.png" width="150px", height = "150px">
   </center>
   </div>

   <div id = "right">
   <b>Hi!</b><p>

      Welcome to my website. I'm Peter Toft, a student at Glendon College. I a
m interested in computer music,  machine learning,
      data visualization and French studies.
   <br><br>
   <a href="mailto:tu@gmail.com">Contact me</a>
   <br>

   </div>

</div>


</p>

<p>

</body>
</html>
```

Write the style.css so that your result will look like this:

Welcome.....!



Hi!

Welcome to my website. I'm Peter Toft, a student at Glendon College. I am interested in computer music, machine learning, data visualization and French studies.

Contact me

Exercise 9: 20 points .Web security

(a) Make the URL: http://oldtown.glendon.yorku.ca/~lastname/practice202/

(b) Protect the URL above with username "tintin" and password "glendon2020".

Here is your .htaccess file:

AuthType Basic
AuthName "restricted area"
AuthUserFile /home/teaching/secret/.htpasswd
Require valid-user

Solution:

1. mkdir /home/lastname/www/practice2
2. Create .htaccess at PATH above has the following 4 lines as content:

   AuthType Basic
   AuthName "restricted area"
   AuthUserFile /home/lastname/secret/.htpasswd
   Require valid-user

3. Create directory /home/lastname/secret/
   (use the command mkdir)
4. Create username and password:
   htpasswd -c /home/lastname/secret/htpasswd username

   - done

Links:

https://www.w3schools.com/

Notes & Misc. Commands:

```
ffmpeg -loop 1 -i image.png -c:v libx264 -pix_fmt yuv420p -vf
scale=320:240 out.mp4


ffmpeg -r 5 -i img%03d.jpg -vcodec h264 -pix_fmt yuv420p -crf 22 -s
1920x1080 MOVIE.mp4

Recording using your sound card (stereo mix) and Audacity.
```