

# UFR 27 — L2 MIASHS Renforcement Informatique

## Développement WEB

Durée de l'épreuve: 1h30

Responsable: Nicolas Herbaut

16 Janvier 2019

N° carte d'étudiant : \_\_\_\_\_

**Aucun document n'est autorisé. Un aide mémoire a été ajouté à la fin du sujet.  
Vous devez répondre aux parties 1 et 2 directement sur le sujet, pensez donc à remettre ce sujet à l'intérieur de votre copie d'examen !**

**Les téléphones et autres mobiles/tablettes doivent être éteints dans votre sac. Le sac doit être posé au sol devant le tableau**

### 1 Bases des languages (3 pts)

1. (1 point) Que signifie CSS?

**Solution:** Cascading Stylesheets

2. (1 point) Quel est le rôle du HTML dans une page WEB?

**Solution:** Le HTML permet de structurer les données affichées dans une page web. Il est responsable de l'expression du contenu.

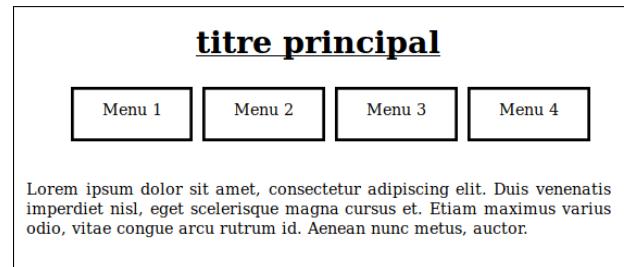
3. (1 point) Quelle balise est utilisée pour créer des liens dans une page web? Donnez un exemple d'un lien vers le site de votre Université.

**Solution:**

```
<a href="https://pantheonsorbonne.fr">Mon université</a>
```

### 2 Mise en page simple (5pt)

On souhaite obtenir le rendu suivant pour la page principal d'un site.



- (2.5 points) Donnez le HTML correspondant à ce qui est affiché. Pour cela, vous utiliserez des listes non ordonnées pour les menus.

**Solution:**

```
<div id="title"><h1>titre principal</h1>
<ul>
    <li>Menu 1</li>
    <li>Menu 2</li>
    <li>Menu 3</li>
    <li>Menu 4</li>
</ul>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis venenatis imperdierit nisl,
    eget scelerisque magna cursus et. Etiam maximus varius odio, vitae congue arcu rutrum
    id. Aenean nunc metus, auctor. </p>
</div>
```

- (2.5 points) Donnez le CSS correspondant à ce qui est affiché. Pour cela, nous vous donnons les spécifications supplémentaires suivantes:

- tout le contenu doit être centré horizontalement sur la page, avec une largeur de 600px et une hauteur de 100px
- le titre principal doit être souligné et centré horizontalement
- le paragraphe doit être justifié
- chaque élément du menu doit mesurer 30x100 px
- avoir un margin de 5px et un padding de 10px
- chaque élément du menu doit avoir une bordure
- chaque élément du menu doit être flottant et centré
- le menu en lui même doit être séparé du paragraphe de 30px et sa hauteur totale doit être de 70px

**Solution:**

```
div{
    width:600px;
    margin:auto;
    height:100px;
}
h1{
    text-decoration:underline;
    text-align:center;
}

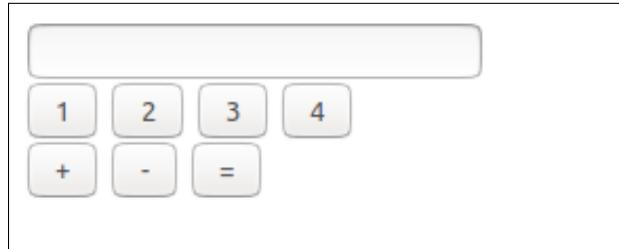
p{
    text-align:justify;
}
li{
    height:30px;
    float:left;
    list-style:none;
    width:100px;
    text-align:center;
    border-style:solid;
    padding:10px;
    margin:5px;
}
ul{
```

**Solution:**

**Solution:**

### 3 Mini-calculatrice JavaScript

On souhaite réaliser une mini-calculatrice en JS qui est exécutée par le navigateur.



Voici un exemple de séquence d'appel:

1. clic sur 1
  2. l'élément total affiche 1
  3. clic sur 2
  4. l'élément total affiche 12
  5. clic sur +
  6. l'élément total est vidé
  7. click sur 3
  8. l'élément total affiche 3
  9. click sur "="
  10. l'élément total affiche 15.
1. (4 points) Donnez le HTML permettant d'afficher cette mini-calculatrice. Vous prendrez bien soin de donner des classes aux éléments remplissant la même fonction (nombres, opérateurs...) et des id aux éléments remplissant une fonction unique (égal, total...)

#### Solution:

```
<input id="total">
<br>
<input class="number" id="btn1" type="button" value="1"/>
<input class="number" id="btn2" type="button" value="2"/>
<input class="number" id="btn3" type="button" value="3"/>
<input class="number" id="btn4" type="button" value="4"/>
<br>
<input class="operator" type="button" value="+"/> />
<input class="operator" type="button" value="-"/> />
<input id="equals" type="button" value="="/> />
```

2. (4 points) Donnez le code JS permettant d'enregistrer les évènements sur les boutons. On rappelle que la fonction que vous devez créer pour recevoir le déclenchement d'un évènement (ici "event") est de la forme:

```
//elt=l'élément sur lequel enregistrer l'évènement
elt.addEventListener("event",callback);

//la fonction qui va traiter l'évènement
function callback(evt){
  var value = evt.target.value;
  //ici la variable value permet d'accéder à la valeur de l'élément cliqué,
  //passé en paramètre de la fonction
}
```

**Solution:**

```
for(let btn of document.getElementsByClassName("number")){
  btn.addEventListener("click",handleButton);
};
for(let op of document.getElementsByClassName("operator")){
  op.addEventListener("click",handleOperator);
}
document.getElementById("equals").addEventListener("click",handleEquals);
```

3. (4 points) Donnez le code permettant de faire fonctionner la calculatrice.

**Solution:**

```
var operand1=0;
var operator="";
for(let btn of document.getElementsByClassName("number")){
  btn.addEventListener("click",handleNumber);
};

for(let op of document.getElementsByClassName("operator")){
  op.addEventListener("click",handleOperator);
};
document.getElementById("equals").addEventListener("click",handleEquals);

function handleNumber(evt){
  document.getElementById("total").value+=evt.target.value;
}
function handleOperator(evt){
  operator=evt.target.value;
  operand1=parseInt(document.getElementById("total").value);
  document.getElementById("total").value="";
};

function handleEquals(evt){
  computeOperation();
};

function computeOperation(){
  var value=0;
  if(operator=="+"){
    value=operand1+parseInt(document.getElementById("total").value);
  }
  else{
    value=operand1-parseInt(document.getElementById("total").value);
  }
  document.getElementById("total").value=value;
}
```

Regular Expressions Syntax	
^	Start of string
\$	End of string
.	Any single character
(a b)	a or b
(...)	Group section
[abc]	In range (a, b or c)
[!abc]	Not in range
\s	White space
a?	Zero or one of a
a*	Zero or more of a
a <sup>*</sup>	Zero or more, ungreedy
a <sup>+</sup>	One or more of a
a <sup>+</sup> ?	One or more, ungreedy
a{3}	Exactly 3 of a
a{3,}	3 or more of a
a{6}	Up to 6 of a
a{3,6}	3 to 6 of a
a{3,6}?	3 to 6 of a, ungreedy
\	Escape character
[:punct:]	Any punctuation symbol
[:space:]	Any space character
[:blank:]	Space or tab
There's an excellent regular expression tester at: <a href="http://regexpal.com/">http://regexpal.com/</a>	
Pattern Modifiers	
g	Global match
i *	Case-insensitive
m *	Multiple lines
s *	Treat string as single line
JavaScript Booleans	
toSource()	valueOf()
toString()	
JavaScript Dates	
Date()	setMonth()
getDate()	setFullYear()
getDay()	setHours()
getMonth	setMinutes()
getFullYear	setSeconds()
getYear	setMilliseconds()
getHours	setTime()
getMinutes	setUTCDate()
getSeconds	setUTCDay()
getMilliseconds	setUTCMonth()
getTime	setUTCFullYear()
getTimezoneOffset()	setUTCHours()
getUTCDate()	setUTCMinutes()
getUTCDay()	setUTCSeconds()
getUTCMonth()	setUTCMilliseconds()
getUTCFullYear()	toSource()
getUTCHours()	toString()
getUTCMinutes()	toGMTString()
getUTCSeconds()	toUTCString()
getUTCMilliseconds()	toLocaleString()
parse()	UTC()
setDate()	valueOf()
Pattern Modifiers (cont)	
x *	Allow comments and whitespace in pattern
e *	Evaluate replacement
U *	Ungreedy pattern
* PCRE modifier	
JavaScript RegExp Object	
compile()	lastParen
exec()	leftCOnText
global	multiline
ignoreCase	rightContext
input	source
lastIndex	test()
lastMatch	
JavaScript Event Handlers	
onabort	onmousedown
onblur	onmousemove
onchange	onmouseout
onclick	onmouseover
ondblclick	onmouseup
ondragdrop	onmove
onerror	onreset
onfocus	onresize
onkeydown	onselect
onkeypress	onsubmit
onkeyup	onunload
onload	
JavaScript Strings	
charAt()	slice()
charCodeAt()	split() x
concat()	substr()
fromCharCode()	substring()
indexOf()	toLowerCase()
lastIndexOf()	toUpperCase()
length	toLocaleLowerCase()
localeCompare()	toLocaleUpperCase()
match() x	toSource()
replace() x	valueOf()
search() x	
String object methods with an x support regular expressions.	
JavaScript Arrays	
concat()	slice()
join()	sort()
length	splice()
pop()	toSource()
push()	toString()
reverse()	unshift()
shift()	valueOf()
JavaScript Numbers and Maths	
abs()	min()
acos()	NEGATIVE_INFINITY
asin()	PI
atan()	POSITIVE_INFINITY
atan2()	pow()
ceil()	random()
cos()	round()
E	sin()
exp()	sqrt()
floor()	SQRT1_2
LN10	SQRT2
LN2	tan()
log()	toSource()
LOG10E	toExponential()
LOG2E	toFixed()
max()	toPrecision()
MAX_VALUE	toString()
MIN_VALUE	valueOf()
NaN	
JavaScript Functions	
decodeURI()	isNaN()
decodeURIComponent()	Number()
encodeURI()	parseFloat()
encodeURIComponent()	parseInt()
escape()	String()
eval()	unescape()
isFinite()	

## CSS CHEAT SHEET

### Shorthand\*

background  
border  
border-bottom  
border-left  
border-right  
border-top  
font  
list-style  
margin  
padding

### Comments

/\* Comment \*/

### Pseudo Selectors

:hover  
:active  
:focus  
:link  
:visited  
:first-line  
:first-letter

### Media Types

all  
braille  
embossed  
handheld  
print  
projection  
screen  
speech  
tty  
tv

### Units

Length %  
em  
pt  
px

Keywords  
bolder  
lighter  
larger

### SYNTAX

#### Syntax

selector {property: value;}

#### External Style Sheet

<link rel="stylesheet" type="text/css"  
href="style.css" />

#### Internal Style

<style type="text/css">  
selector {property: value;}  
</style>

#### Inline Style

<tag style="property: value">

### GENERAL

Class	String preceded by a period
ID	String preceded by a hash mark
div	Formats structure or block of text
span	Inline formatting
color	Foreground color
cursor	Appearance of the cursor
display	block, inline, list-item, none
overflow	How content overflowing its box is handled
	visible, hidden, scroll, auto
visibility	visible, hidden

### FONT

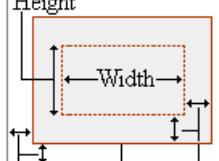
font-style	italic, normal
font-variant	normal, small-caps
font-weight	bold, normal, lighter, bolder, integer (100-900)
font-size	Size of the font
font-family	Specific font(s) to be used

### TEXT

letter-spacing	Space between letters
line-height	Vertical distance between baselines
text-align	Horizontal alignment
text-decoration	blink, line-through, none, overline, underline
text-indent	First line indentation
text-transform	capitalize, lowercase, uppercase
vertical-align	Vertical alignment
word-spacing	Spacing between words

### BOX MODEL

#### Height



#### Margin Border Padding

### BORDER

border-width	Width of the border
border-style	dashed, dotted, double, groove, inset, outset, ridge, solid, none
border-color	Color of the border

### POSITION

clear	Any floating elements around the element?
	both, left, right, none
float	Floats to a specified side
	left, right, none
left	The left position of an element
	auto, length values (pt, in, cm, px)
top	The top position of an element
	auto, length values (pt, in, cm, px)
position	static, relative, absolute
z-index	Element above or below overlapping elements?
	auto, integer (higher numbers on top)

### BACKGROUND

background-color	Background color
background-image	Background image
background-repeat	repeat, no-repeat, repeat-x, repeat-y
background-attachment	Background image scroll with the element?
	scroll, fixed
background-position	(x y), top, center, bottom, left, right

### LIST

list-style-type	Type of bullet or numbering in the list
	disc, circle, square, decimal, lower-roman, upper-roman, lower-alpha, upper-alpha, none
list-style-position	Position of the bullet or number in a list
	inside, outside
list-style-image	Image to be used as the bullet in a list



HTML 5 NEW TAG TAG NOT SUPPORTED IN HTML 5	
<!--...-->	Define a comment
<!DOCTYPE>	Defines the document type
<a>	Defines a hyperlink
<abbr>	Defines an abbreviation
<acronym>	Used to define an embedded acronyms
<address>	Defines an address element
<applet>	Used to define an embedded applet
<area>	Defines an area inside an image map alt, coords, href, hreflang, media, ping, rel, shape, target, type
<article>	Defines an article cite, pubdate
<aside>	Defines content aside from the page content
<audio>	Defines sound content autobuffer, autoplay, controls, src
<b>	Defines bold text
<base>	Defines a base URL for all the links in a page href, target
<basefont>	Used to define a default font-color, font-size, or font-family for all the document
<bdo>	Defines the direction of text display dir
<big>	Used to make text bigger
<blockquote>	Defines a long quotation cite
<body>	Defines the body element
 	Inserts a single line break
<button>	Defines a push button autofocus, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, name, type, value
<canvas>	Defines graphics height, width
<caption>	Defines a table caption
<center>	Used to center align text and content
<cite>	Defines a citation
<code>	Defines computer code text autobuffer, autoplay, controls, src
<col>	Defines attributes for table columns
<colgroup>	Defines groups of table columns span
<command>	Defines a command button checked, disabled, icon, label, radiogroup, type

<datalist>	Defines a dropdown list
<dd>	Defines a definition description
<del>	Defines deleted text cite, datetime
<details>	Defines details of an element open
<dialog>	Defines a dialog (conversation)
<dfn>	Defines a definition term
<dir>	Used to define a directory list
<div>	Defines a section in a document
<dl>	Defines a definition list
<dt>	Defines a definition term
<em>	Defines emphasized text
<embed>	Defines external interactive content or plugin height, src, type, width
<fieldset>	Defines a fieldset disabled, form, name
<figure>	Defines a group of media content, and their caption
<font>	Used to define font face, font size, and font color of text
<footer>	Defines a footer for a section or page
<form>	Defines a form accept-charset, action, autocomplete, enctype, method, name, novalidate, target
<frame>	Used to define one particular window (frame) within a frameset
<frameset>	Used to define a frameset, which organizes multiple windows (frames)
<h1> to <h6>	Defines header 1 to header 6
<head>	Defines information about the document
<header>	Defines a header for a section or page
<hgroup>	Defines information about a section in a document
<hr>	Defines a horizontal rule
<html>	Defines an html document manifest, xmlns
<i>	Defines italic text
<iframe>	Defines an inline sub window height, name, sandbox, seamless, src, width
<img>	Defines an image alt, src, height, ismap, usemap, width
<input>	Defines an input field accept, alt, autocomplete, autofocus, checked, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, height, list, max, maxlength, min, multiple, name, pattern, placeholder, readonly, required, size, src, step, type, value, width
<ins>	Defines inserted text cite, datetime
<keygen>	Defines a generated key in a form autofocus, challenge, disabled, form, keytype, name
<kbd>	Defines keyboard text
<label>	Defines an inline sub window for, form
<legend>	Defines a title in a fieldset
<li>	Defines a list item value
<link>	Defines a resource reference href, hreflang, media, rel, sizes, type
<map>	Defines an image map name
<mark>	Defines marked text
<menu>	Defines a menu list label, type
<meta>	Defines meta information charset, content, http-equiv, name
<meter>	Defines measurement within a predefined range high, low, max, min, optimum, value
<nav>	Defines navigation links
<noframes>	Used to display text for browsers that do not handle frames
<noscript>	Defines a noscript section
<object>	Defines an embedded object data, form, height, name, type, usemap, width
<ol>	Defines an ordered list reversed, start
<optgroup>	Defines an option group label, disabled
<option>	Defines an option in a drop-down list disabled, label, selected, value
<output>	Defines some types of output for, form, name
<p>	Defines a paragraph
<param>	Defines a parameter for an object name, value
<pre>	Defines preformatted text
<progress>	Defines progress of a task of any kind max, value
<q>	Defines a short quotation cite
<rp>	Used in ruby annotations to define what to show browsers that do not support the ruby element
<rt>	Defines explanation to ruby annotations
<ruby>	Defines ruby annotations
<s>, <strike>	Used to define strikethrough text.

## HTML5 TAG CHEAT SHEET

Created by WebsiteSetup.org