

## Homework / Terminology & Definitions (CSS)

### Clémence TAN (group B)

1. What are the main differences between external, internal, and inline CSS? (pptx p38)

=> **external CSS** : the CSS is linked via a separated css file, with the css link declared inside the <head> element of the html file

=> **internal CSS** : the CSS is linked via the <style> element declared inside the <head> element of the html file

=> **inline CSS** : the CSS is declared directly in the html element line, like this :

```
<p style="color:lightgreen;">I am green.</p>
```

2. What is the syntax for class and ID selectors?

=> class : .

=> ID : #

3. How would you apply a single rule to two different selectors?

=> for example : (the same color applied to two different selectors)

```
div, p {  
  color: #f00;  
}
```

4. Given an element that has an id of title and a class of primary, how would you use both attributes for a single rule?

```
#title, .primary {  
  color: #f00;  
}
```

5. What does the descendant combinator do? (pptx p15)

=> A descendant combinator will only cause elements that match the last selector to be selected. The example below, only the .child will be affected

```
.grandparent .child {  
  color: pink;  
}
```

6. Between a rule that uses one **class** selector and a rule that uses three **type** selectors, which rule has the higher specificity?

=> **class selector**

7. From inside to outside, what is the order of box-model properties?

=> (content) → padding → border → margin

8. What does the box-sizing CSS property do?

=> It sets how the total width and height of an element is calculated.

9. What is the difference between the standard and alternative box model?

=> the standard box is the default, where a declaration of the width and height properties only defines the width and height of the content box.

=> the alternative box is settled by using :

```
* {  
  box-sizing: border-box;  
}
```

**10.** Would you use margin or padding to create more space between 2 elements? (pptx p43)

=> margin

**11.** Would you use margin or padding to create more space between the contents of an element and its border? (pptx p43)

=> padding

**12.** Would you use margin or padding if you wanted two elements to overlap each other?

=> padding

**13.** What is the difference between a block element and an inline element?

=> **block element** starts from a new line and cover the entire horizontal space

=> **inline element** covers only the area which is bounded by the tags in the HTML element and never start from the new line

**14.** What is the difference between an inline element and an inline-block element?

=> **inline element** covers only the area which is bounded by the tags in the HTML element and never start from the new line

=> **inline-block element** displays an element as an inline-level block container (ie. 3 blocks in a line)

**15.** Is an h1 block or inline?

=> block

**16.** Is button block or inline?

=> inline

**17.** Is div block or inline?

=> block

**18.** Is span block or inline?

=> inline

**19.** What's the difference between a flex container and a flex item?

=> a flex container contains flex items

=> but a flex item can also be a flex container itself

**20.** How do you create a flex item?

=> we will need to turn the parent of the flex item into a flex container, by typing :  
display : flex

**21.** What are the 3 values defined in the shorthand flex property?

=> flex-grow → flex-shrink → flex-basis

**22.** How do you make flex items arrange themselves vertically instead of horizontally?

=> by declaring its parent as a flex container + `flex-direction: column;`

**23.** What is the difference between justify-content and align-items?

=> justify-content handles alignment of all items on the main axis

=> align-item handles alignment of all items on the cross axis

**24.** How do you use flexbox to completely center a div inside a flex container?

=> justify-conter : center

**25.** What's the difference between justify-content: space-between and justify-content: space-around?

=> **justify-content: space-between** : this distributes items evenly, with the 1st item flushed with the start and the last item flushed ith the end

=> **justify-content: space-around** : this distributes items evenly, items have a hal-size space on either end