

Nome: _____

Nº aluno: _____

Notas Importantes:

Deve assinalar apenas uma opção como sendo a correta e/ou mais completa.

Nas questões de **escolha múltipla**, cada resposta errada implica uma penalização na respetiva cotação da pergunta (25%). Se o aluno não responder não há lugar a qualquer penalização. Deve identificar a resposta correta através da especificação da respetiva letra no espaço destinado para esse efeito.

As cotações foram definidas tendo por base 20 valores.

Folha de Respostas

Questão	Resposta	Questão	Resposta	Questão	Resposta
1		15		32	
2		16		33	
3		17		34	
4		18		35	
5		19		36	
6		20		37	
7		21		38	
8		24		39	
9		25			
10		26			
11		27			
12		29			
13		30			
14		31			

22.1)

--

22.2)

22.3)

22.4)

23.

28.

1. Os conceitos *Single Page Application (SPA)* e *Multi page Application (MPA)* referem-se à estrutura da aplicação:

- A. determinando diferentes interações com o servidor.
- B. não tendo qualquer influência na forma como é obtida informação do servidor.
- C. determinando diferentes interações com o servidor, sendo que no caso do MPA a atualização do conteúdo é controlada por um *script* executado no cliente.
- D. determinando diferentes interações com o servidor, sendo que no caso do SPA a atualização do conteúdo é controlada por um *script* SASS executado no cliente.

[0.4 Valores]

2. As *Single Page Application (SPA)* implicam:

- A. minimizar a interação com o utilizador.
- B. uma interação mais fluída com o utilizador, uma vez que existe um aumento no volume de dados a solicitar ao servidor.
- C. uma interação menos fluída com o utilizador, uma vez que existe uma redução no volume de dados a solicitar ao servidor.
- D. uma interação mais fluída com o utilizador, uma vez que existe uma redução no volume de dados a solicitar ao servidor.

[0.4 Valores]

3. Uma aplicação web baseia-se num modelo cliente servidor em que a comunicação é estabelecida através do protocolo:

- A. HTTP.
- B. HTML.
- C. LESS.
- D. XHTML.

[0.4 Valores]

4. O conceito de *Search Engine Optimization* também depende:

- A. do dispositivo utilizado para aceder aos conteúdos (*web mobile*).
- B. dos atributos definidos na *tag* `<meta>`.
- C. dos atributos definidos na *tag* `<script>`.
- D. dos atributos definidos na *tag* `<header>`.

[0.4 Valores]

5. O *Responsive Web Design* também está relacionado com a optimização do espaço do *viewport*, a qual pode ser implementada através da *tag*:

- A. `<meta>`
- B. `<picture>`.
- C. `<screen>`.
- D. `<viewport>`.

[0.4 Valores]

6. A ultima versão do HTML (HTML5):

- A. é totalmente incompatível com as versões anteriores.
- B. simplifica a sintaxe, não introduzindo, no entanto, qualquer novo elemento HTML.
- C. simplifica a sintaxe, não introduzindo, no entanto, qualquer novo atributo HTML.
- D. é compatível com as versões anteriores.

[0.4 Valores]

7. Se um elemento HTML não possui conteúdo:

- A. também não possui atributos.
- B. pode, no entanto, possuir atributos.
- C. designa-se por *empty HTML entity*.
- D. designa-se por *empty HTML block*.

[0.4 Valores]

8. Um elemento HTML pode possuir:

- A. vários atributos separados por “;” (ponto e vírgula).
- B. vários atributos separados por espaço, sendo que não pode ser atribuído mais de um valor a cada atributo.
- C. vários atributos separados por espaço.
- D. vários atributos separados por “,” (vírgula).

[0.4 Valores]

9. De acordo com a última versão do HTML a não especificação do `<head>`:

- A. impede a interpretação da *tag* `<title>`.
- B. impede a interpretação de todas as *tags* HTML.
- C. impede a interpretação apenas das *tags* vulgarmente contidas no `<head>`.
- D. permite a interpretação da *tag* `<title>`.

[0.4 Valores]

10. Os formulários constituem um importante elemento de interatividade com o utilizador, neste contexto a *tag* `<textarea>`:

- A. é uma *empty tag* e permite a introdução de várias linhas de texto.
- B. permite a introdução de várias linhas de texto.
- C. permite a introdução de apenas uma linha de texto.
- D. é um *empty element* e permite a introdução de várias linhas de texto.

[0.4 Valores]

11. O atributo *placeholder* pode ser aplicado:

- A. apenas à *tag* `<input>`.
- B. apenas à *tag* `<textarea>`.
- C. apenas à *tag* `<select>`.
- D. às *tags* `<input>` e `<textarea>`.

[0.4 Valores]

12. A *tag* <div> é:

- A. equivalente ao elemento .
- B. um *container* específico para elementos *inline*.
- C. um *container* genérico muito relevante no contexto do HTML.
- D. um *container* genérico pouco relevante no contexto do HTML.

[0.4 Valores]

13. A *tag* <meta> é especificada no:

- A. <body> e é considerada *deprecated*.
- B. <body> e não é considerada *deprecated*.
- C. <head> e não é considerada *deprecated*.
- D. <head> e é considerada *deprecated*.

[0.4 Valores]

14. Uma lista :

- A. só pode ter como sublista uma lista .
- B. só pode ter como sublista uma lista .
- C. pode ter como sublista uma lista ou .
- D. pode ter como sublista uma lista ou .

[0.4 Valores]

15. A ligação a um ficheiro externo (CSS) é implementada tendo por base a *tag* <link>, a qual:

- A. não suporta atributos.
- B. suporta apenas os atributos *href* e *rel* separados por espaço
- C. suporta vários atributos separados por espaço.
- D. suporta apenas os atributos *href*, *rel* e *src* separados por espaço.

[0.4 Valores]

16. No contexto da tecnologia CSS, os seletores de ***class*** utilizados para identificar campos de formulários, são formados tendo por base:

- A. # seguido pela designação do atributo *class*.
- B. \$ seguido pela designação do atributo *class*.
- C. % seguido pela designação do atributo *class*.
- D. • seguido pela designação do atributo *class*.

[0.4 Valores]

17. No contexto da tecnologia CSS, os ***pseudo-element selectors***, referem-se:

- A. a elementos fictícios criados com base na estrutura HTML (*::first-letter*, *::first-line*, *::selection*, ...).
- B. aos diversos elementos HTML (<p>, <div>, <a>, ...).
- C. a elementos fictícios criados com base na estrutura HTML (*::after*, *::first-line*, *:link*, ...).
- D. a elementos fictícios criados com base na estrutura HTML (*:link*, *:visited*, *:hover*, ...).

[0.4 Valores]

18. No contexto da tecnologia CSS na formatação de uma página HTML os conceitos de Herança e Especificidade:

- A. não podem ser considerados em simultâneo.
- B. são irrelevantes.
- C. devem ser considerados.
- D. devem ser considerados, sendo que a herança prevalece sempre sobre a especificidade.

[0.4 Valores]

19. Os *pseudo-class selectors* :

- A. aplicam-se exclusivamente aos *links* `<a>`.
- B. não se aplicam aos *empty elements*.
- C. aplicam-se exclusivamente aos *links* `<a>` e aos *containers* `<div>`.
- D. podem ser aplicados a diversos elementos estruturais.

[0.4 Valores]

20. No *fluid layout* as dimensões dos elementos estruturais são estabelecidas em:

- A. *px* ou *em*.
- B. %.
- C. *px* ou %.
- D. *px*.

[0.4 Valores]

21. O seletor [*xpto*] permite selecionar:

- A. os elementos `<div>` cujo atributo *xpto* se encontra definido.
- B. todos os elementos HTML cujo valor do atributo é *xpto*.
- C. todos os elementos HTML cujo atributo *xpto* se encontra definido.
- D. os elementos `<div>` cujo valor do atributo é *xpto*.

22. Considerando a estrutura HTML representada na figura 1, indique justificando quais os elementos que são afetados pelos seletores seguintes:

```
<body><div>
  <p> Tecnologias <span>Web</span></p>
  <ul>
    <li>Estrutura</li>
    <li>Apresentação</li>
    <li>Comportamento</li>
  </ul>
</div>
<div>
  <p>Conteúdos <span>Teórica:</span></p>
  <ul>
    <li>HTML </li>
    <li>CSS </li>
    <li>JavaScript</li>
  </ul>
</div></body>
```

Figura 1 – Estrutura HTML

Nota Importante: as alíneas devem ser consideradas de forma independente.

22.1 [0.6 Valores]

`li:nth-of-type(4){color:orange}`

22.2 [0.6 Valores]

`body>div ul li{color:orange}`

22.3 [0.6 Valores]

`body div ul:first-of-type li{color:orange}`

22.4 [0.6 Valores]

`body div ul:first-child li{color:orange}`

[1.6 Valores]

23. Considere os seguintes códigos HTML e CSS, e o respetivo output:

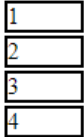
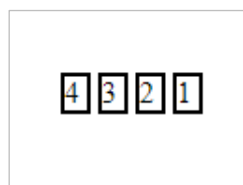
<pre> 1 2 3 4 </pre>	<pre>li{width:50px; border:2px solid black; list-style-type: none; margin:2px;} ul{width:80px}</pre>	
---	---	---

Figura 2 – Código HTML/CSS/Output

especifique qual o código CSS de forma a alterar o *output* tal como abaixo representado:



[0.4 Valores]

24. No contexto do CSS *flexbox*, as propriedades são aplicadas:

- A. apenas ao *flex container*.
- B. apenas aos *flex items*.
- C. ao *flex-grid* e aos *flex-items*.
- D. ao *flex container* e aos *flex-items*.

[0.4 Valores]

25. No contexto das *CSS Grid* a unidade **fr**:

- A. não se encontra definida, como tal não pode ser utilizada.
- B. é semelhante à unidade **em**.
- C. é particularmente interessante uma vez que permite o ajustamento da *grid* ao espaço disponível.
- D. não pode ser utilizada em conjunto com a unidade **px**.

[0.4 Valores]

26. No contexto *CSS grid* a identificação das linhas que constituem a *grid*:

- A. pode ser numérica ou baseada em nomes.
- B. é exclusivamente numérica.
- C. é exclusivamente baseada em nomes.
- D. não pode ser efetuada.

[0.4 Valores]

27. A propriedade *grid-area* destina-se a definir:

- A. as dimensões das diferentes áreas da *CSS grid*.
- B. a afetação dos conteúdos às diferentes áreas (previamente identificadas) que constituem a *CSS grid*.
- C. a designação das diferentes áreas da *CSS grid*, o que pode ser muito útil na atribuição dos conteúdos às diversas áreas.
- D. as designações dos diferentes *layouts* que constituem as diferentes áreas da *CSS grid*.

[1.6 Valores]

28. Considere o seguinte código HTML:

```
<div>Exame Tweb!</div>
```

Especifique o código CSS de forma que em caso de sobreposição do cursor do rato, o elemento `<div>` passe progressivamente, durante 3s, de uma largura de 150px para 300px, sendo que a cor de fundo do elemento também é alterada progressivamente de laranja para *cadetblue*. Ao longo de toda a transição, a cor do texto permanece branca com uma dimensão de 130% relativamente à dimensão atual da fonte.

[0.4 Valores]

29. A definição dos *breakpoints* nos *CSS media queries* deve ter em consideração o *device pixel ratio* que pode ser definido como a relação entre os:

- A. píxeis *device* e os píxeis CSS (ex: DPR=2, 2 píxeis *device* formam 1 píxel *CSS*).
- B. píxeis *CSS* e os píxeis *device* (ex: DPR=2, 2 píxeis *CSS* formam 1 píxel *device*).
- C. píxeis *CSS* e os píxeis *device* (ex: DPR=2, 4 píxeis *CSS* formam 1 píxel *device*).
- D. píxeis *device* e os píxeis *CSS* (ex: DPR=2, 4 píxeis *device* formam 1 píxel *CSS*).

[0.4 Valores]

30. O *Responsive Web Design* tem necessariamente de ser implementado com base em imagens:

- A. fixas assim como numa correta definição de *CSS media queries*.
- B. fixas e numa resolução fixa do *viewport*.
- C. flexíveis e numa resolução fixa do *viewport*.
- D. flexíveis assim como numa correta definição de *CSS media queries*.

[0.4 Valores]

31. A declaração de variáveis em *SASS* inicia-se com:

- A. `#`
- B. `$`
- C. `%`
- D. `*`

[0.4 Valores]

32. No contexto do SASS os *partials* são ficheiros que devem ser importados por outros ficheiros `*.scss`, não podendo deste modo ser compilados diretamente para `*.css`. A designação destes ficheiros tem necessariamente de se iniciar por:

- A. `#`
- B. `%`
- C. `_` (*underscore*)
- D. `&`

[0.4 Valores]

33. Um script externo (`*.js`):

- A. só pode ser executado como resposta a um evento.
- B. só pode ser executado se inserido numa função.
- C. tem obrigatoriamente de conter um *event listener*.
- D. pode ser executado sem qualquer controlo após o download da aplicação web.

[0.4 Valores]

34. O *JavaScript* possui alguns *Built-in Objects*, entre os quais é possível destacar:

- A. *Integer/Date/Array*.
- B. *Float/Date/Array*.
- C. *Math/Date/Array*.
- D. *Char/Math/Boolean*.

[0.4 Valores]

35. Em *JavaScript*, após a sua criação de forma literal, um objeto:

- A. pode ser alterado, mas apenas relativamente às suas propriedades (adição/eliminação).
- B. pode ser alterado em qualquer momento, com a adição/eliminação de propriedades/métodos.
- C. não pode ser alterado.
- D. pode ser alterado, mas apenas relativamente aos seus métodos (adição/eliminação).

[0.4 Valores]

36. Em *JavaScript* um *Array* pode armazenar diferentes tipos de dados,:
- A. não possui propriedades nem métodos.
 - B. possui propriedades (ex: *length*) e métodos (ex: *sort()*).
 - C. possui propriedades (ex: *length*) mas não possui métodos.
 - D. possui propriedades (ex: *length*) e métodos (ex: *toUpperCase()*).

[0.4 Valores]

37. No contexto do *JavaScript* a propriedade ***style*** :
- A. é equivalente à propriedade ***css***.
 - B. permite definir na mesma instrução, o valor de múltiplas propriedades CSS em simultâneo.
 - C. permite definir o valor de apenas uma propriedade CSS por instrução, sendo que a alteração de múltiplas propriedades implica múltiplas instruções.
 - D. não se encontra definida.

[0.4 Valores]

38. O método *document.querySelectorAll()* permite aceder a um elemento HTML através de seletores:
- A. *JavaScript* e requer indexação.
 - B. CSS e requer indexação.
 - C. CSS e dispensa indexação.
 - D. CSS e requer indexação com a exceção de quando existe apenas um elemento HTML referenciado pelo seletor.

[0.4 Valores]

39. No contexto do *JavaScript*, o *Event Object*:
- A. é criado automaticamente sempre que é despoletado um evento.
 - B. não existe.
 - C. pode ser criado de forma literal como resposta à ocorrência de um evento.
 - D. só pode ser criado se tiver sido definido um *event listener*.