

# Html And Css Exam

## 1- Explain :has() pseudo-class

دي بيسموها "Parent Selector" علشان بجدها عن طريق اللي جواها يعني مثلاً

```
body>
  <div>
    <img src="" alt="">
  </div>
```

هنا لو عايز اختار أي div جواه صورة بنعمل كده

```
div:has(img) {
```

كده يجيبي كل ال divs اللي جواهم صورة

## 2- What is CSS aspect-ratio property?

الـ aspect-ratio دي خاصية بتخلينا ثبت النسبة بين العرض والطول للعنصر. أهميتها بتبان جداً في الـ Responsive Design بتضمن إن الصور أو الفيديوهات متتضغطش (stretch) لما حجم الشاشة يتغير، وبتحافظ على الـ UI متناسق من غير ما نضطر نحدد ثابت لكل شاشة Height

```
.video-container {
  width: 100%;
  aspect-ratio: 16 / 9;
}
```

## 3- How would you optimize CSS for performance?

أشيل المسافات والتعليقات. **Minification**

أمسح الكود اللي مش مستخدم (بأدوات زي **PurgeCSS**)

أحمل الكود المهم لأول جزء يظهر للمستخدم (**Above the fold**) في الـ **<head>**. **Critical CSS**

لأنها بتعمل تسلسل في التحميل بيقطع الدنيا. **Avoid @import**

## 4- CSS Custom Properties (Variables) vs Sass variables?

- بتقدر تغير قيمتها بالـ JavaScript في وقت التشغيل (Runtime)
- بتقدر تغير قيمتها جوه Media Query (مثلاً لون الـ background يتغير في الموبايل بسطر واحد)
- بينما Sass variables هي "Static"، مجرد ما الكود بيتتحول له CSS عادي قيمتها بتثبت وما ينفعش تتغير في المتصفح.

## 5- Grid Calculation (500px container)

- النتيجة: العمود الأول 50px، الثاني 150px، الثالث 300px

## 6- What is the Critical Rendering Path (CRP)?

هي الخطوات اللي المتصفح بي Shirley فيها من أول ما يستلم كود الـ HTML لحد ما يرسم البيكسلات على الشاشة. بتعدى بـ 5 مراحل:

1. بناء الـ DOM (Document Object Model): المتصفح بيقرأ كود الـ HTML وبيحوله لـ "شجرة من العناصر" (Nodes) دي المرحلة اللي المتصفح بيفهم فيها هيكل الصفحة (مين الأب ومنين الابن).
2. بناء الـ CSSOM (CSS Object Model): المتصفح يحلل ملفات الـ CSS وقواعد التنسيق. المرحلة دي مهمة عشان المتصفح يعرف كل عنصر ستايله إيه، وايه الخصائص اللي هيورثها من العناصر اللي فوقية.
3. بناء الـ Render Tree: هنا المتصفح يدمج الـ DOM مع الـ CSSOM النتيجة تكون شجرة تانية خالص فيها بس العناصر اللي "هتظهر" فعلاً للمستخدم (يعني أي حاجة واحدة display: none بتقسى من الشجرة دي).
4. مرحلة الـ Reflow (أو Layout): بعد ما عرفنا إيه اللي هيظهر، المتصفح بيبدأ يحسب "الأرقام". يحدد بالضبط كل عنصر مكانه فين (Coordinates) ومساحته كام بيكسلي بالنسبة لحجم شاشة المستخدم (Viewport).
5. مرحلة الـ Paint: دي آخر مرحلة، وهي "التلوين". المتصفح بيمسك كل الحسابات اللي عملها وبيحولها لبيكسلات حقيقة على الشاشة. هنا بيترسم النص، الألوان، الصور، الظل (Shadows)، والحواف (Borders) (Pixels).

## 7- What is the purpose of clamp()?

دي بتعمل "Responsive Typography" أو أحجام مرنة من غير Media Queries بتأخذ 3 قيم: (أقل حجم، الحجم المثالي، أكبر حجم).

- مثال: `font-size: clamp(1rem, 5vw, 2rem)`
- ده معناه الخط هيكبر ويصغر مع الشاشة بس مش هيقل عن rem1 ولا هيزيyd عن rem2

## 8- What is 'Layout Thrashing'?

دي مشكلة أداء بتحصل لما الا JavaScript يقدر يقرأ ويعدل في الا DOM ورا بعضه بسرعة (زي إنك تجيبي offsetHeight وبعدين تغير style.height جوه) ده بيجر المتصفح إنه يعيد حسابات الا Layout كلها مرة في ثانية واحدة، فيدخل الصفحة "تهنج"

## 9- Best way for three choices (Select only one)?

أفضل وأصح طريقة هي استخدام الا (`<input type="radio">`) name attribute و يكون لهم نفس الا Radio Buttons يضمن ان المتصفح لو حده يخليلك تختار واحد بس، وكمان هي الطريقة الأصح لـ Accessibility (عشان اللي يستخدموا Screen Readers)

## 10- :nth-child for every 3rd item starting from 2nd?

المعادلة هتكون:

`:nth-child(3n + 2)`

- الشرح:  $3n$  يعني كل 3 عناصر، و  $2 +$  يعني ابدأ من العنصر الثاني.

## QUESTION 3

### 1- Flex-direction Values:

The flex-direction property defines the main axis along which flex items are placed in the flex container.

- Example 1:** row (Default direction; items are placed from left to right).
- Example 2:** row-reverse (Items are placed horizontally but in reverse order).
- Example 3:** column (Items are placed vertically from top to bottom).
- Example 4:** column-reverse (Items are placed vertically from bottom to top).

### 2- Blue Elements Selection:

The selector used is `h2 ~ p`, which is the General Sibling Combinator. It selects all `<p>` elements that follow an `<h2>` element within the same parent container.

Analysis of the HTML structure:

- P1:** Will NOT be blue because it precedes the `<h2>`.
- H2:** The reference element.
- P3:** Will be BLUE because it is a sibling that follows the `<h2>`.
- P4:** Will be BLUE because it is also a sibling that follows the `<h2>`.

