**Question 1: What is a CSS Selector?**

A **CSS selector** is a pattern used to select and apply styles to specific HTML elements in a web page. It defines which elements a set of CSS rules should apply to.

**Examples of CSS Selectors:**

1. **Element Selector**: Targets all HTML elements of a specific type.
2. p {
3. color: blue;
4. }

*Applies the blue color to all <p> elements.*

5. **Class Selector**: Targets elements with a specific class attribute. Prefixed with a dot (.).
6. .highlight {
7. background-color: yellow;
8. }

*Applies a yellow background to all elements with the class highlight.*

9. **ID Selector**: Targets a single element with a specific id attribute. Prefixed with a hash (#).
10. #header {
11. font-size: 24px;
12. }

*Applies font size of 24px to the element with the id of header.*

---

**Question 2: Explain the Concept of CSS Specificity**

**CSS specificity** determines which style rules take precedence when multiple rules target the same element. It is calculated based on the types of selectors used.

**Specificity Calculation:**

1. **Inline Styles**: Highest specificity (e.g., style="color: red;").
2. **ID Selectors**: High specificity (#id).
3. **Class Selectors, Attributes, and Pseudo-classes**: Medium specificity (.class, [type="text"], :hover).
4. **Element Selectors**: Low specificity (h1, p).

**How Conflicts Are Resolved:**

1. **More Specific Rule Wins**:
    - #header (ID) overrides .highlight (Class).

2. **Equal Specificity**: The **last rule** in the stylesheet wins.
    - Example:
    - h1 {
    - color: blue;
    - }
    - h1 {
    - color: red;
    - }

*h1 will be red because the second rule is applied last.*

3. **Inline Styles Override** All External or Internal Rules:
4. <h1 style="color: green;">Hello</h1>

*The inline color: green takes precedence.*

---

**Question 3: Difference Between Internal, External, and Inline CSS**

**1. Internal CSS**

- Defined within a <style> tag in the <head> section of an HTML document.

```
<style>
  h1 {
    color: blue;
  }
</style>
```

**Advantages**:

- Easier to test small styling changes.
- Keeps styles centralized within the document.

**Disadvantages**:

- Not reusable across multiple pages.
- Can clutter the HTML file.

**2. External CSS**

- Written in a separate .css file and linked using the <link> tag.

```
<link rel="stylesheet" href="styles.css">
```

**Advantages**:

- Styles are reusable across multiple pages.
- Keeps HTML clean and focused on structure.

**Disadvantages**:

- Requires an additional HTTP request to load the CSS file.
- May not be suitable for small projects with minimal styles.

**3. Inline CSS**

- Applied directly to an element using the style attribute.

```
<h1 style="color: red;">Hello</h1>
```

**Advantages**:

- Useful for quick fixes or testing.
- High specificity, overriding external/internal styles.

**Disadvantages**:

- Not reusable.
- Difficult to maintain, especially in large projects.
- Violates the separation of structure (HTML) and presentation (CSS).

---

**Summary**:

- Use **external CSS** for large projects to ensure reusability and maintainability.
- Use **internal CSS** for single-page projects or quick styling in prototypes.
- Use **inline CSS** sparingly, mostly for testing or overriding.