ASSIGNMENT NO: 04

GROUP NO: 02

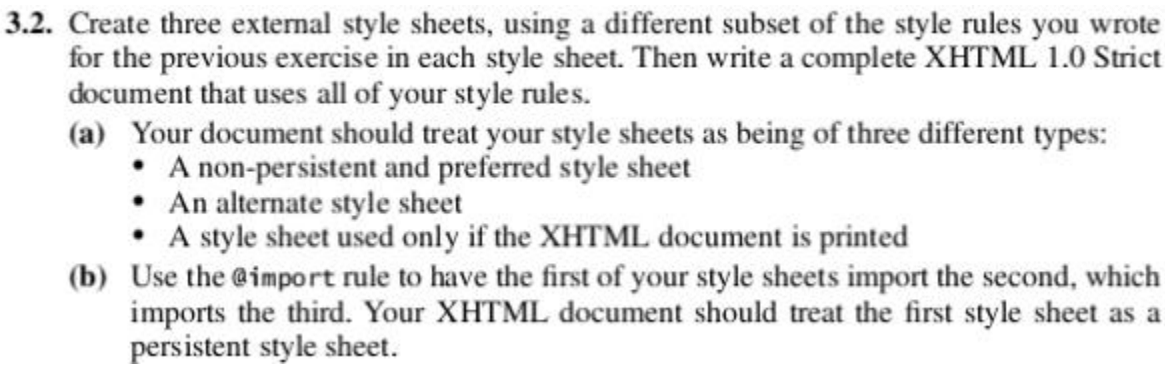
CHAPTER # 03 (CASCADING STYLE SHEET)

QUESTION NO: 3.2 & 3.19

REPRESENTED TO: MISS HUMERA TARIQ

UMAER BASHA INSTITUTE OF INFORMATION AND TECHNOLOGY

Question: 3.2



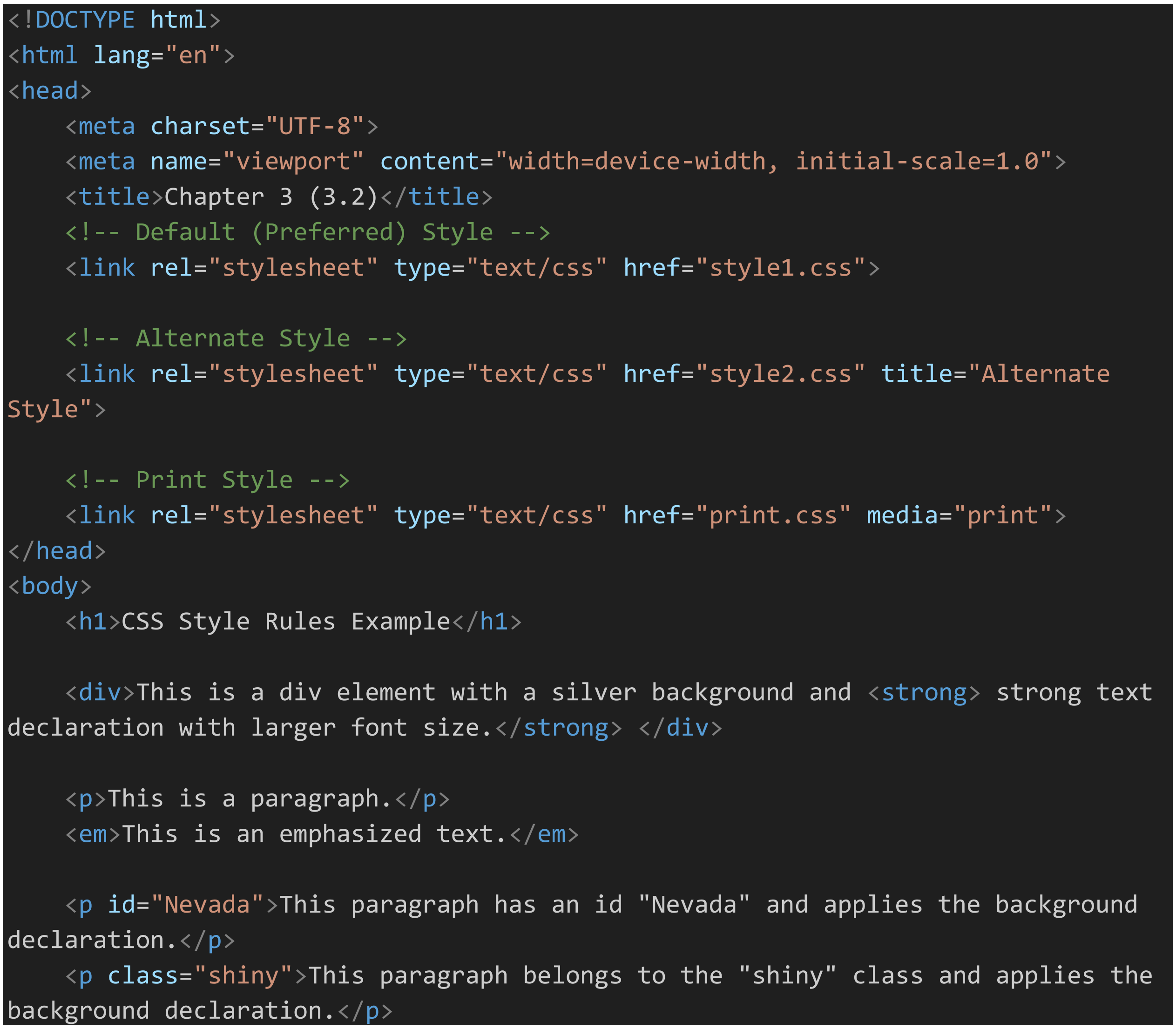
3.2. Create three external style sheets, using a different subset of the style rules you wrote for the previous exercise in each style sheet. Then write a complete XHTML 1.0 Strict document that uses all of your style rules.

(a) Your document should treat your style sheets as being of three different types:

- A non-persistent and preferred style sheet
- An alternate style sheet
- A style sheet used only if the XHTML document is printed

(b) Use the `@import` rule to have the first of your style sheets import the second, which imports the third. Your XHTML document should treat the first style sheet as a persistent style sheet.

index.html



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Chapter 3 (3.2)</title>
    <!-- Default (Preferred) Style -->
    <link rel="stylesheet" type="text/css" href="style1.css">

    <!-- Alternate Style -->
    <link rel="stylesheet" type="text/css" href="style2.css" title="Alternate
Style">

    <!-- Print Style -->
    <link rel="stylesheet" type="text/css" href="print.css" media="print">
</head>
<body>
    <h1>CSS Style Rules Example</h1>

    <div>This is a div element with a silver background and <strong> strong text
declaration with larger font size.</strong> </div>

    <p>This is a paragraph.</p>
    <em>This is an emphasized text.</em>

    <p id="Nevada">This paragraph has an id "Nevada" and applies the background
declaration.</p>
    <p class="shiny">This paragraph belongs to the "shiny" class and applies the
background declaration.</p>
```

```html
    <p>
        <span class="bigger">This span has the "bigger" class and applies the text declaration.</span>
    </p>

    <p>
        <span>This is a span element containing
            <span>a nested span that also applies the text declaration.</span>
        </span>
    </p>

    <p>
        <a href="#">This is a hyperlink. Hover over it to see the background change.</a>
    </p>
</body>
</html>
```

style1.css

```css
@import url('style2.css');

div, #Nevada, .shiny {
    background-color: silver;
}

strong, p, em, .bigger, span {
    font-size: larger;
}

a:hover {
    background-color: silver;
}
```

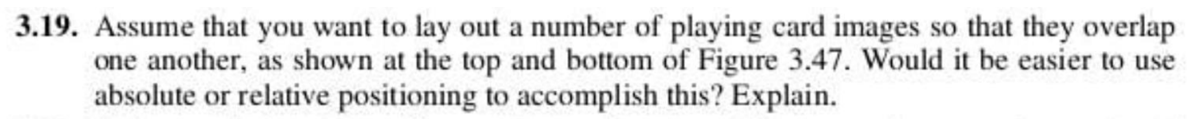
style2.css

```css
p, em {
    font-size: larger;
    background-color: silver;
}
```

print.css

```
p {
    font-size: larger;
}
```

Question: 3.19



3.19. Assume that you want to lay out a number of playing card images so that they overlap one another, as shown at the top and bottom of Figure 3.47. Would it be easier to use absolute or relative positioning to accomplish this? Explain.

Relative positioning for the overlapping playing card layout would be less easy and less effective than absolute positioning. Here’s why:

1. **Fine Control OverPositions:**
Using absolute positioning you can set the position of all cards where you want with top, left, z-index etc. This means that cards can overlap seamlessly, and the right z-ordering is followed.
2. **Avoiding Unexpected Shifts:**
It takes the element out of normal document flow; meaning it can position itself based on where it would be normally in the document flow.
3. **Stacking Order with z-index:**
With absolute positioning, it is possible to control which card is on top with z-index.
This is crucial where the aesthetics of the card arrangement are concerned.

Conclusion:
By using absolute positioning, ensuring that each card is accurately positioned to create an overlapping effect is accomplished without layout problems