

2025 CpSc H Q18

Section: Web Design and Development

Topic: Implementation Using HTML, CSS and JavaScript

Question Summary

- (a) Draw the navigation structure for a car marketplace site.
- (b) Rewrite the HTML input for price so it must be entered and be between £ 200 and £ 75,000.
- (c) (i) Fix CSS so explanatory text appears below the floated sections; (ii) set Arial font for text in Silver/Gold sections.
- (d) (i) Explain why the four-image layout is not fit for purpose even full screen; (ii) explain why it still fails on smartphones.
- (e) Explain block vs inline behaviour for the paragraphs and the anchor.

Worked Solution

(a) Navigational structure

Home

- | — Create account
- | — List your car
- | — Search
 - | — New cars
 - | — Used cars
 - | — Electric cars
- | — Car leasing (info)

(b) Validated price input (single rewritten line)

```
<input type="number" name="price" min="200" max="75000" required>
```

This enforces required entry and restricts values to £ 200 – £ 75,000.

(c)(i) Place the explanatory <div> below the floated sections

```
div { clear: both; }
```

Clearing both floats forces the div onto the next line beneath #silver and #gold.

(c)(ii) Apply Arial to text in both sections

```
section { font-family: Arial; }
```

Both Silver and Gold are <section> elements, so this targets the required text.

(d)(i) Why not fit for purpose on full screen

Each image is 600px wide with margin 10px on each side; effective width per image = $600 + 20 = 620\text{px}$. Two images side-by-side require 1240px, but the container div is set to 1210px, so the second image wraps or overlaps. (Also, `display:block` is redundant with `float:left` and complicates layout.)

(d)(ii) Why it still fails on smartphones

The layout uses fixed pixel widths and no responsive rules (e.g., no `<meta name="viewport" ...>`, no media queries, no flexible grid). Mobile browsers shrink/zoom the page to fit, so the 600px images and 1210px container do not reflow to the small viewport, breaking the intended layout.

(e) Block vs inline behaviour

`<p>` elements are block-level, so each paragraph starts on a new line and occupies the full available width. `<a>` is an inline element, so it flows within the same line as paragraph text unless styled otherwise. This is why the anchor appears on the same line as the surrounding paragraph.

Final Answer

(a) Site map shown (Home → Create account, List your car, Search → New/Used/Electric, Car leasing).

(b) `<input type="number" name="price" min="200" max="75000" required>`

(c)(i) `div { clear: both; }` (places text below floated sections).

(c)(ii) `section { font-family: Arial; }`

(d)(i) Container width 1210px is too narrow for two images of effective width 620px each (needs $\geq 1240\text{px}$).

(d)(ii) Fixed pixel widths and no responsive meta/CSS mean smartphones still render incorrectly.

(e) `<p>` is block-level (new lines), `<a>` is inline (stays on the same line).

Revision Tips

- Use `clear` to control flow around floats; consider Flexbox/Grid instead of floats for modern layouts.
- For forms, enforce ranges with `min/max/required` and suitable step.
- Make layouts responsive: `viewport` meta tag, media queries, percentage/flex units instead of fixed pixels.

- Remember block vs inline behaviour affects line breaks and flow.

Home

Create
account

List
your car

Search

New
cars

Used
cars

Flight

Electric
cars

Car
leasing
(info)

