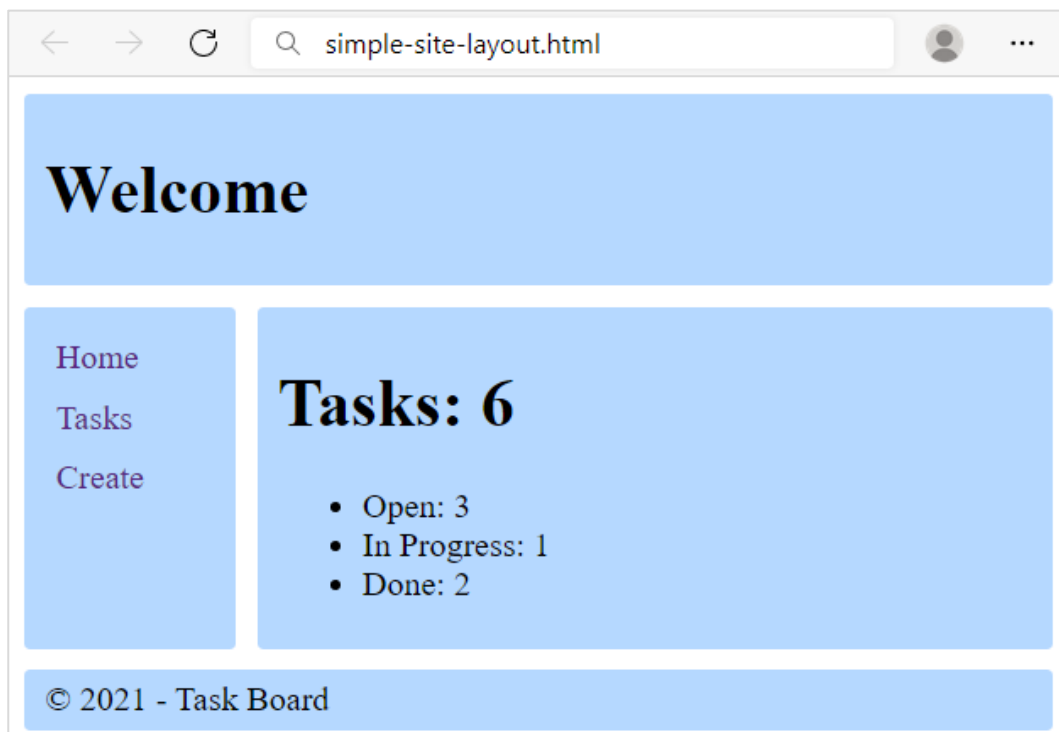# Exercise: Position & Grid

Problems for exercises and homework for the "HTML and CSS" course @ SoftUni. Submit your solutions in the SoftUni Judge system at https://judge.softuni.org/Contests/3334/Position-and-Grid.

1. Simple Site Layout

Create a Web page, holding like the following:

Use the texts from the file **site-texts.txt**.

Create two files: **simple-site-layout.html** and **styles.css**.

Change the document title to "**Simple Site Layout**"

Requirements

<body> that contains: **header**, **aside**, **main**, **footer** all with:

- o border-radius: **3px**;
- o background: **rgb(181, 216, 255)**;
- o padding: **5px 10px**;

The elements into the **body**:

- **<header>** with
  - o **<h1>** text that is displayed as **inline block** and is **vertically aligned** in the **middle**;
- **<aside>** tag that contains:
  - o **<ul>** tag for unordered list
    - list-style-type: **none**;
    - margin: **0px**;
    - padding: **5px**;
    - **<li>** tag for list item

- **`<a>`** tag for hyperlink
  - display**: block;**
  - padding**: 5px 0px;**
  - color**: rgb(86, 40, 129);**
  - text-decoration**: none;**
- **`<main>`** tag that contain:
  - **`<h1>`** text title for the tasks
  - **`<ul>`** tag with:
    - **`<li>`** for the tasks with their status
- **`<footer>`** tag that contain:
  - **`<div>`** with copyright sign and text

CSS Grid

In the **styles.css:**

- Make the **body** grid container by **displaying** the **grid**.
- Define **two grid columns** and their size. **100px** for the first column and **auto** for the second.
- **Define** the **grid areas: header**, **main**, **aside**, **footer**.
- **Reference** the **grid areas**: **"header header"**, **"aside main"**, **"footer footer"**.
- Set **gap: 10px**.
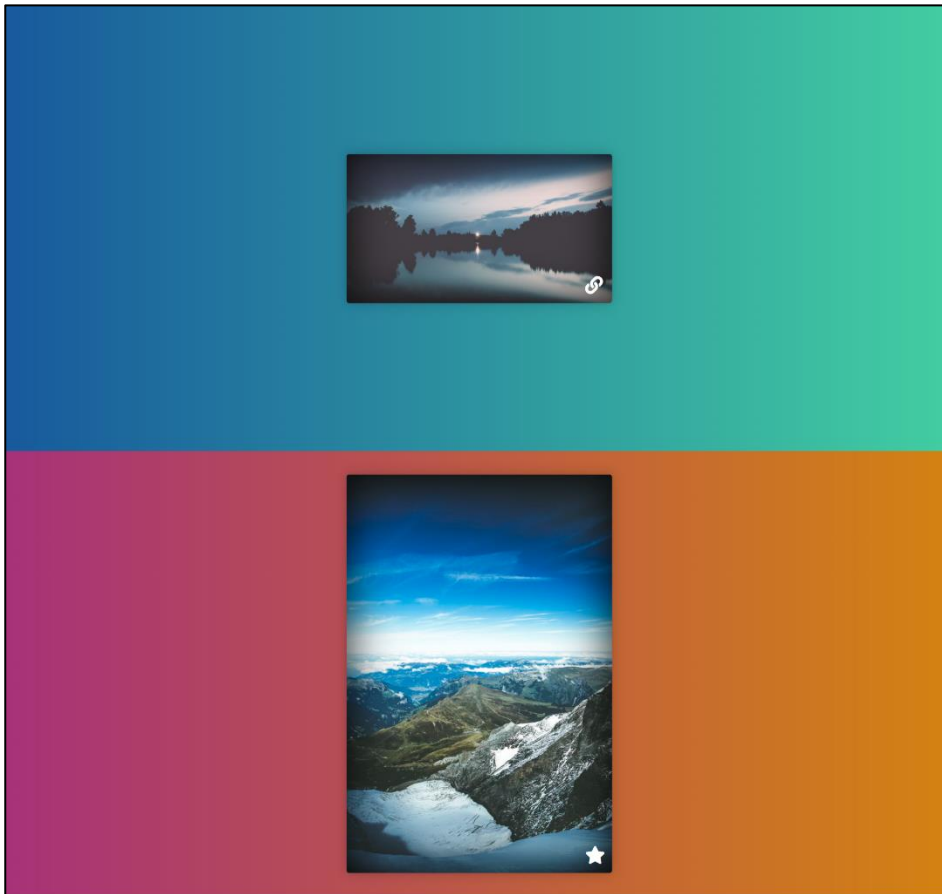
Hints

```
body {
    display: grid;
    grid-template-areas:
        "header header"
        "aside main"
        "footer footer";
    grid-template-columns: 100px auto;
    gap: 10px;
}

header { grid-area: header; }
main { grid-area: main; }
aside { grid-area: aside; }
footer { grid-area: footer; }
```

2. Position Playground
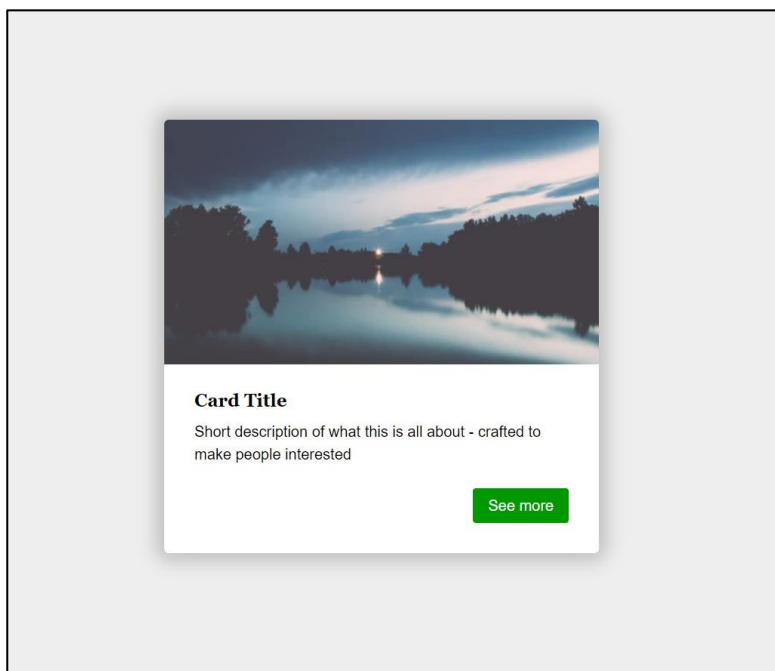
Create a web page like the following:

- Change the document title to "**Positioning Playground**"
- Import Font Awesome into your **CSS** file

Requirements

- Add two container units **(divs)** with class `viewport` inside the body
  - Inside each one adds section with class **card** *(`section.card`)*
    - Change the display property to `block`
    - Try to center the section with position property `absolute`

3. Center Position and Transform
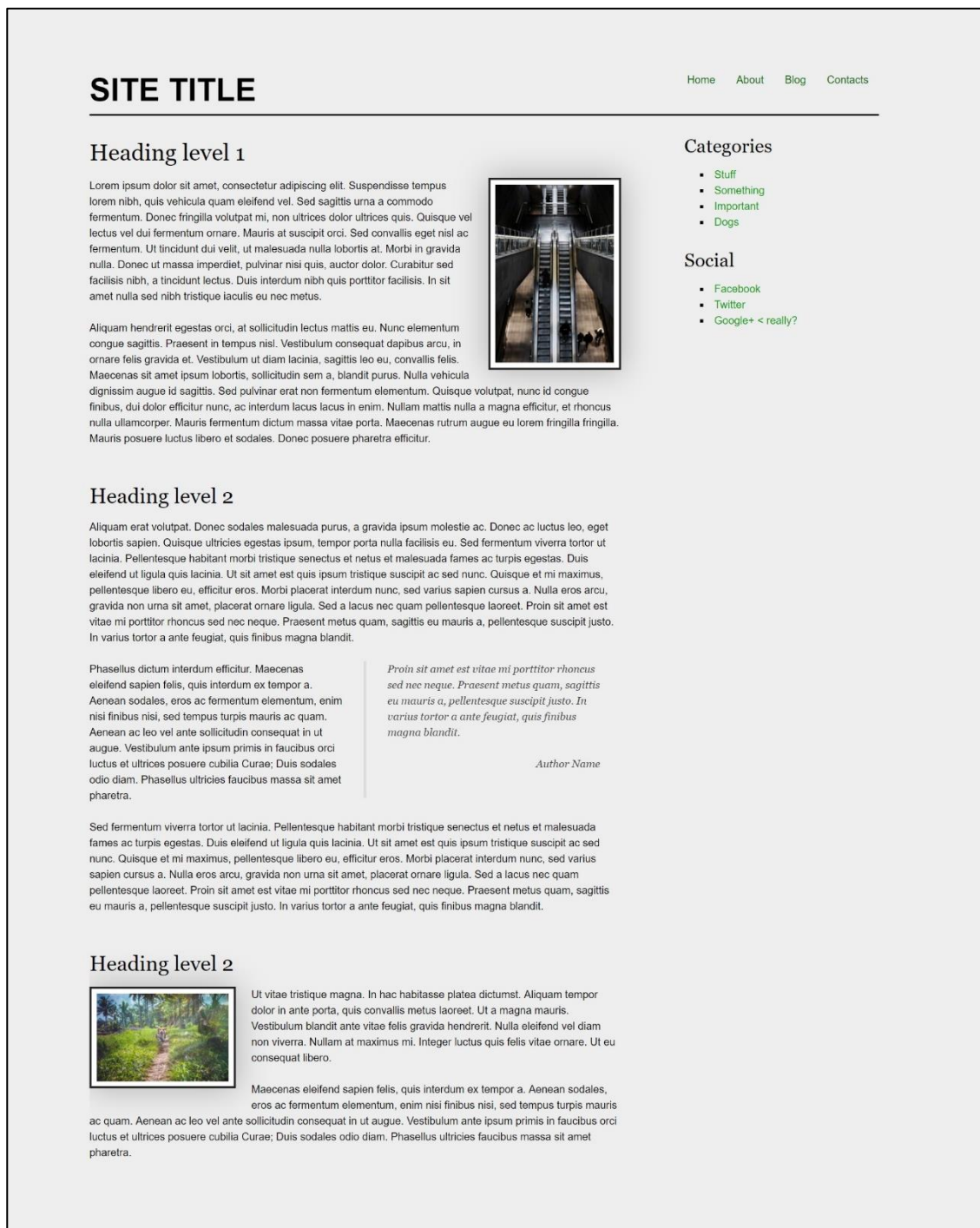
Create a web page like the following:

- Change the document title to "**Center Position and Transform**"

Requirements

- **Section** tag must have **class="card"**
- Create section with two **divs**
  - First div contains image
  - Second div contains: **<h3>**, **<p>** and **a** tags
- Use **rgb(238, 238, 238)** for body background
- Use **rgb(255, 255, 255)** for card background
- Use **rgb(0, 153, 0)** for card button background
- Use font **Georgia, serif** with size **1em/1.2** for the headings
- Set the image position property to **absolute**

Follow us:

4. Navigation

Create a web page like the following:
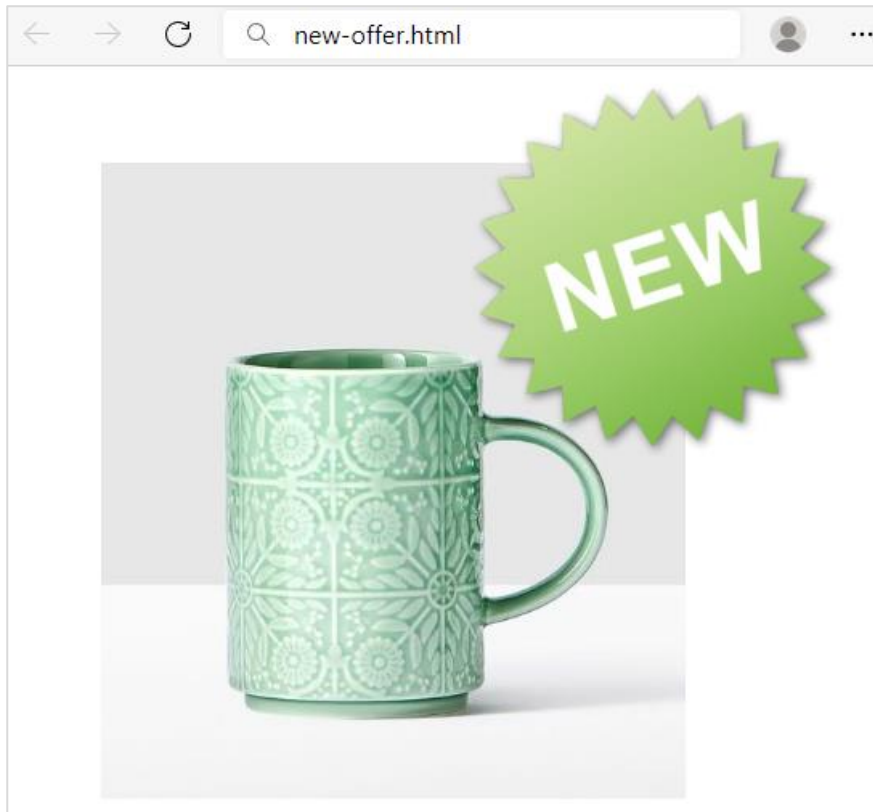

- Change the document title to "**Navigation**"

Requirements

- Create a **header** with a **p** for as title
  - Inside the **header** add a **nav** with a **ul** and 4 **li** elements
- Create a **main**, which contains an **article** with a **header** and **h1** heading

- o Add a **section** inside the **article**
- Create an **aside** element with two sections inside
  - o The sections should have a **h3** heading and a navigation - **nav** with **ul**, **li** and **a** elements
- Use **rgb(0, 102, 0)** for the anchors
- Use **rgb(238, 238, 238)** for body background
- Use **rgb(255, 255, 255)** for image background and **rgba(0, 0, 0, 0.25)** for image box shadow
- Use font **Georgia, serif** for the blockquote
- Use font **Georgia, serif** with size **1em/1.2** for the headings

5. New Offer

Create a web page like the following:

Requirements

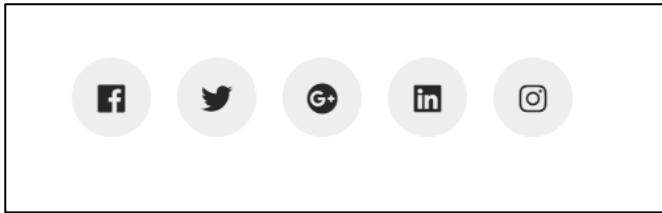**HTML** with **<body>** that contains **<main>** and two **<img>**.

<img> with **new.png** need **class="new"**.

CSS:

- **<main>**:
  - o Width: **700px**
  - o Height: **500px**
  - o Margins: **60px**
- **img.new**:
  - o **Relative** position
  - o Top coordinates: **-200px**
  - o Right coordinates: **150px**

6. Social Media Icons
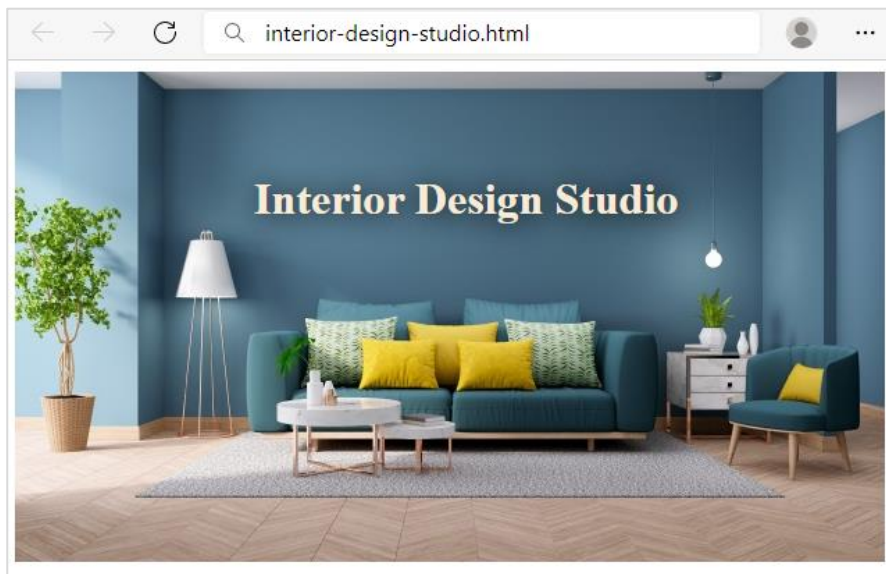
Create a web page like the following:

- Change the document title to "**Social Media Icons**"
- Use **FontAwesome** to add social icons

Requirements

- All you need for your HTML are **ul**, **li** and **a** tags
- Change the **ul** display property to **inline-block**
- Remove list items default style
- Each **a** tag must have **border-radius: 50%;**
- Check the provided screenshots to see the **hover** effects

7. Interior Design Studio

Create a webpage like the following:

Requirements

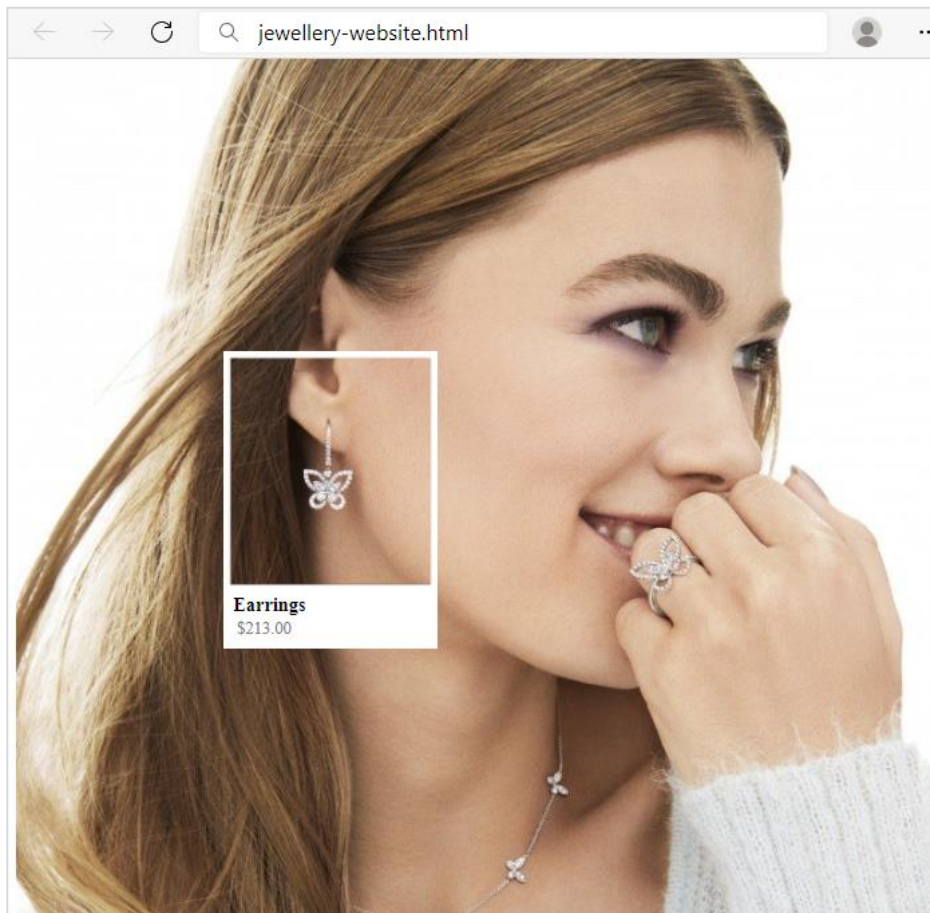**HTML** with **<body>** that contains **<h1>** and **<img>**.

CSS:

- **<h1>**
  - **Absolute** position
  - **Top** coordinates: **60px**

---

o **Left** coordinates: **180px**
o Text color: **antiquewhite**
o `text-shadow: 1px 1px 20px black;`

8. Jewellery Website

Requirements

You will be given a skeleton with **filled HTML** and an **empty CSS** file. Write the missing **CSS** following the guidelines in the task.

- `<section>`
  o Set the position to **absolute**
  o Use the **coordinates**: **250px** and **180px** to position the **frame** correctly
- `<h3>`
  o Font size: **16px**
  o Margin: **0px**
- `<span>`
  o Padding left: **3px**
  o Text color: **gray**
  o Font size: **14px**
- `<div>`
  o Set the position to **relative**
  o Use the **coordinates**: **50px** and **8px** to position the **text** correctly