**Домашнее задание**

**Week 5**

### Вопросы

1. Почему в большинстве ситуаций стоит использовать flexbox-позиционирование?

Это самый новый способ, позволяет создать адаптивный дизайн, удобный в использовании (табличная верстка устарела, блочная модель может быть неудобной из-за различия между указанным и фактическим размерами блоков, float не всегда используем, тк возникают проблемы с обтекаемостью)

1. Самостоятельно изучите способ позиционирования через display: table и ответьте на вопрос, для каких ситуаций оно лучше всего подходит?

Его можно использовать для таблиц, подобно тегу <table>.

Также с его помощью можно сделать выравнивание по вертикали элементов с динамической высотой или горизонтально расположить по центру динамические элементы (в сочетании с margin:auto).

1. Какими способами можно сделать горизонтальное выравнивание по центру? Минимум 3 варианта, можно больше

* В системе flexbox:

justify-content: center; (если flex-direction: row)

align-items: center; (если flex-direction: column)

* Установить margin:auto для блока; (будет и вертикальное, и горизонтальное выравнивание)
* Для строк в тексте: text-align:center;
* Абсолютное позиционирование: position:absolute и добавить отступ left:50% (не уверена, что это сработает, разве что для очень коротких элементов возможно)

1. Какие есть оси во флекс-верстке и как задается их направление?

Главная ось и поперечная ось.

Направление задается через flex-direction: row/row-reverse/column/column-reverse

1. Разберитесь, как работает свойство margin: auto во флекс-верстке, приведите пример использования

Это свойство позволит идеально отцентрировать элемент по вертикали и горизонтали одновременно. Для этого родительскому контейнеру нужно присвоить значение display:flex, а дочернему элементу margin:auto.

.parent {

display: flex;

.child {

margin: auto;

}

1. В чем преимущества box-sizing?

Это свойство позволяет включить border и padding в заданную ширину и высоту блока (если установить box-sizing:border-box)

1. Чем отличается flex-grow от flex-shrink?

Если установить flex-grow:1, то элемент будет поглощать свободное пространство и расширяться), a flex-shrink:1 позволит «ужать» элемент.

1. Как можно добиться следующего позиционирования элементов:

\*картинка\*

Используя flexbox (display:flex), задать justify-content: space-between;

Или же попробовать margin:auto;

1. Какой в итоге будет размер у элемента (можно округлить)?

\*картинка\*

Ширина: 149px;

Высота: 18px

1. Самостоятельно разберитесь, зачем нужно свойство order?

Оно нужно для того, чтобы установить порядок следования элементов. По умолчанию order:0. Вместо 0 можно брать целые и отрицательные числа (эти числа не будут являться порядковыми номерами, но будут задавать «вес» элементов)

1. Каким кодом можно сделать такую таблицу?

\*картинка\*

    <table border="1">

        <tr>

            <td> Columu 1 </td>

            <td> Columu 2 </td>

            <td> Columu 3 </td>

        </tr>

        <tr>

            <td rowspan="2"> Row 1 Cell 1 </td>

            <td> Row 1 Cell 2 </td>

            <td> Row 1 Cell 3 </td>

        </tr>

        <tr>

            <td> Row 2 Cell 2 </td>

            <td> Row 2 Cell 3 </td>

        </tr>

        <tr>

            <td colspan="3"> Row 3 Cell 1 </td>

        </tr>

    </table>

border=”1” использовала для наглядности

1. Изучите материалы и найдите, для каких ситуаций подходит position: fixed?

position:fixed фиксирует элементы на одном месте даже при прокрутке сайта. Такое позиционирование хорошо подходит для меню и других элементов навигации, которые всегда должны быть под рукой.

1. Каким способом лучше всего верстать большие блоки текста?

С помощью блочной модели

1. Как рассчитывается размер flex-контейнера?

Считаем рамки+отступы+размеры содержимого.