**Вопросы 💎**

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

Flexbox позволяет создать адаптивный дизайн намного проще, чем с использованием других видов позиционирования.

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

* Для динамического горизонтального выравнивания по центру
* Для динамического вертикального выравнивания по центру
* Для динамического «прилипающего» подвала

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

* margin: auto (установка ширины элемента предотвратит его вытягивание до краёв контейнера. Затем элемент займет указанную ширину, а оставшееся пространство будет разделено поровну между двумя полями)
* text-align: center
* justify-content: center
* grid

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

Главная и поперечная, направление задается через flex-direction

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

В разметке flexbox присутствует конкретный вариант использования, в котором, возможно, понадобится атрибут justify-self. Например, когда будет необходимо разделить набор flex элементов для создания отдельного шаблона навигации. В таком случае, можно использовать атрибут margin со значением auto. Атрибут с таким значением будет занимать все доступное пространство в своём измерении. Так работает центрирование блока со свойством margin:auto. Установив значение атрибутов margin-left и margin-right в значение auto, обе стороны нашего блока попытаются занять все доступное пространство и, таким образом, заставят двигаться блок в центр.

Установив значение атрибутаmargin в auto для одного элемента в наборе с flex-элементами, выровненных с начала, можно создать разделённую навигацию. Это хорошо работает с разметкой flexbox и свойствами выравнивания. Как только для элемента со свойством margin: auto не остаётся свободного места, он станет вести себя также, как и все остальные flex элементы, и сократится в объёме, пытаясь уместиться в доступное пространство.

1. **В чем преимущества box-sizing?**

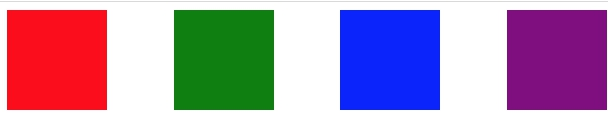
Он дает возможность задавать исключения и поддерживается почти всеми браузерами. И самое главное: При помощи box-sizing: border-box мы можем сказать браузеру, что ширина, которую мы ставим, относится к элементу полностью, включая border и padding:

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

Flex-grow определяет, может ли начальная ширина flex-элемента увеличиваться (расти)

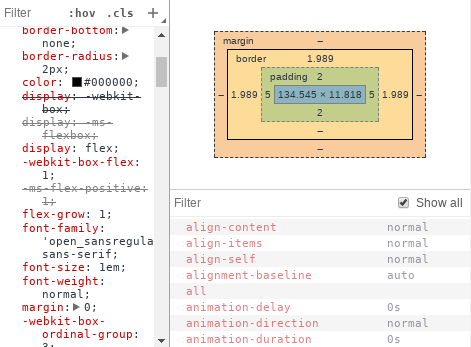
Flex-shrink определяет, может ли ширина flex-элемента уменьшиться.

1. **Как можно добиться следующего позиционирования элементов?**



Через свойство justify-content: space-between

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



139x18

**10. Самостоятельно разберитесь, зачем нужно свойство order?**

Стиль ордер меняет порядок - можно отобразить элементы не в том порядке, в котором они расположены в верстке