**Курс веб-разработке**

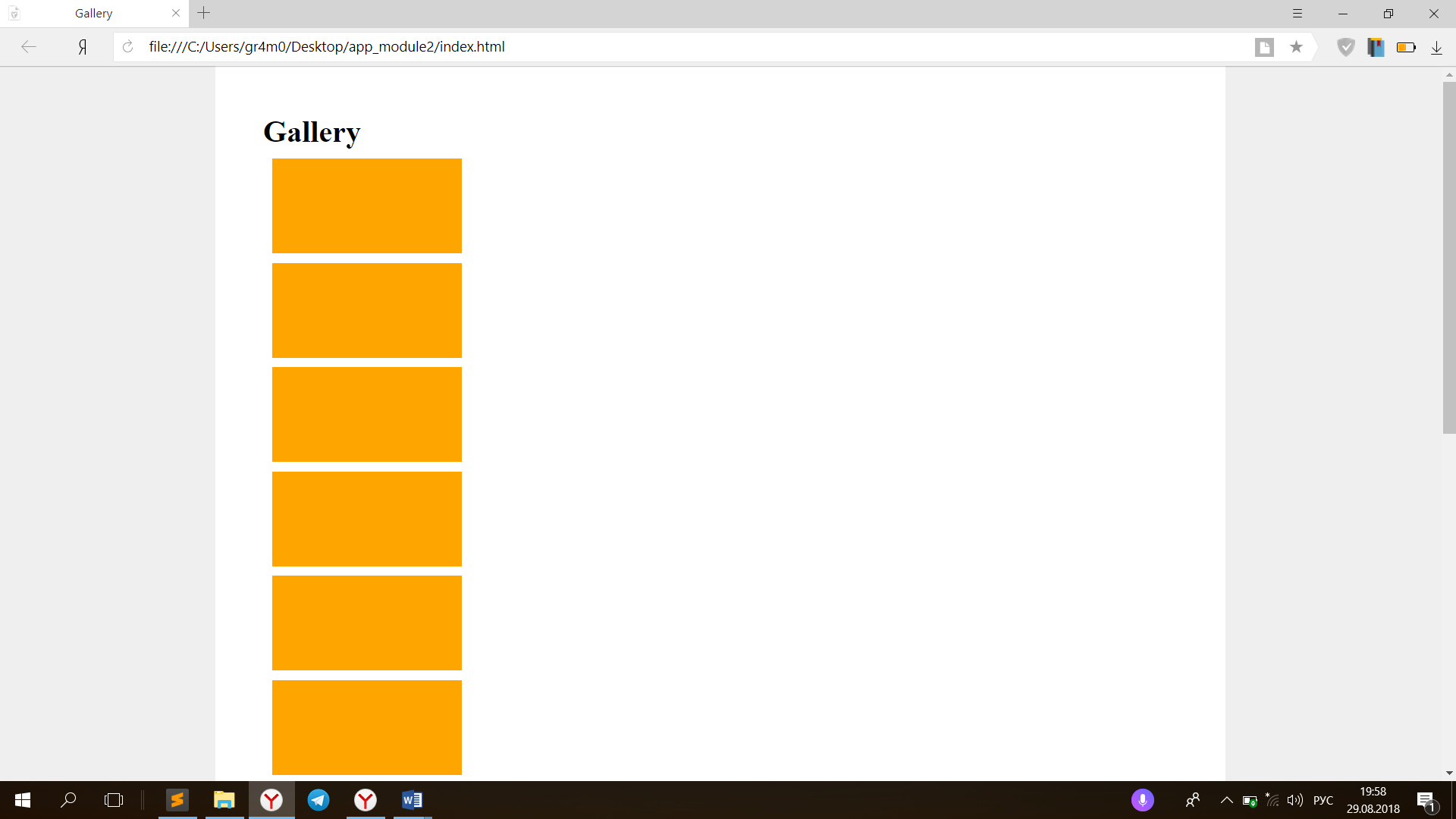
**Второй модуль**

**Третье занятие**

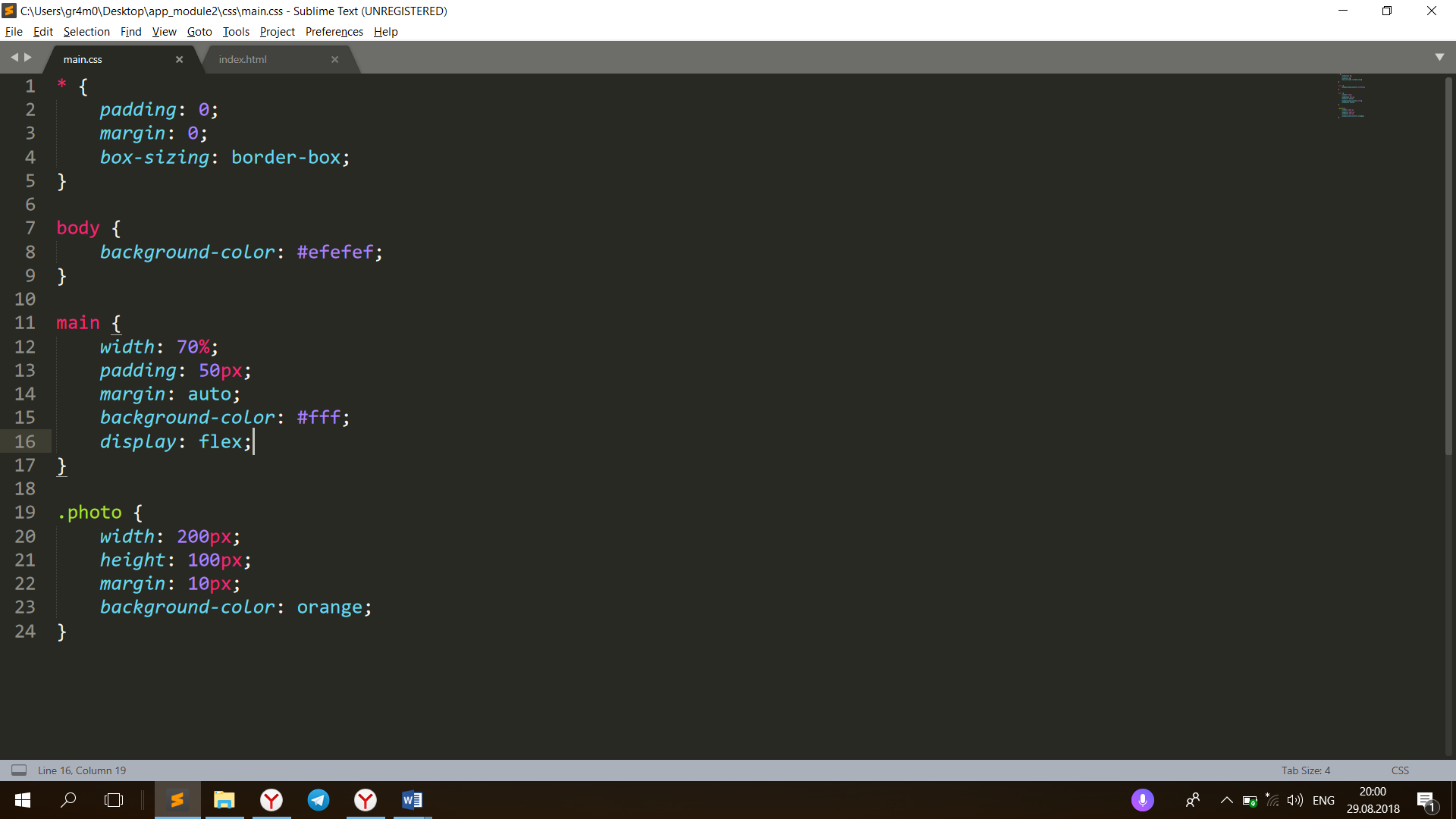
Цель данного занятия — научиться позиционировать элементы на веб-странице.

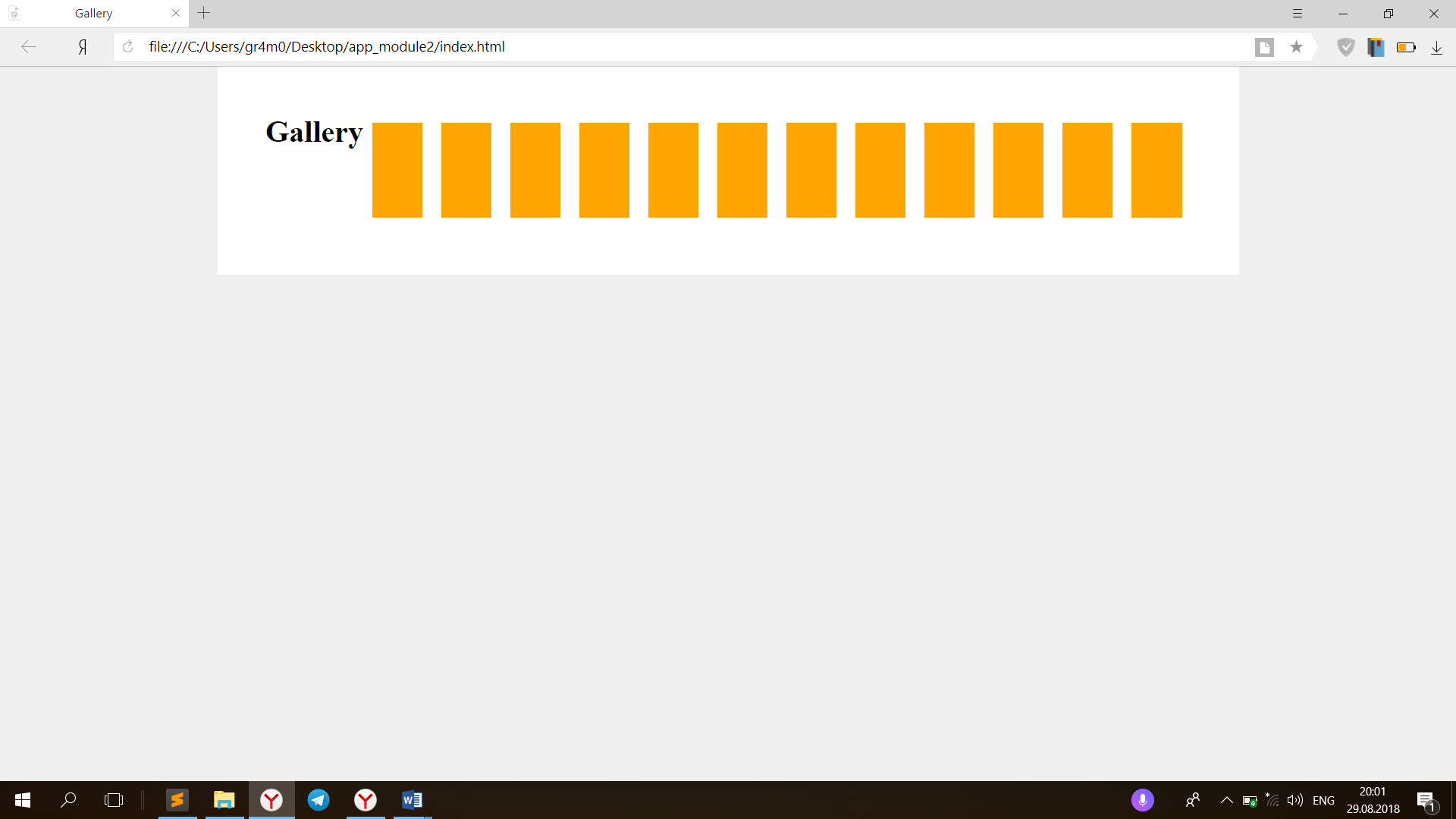
Итак, давайте познакомимся со свойством display. Рассмотрим его основные значения:

* inline — значение по умолчанию, строчное отображение, элемент не переносится на следующую строчку;
* block — блочное отображение, элемент переносится на следующую строчку;
* flex — отображение для контейнера, так называемая гибкая сетка.

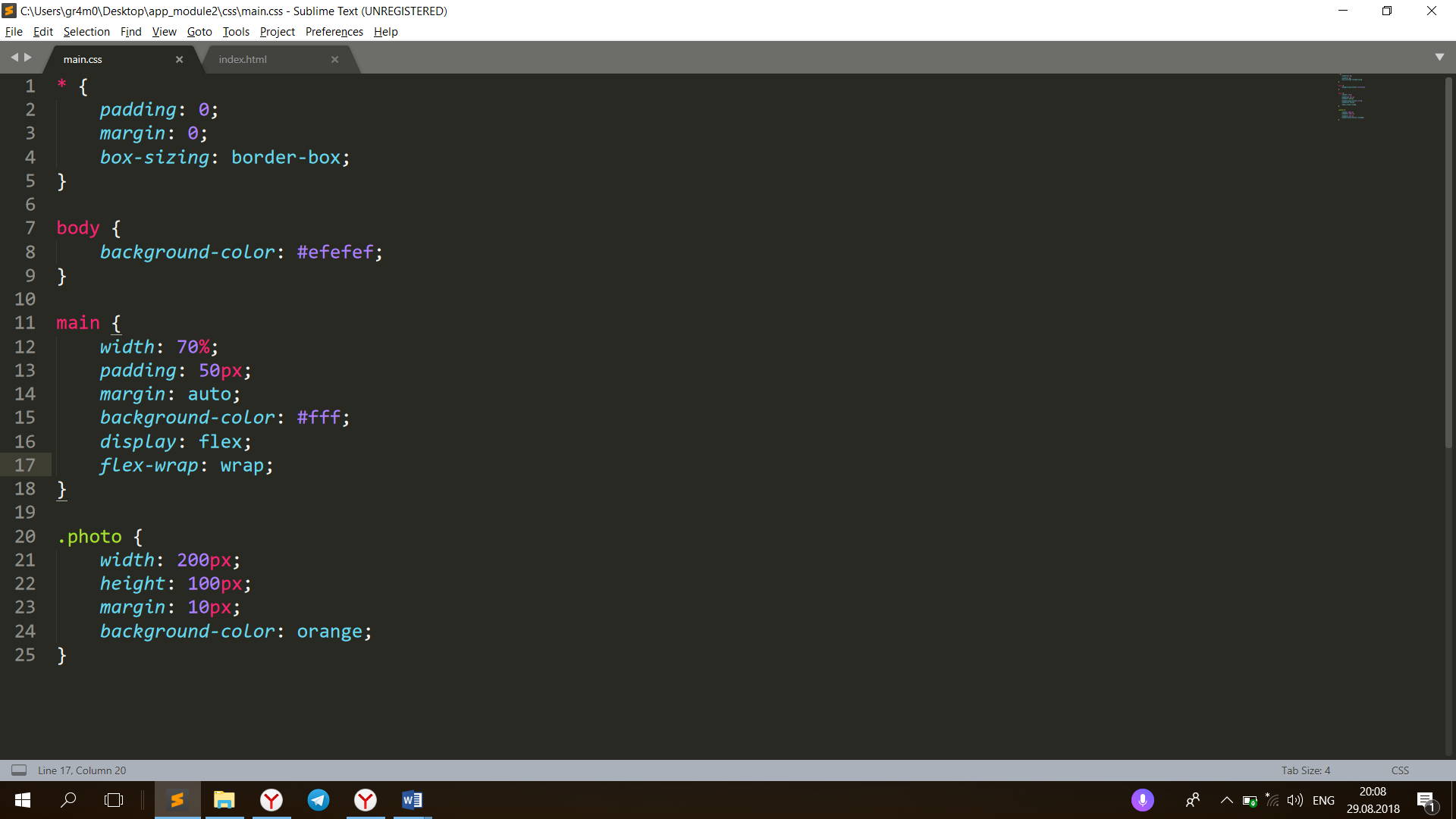
У «дива» по умолчанию отображение блочное, это легко проверить. На предыдущем занятии мы сделали следующую веб-страницу. 

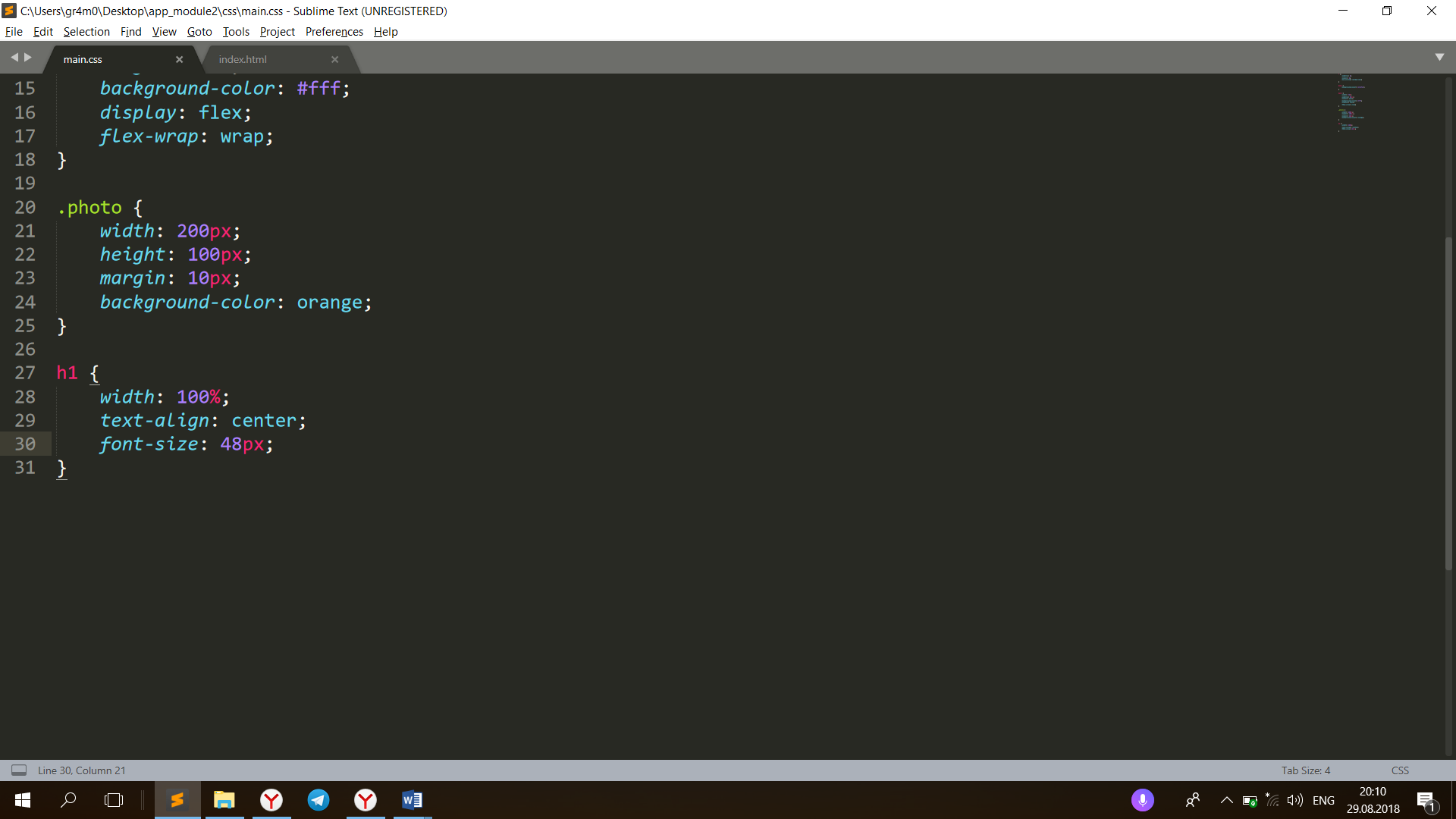
Каждый «див» занимает следующую строку.

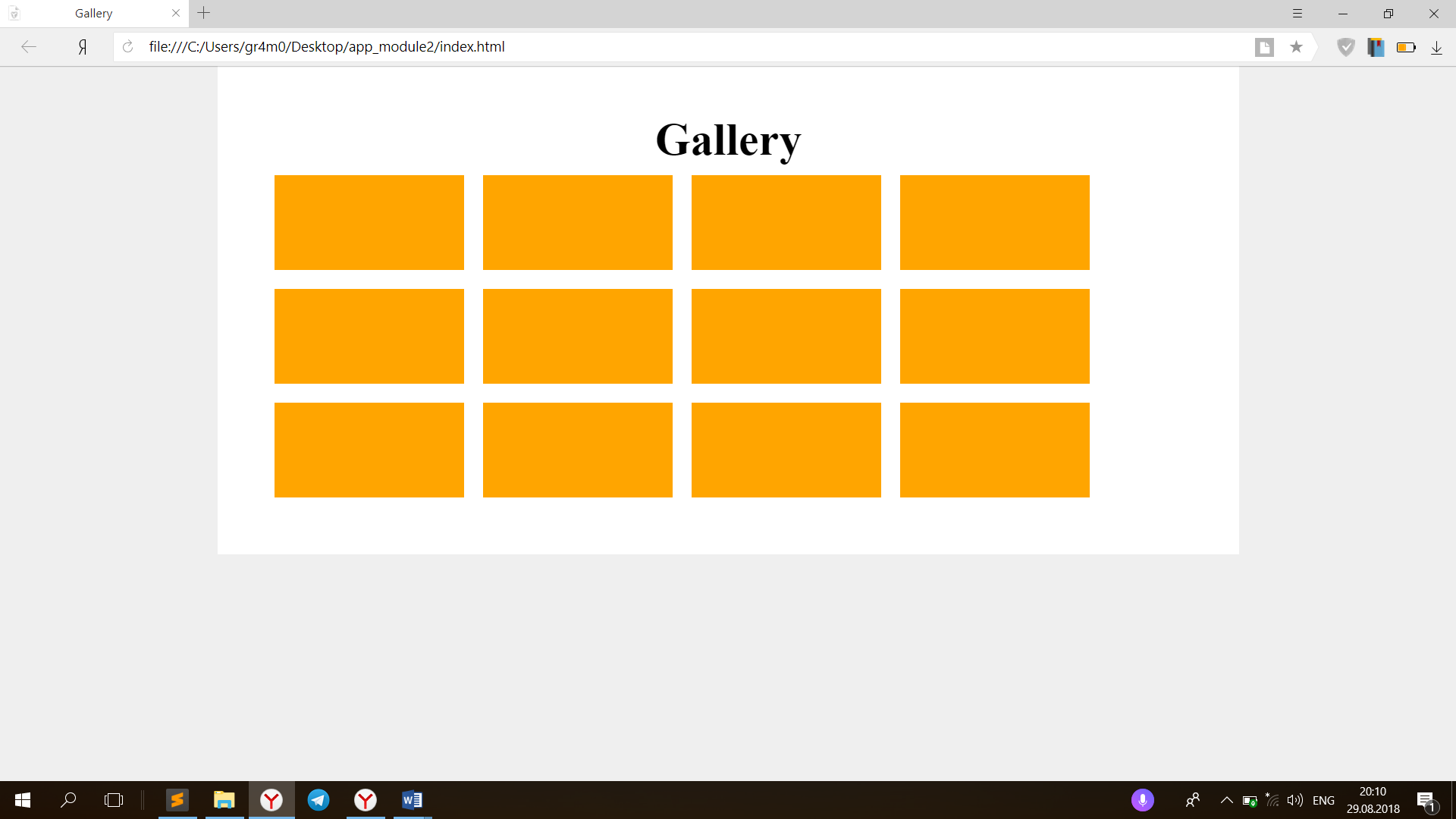
Открываем файл main.css в Sublime Text 3. Для тега main задаем flex-отображение. 

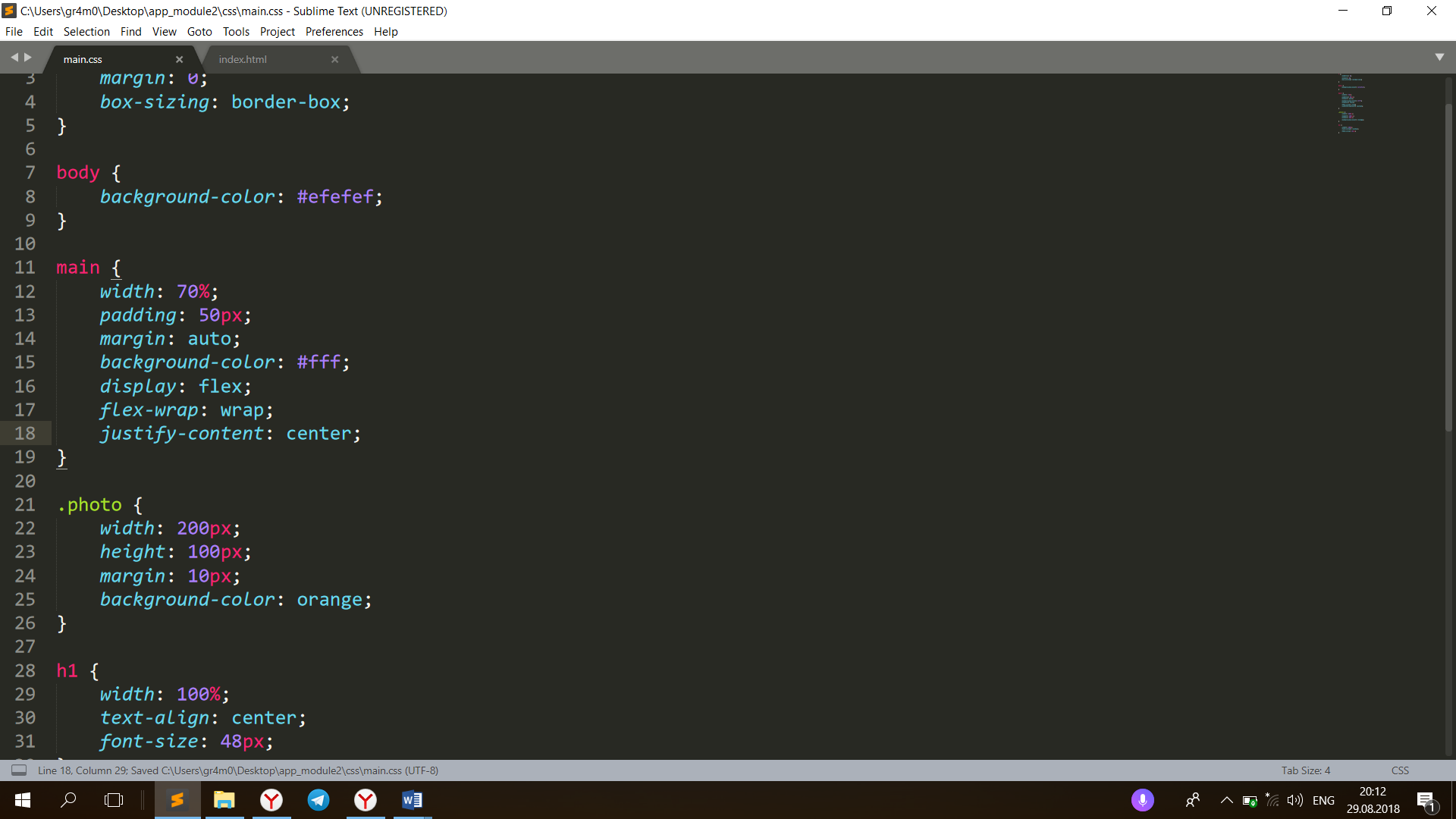


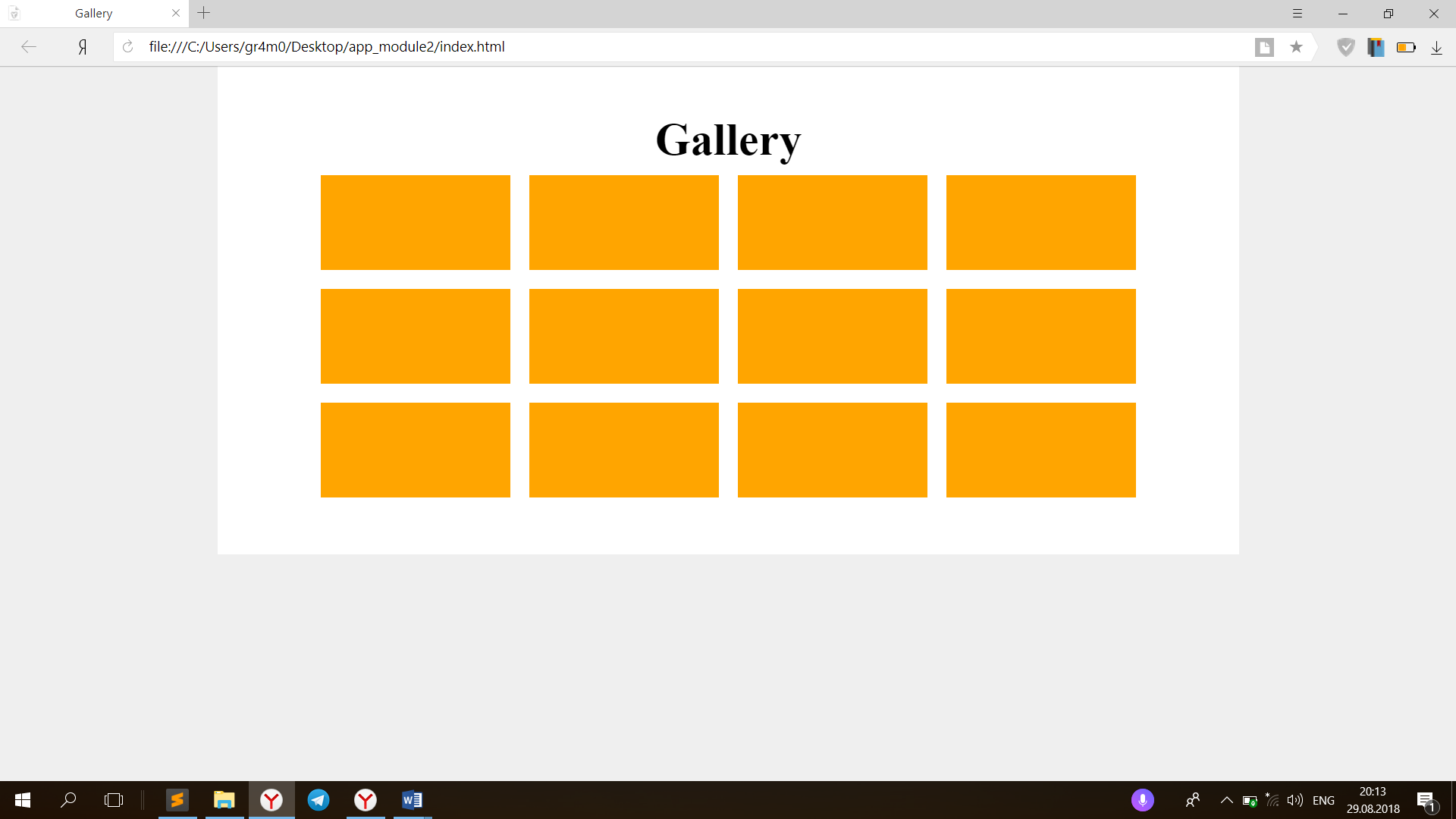
Значение flex выравнивает элементы в строку, независимо от их ширины. Чтобы элементы имели собственные размеры, нужно установить свойство flex-wrap: wrap; таким образом, мы получим предсказуемое поведение контейнера. 

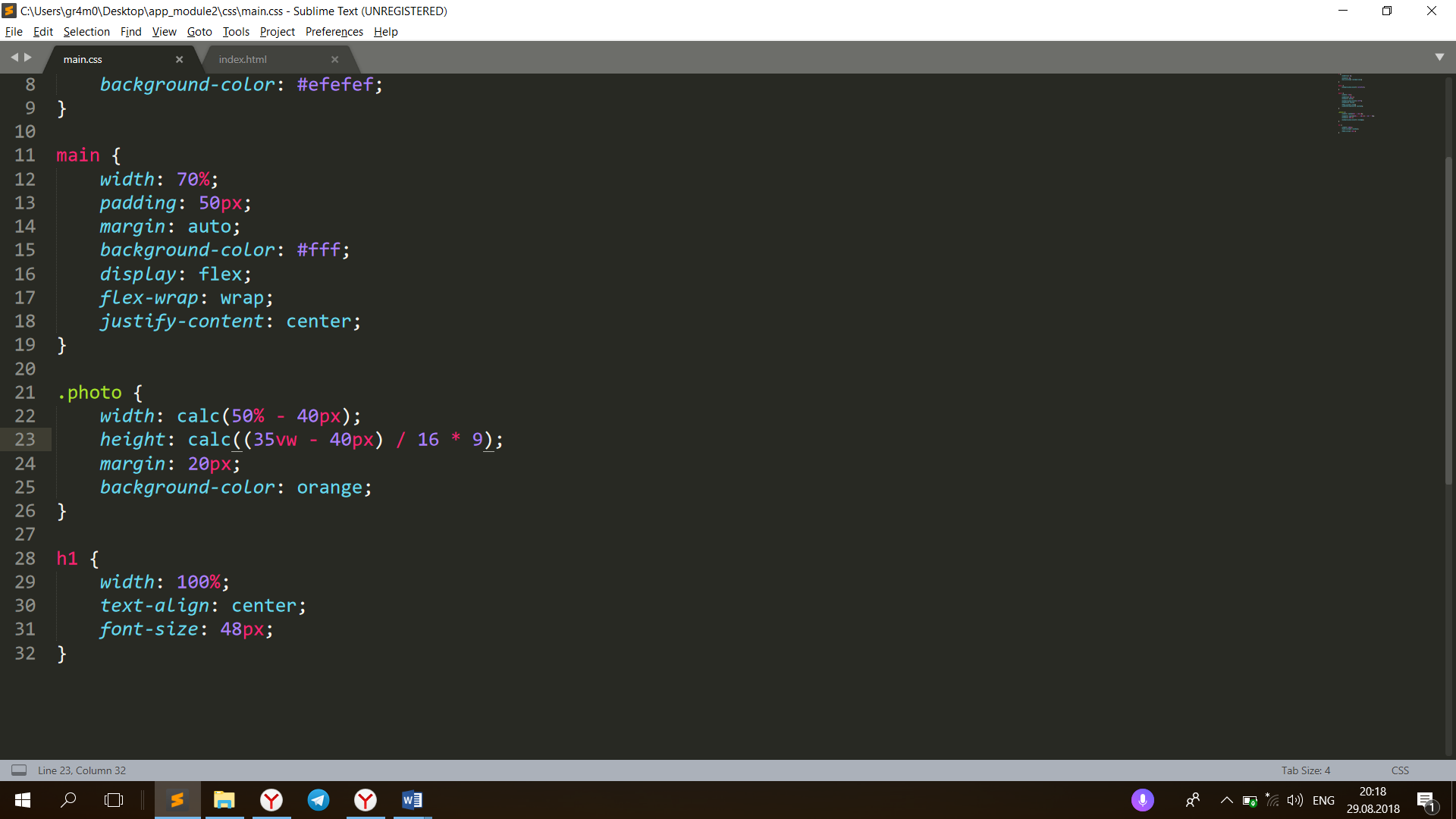


Установим правильное позиционирование для заголовка. Установим ему ширину 100% и выравнивание текста по горизонтали. 

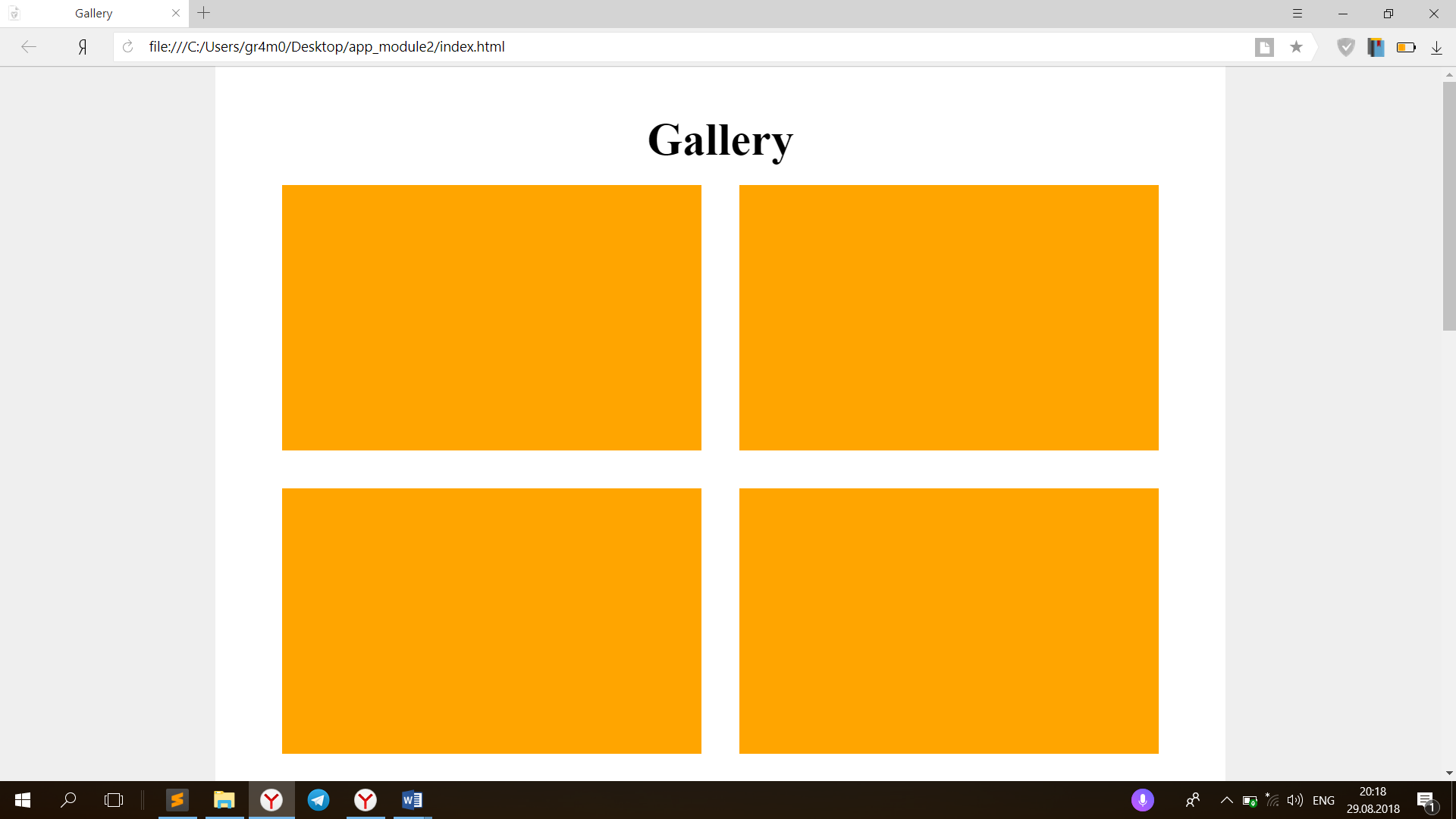


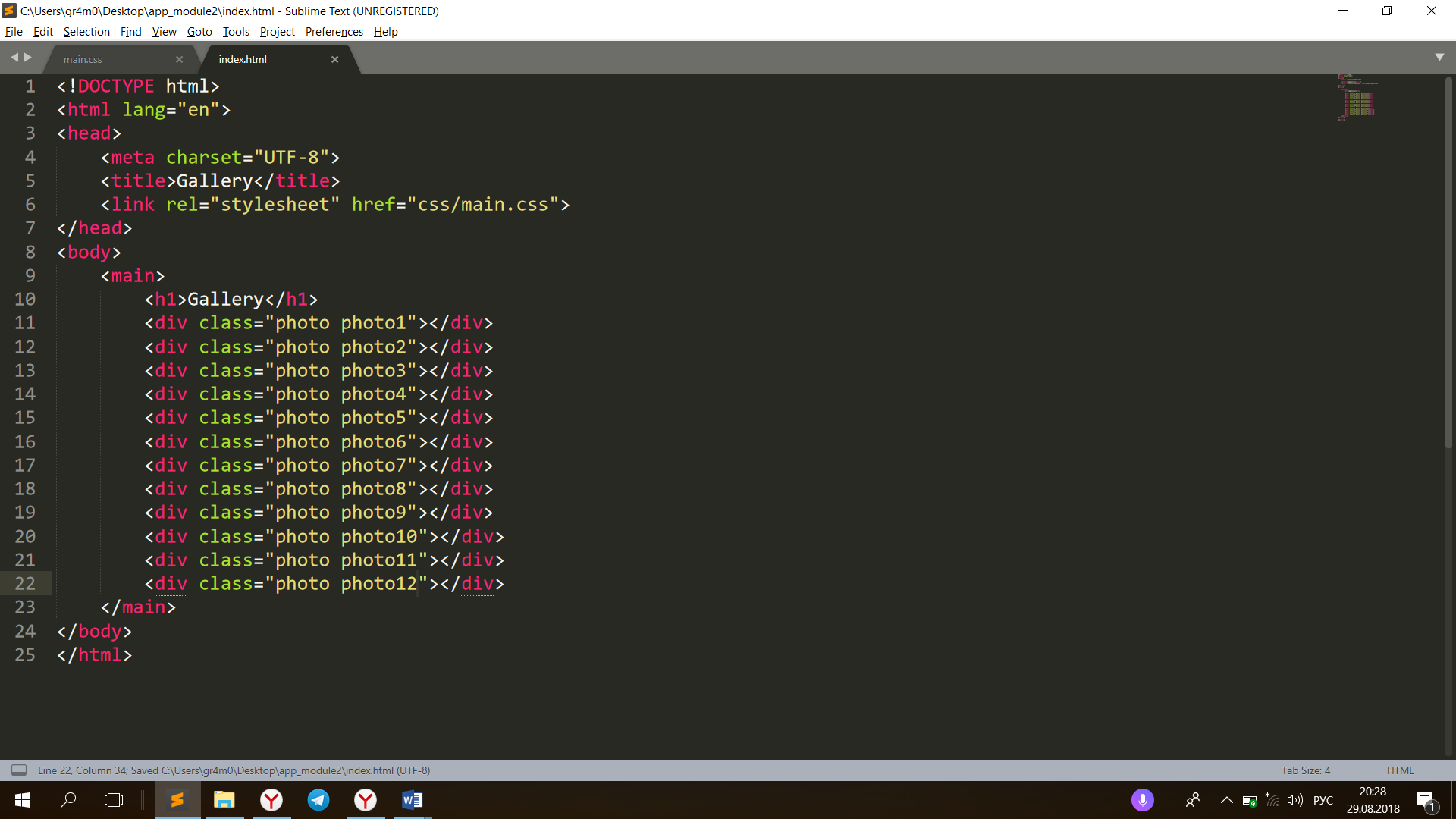
Далее, центрируем наши чудесные оранжевые кирпичики по горизонтали. Свойство justify-content работает только в связке с flex. 



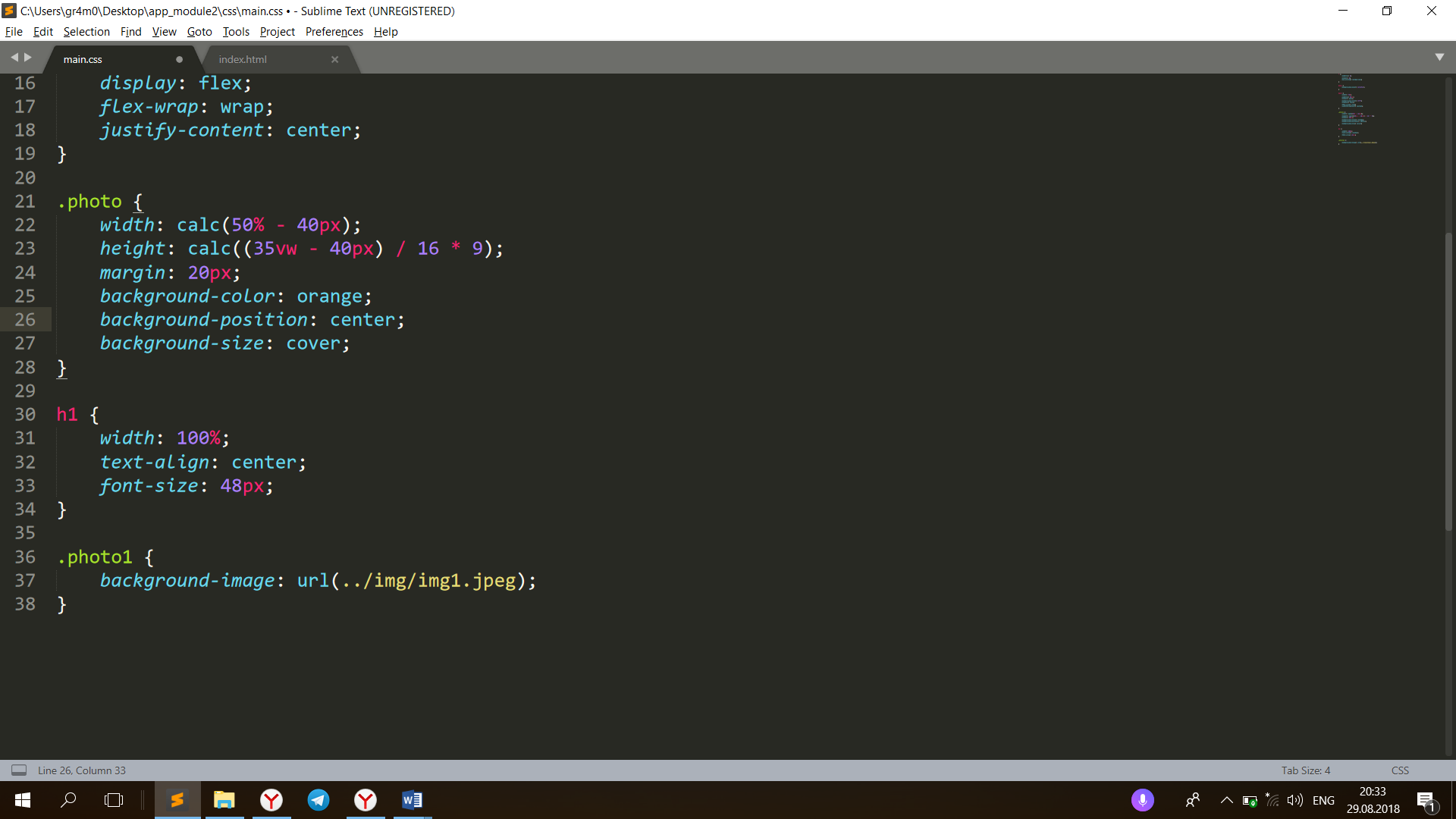
Изменим размеры наших плиточек так, чтобы они стояли в троке по два. 

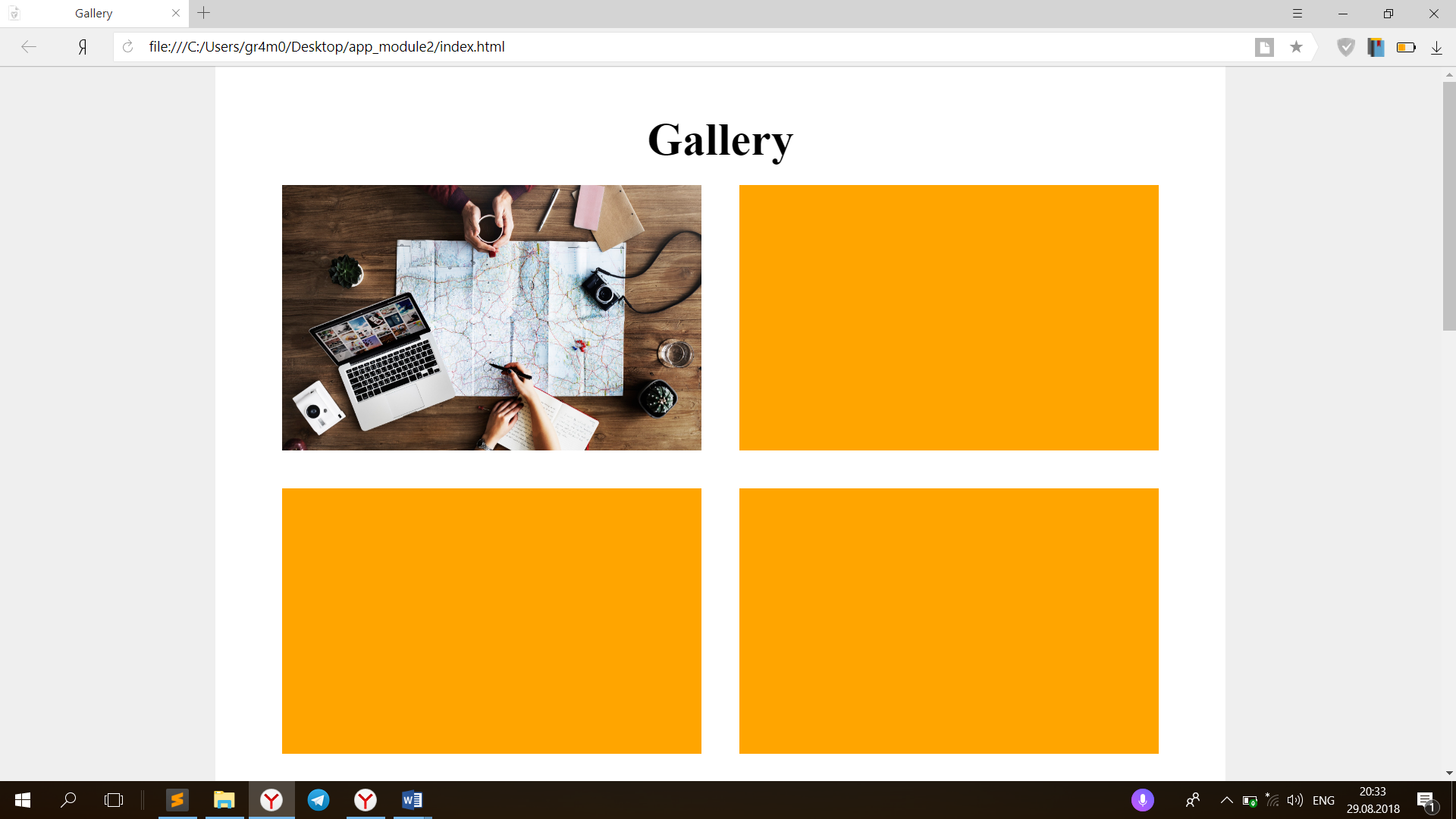
Здесь довольно необычные значения высоты и ширины. Рассмотрим более детально функцию calc(). Она позволяет нам высчитывать значения размеров элементов «на месте». Изначально мы не знаем, какая ширина окна браузера будет у нашего пользователя, поэтому эта функция предоставляет нам очень гибкий инструмент для вычисления размеров элемента.



Настало время убрать наши оранжевые кирпичики и поставить вместо них нормальные изображения. Для этого в файле index.html к каждому элементу с классом photo добавляем еще один класс, характеризующий его порядковый номер, например, photo1, photo2, photo3 и так далее. 

Установим свойство background-image для каждого элемента. Здесь необходимо правильно указать путь до изображений в нашем случае он будет следующим: url(../img/имя\_картинки.расширение). Также добавим еще два css-свойства для класса photo: background-position и background-size. Воспользуемся сокращенным методом ввода, для этого введем bgp и нажмем TAB, введем bgz и нажмем TAB.





Теперь нужно установить фоновое изображение для каждой из плиточек. Это будет домашним заданием.

На этом третье занятие подходит к концу.

