

# Flexbox

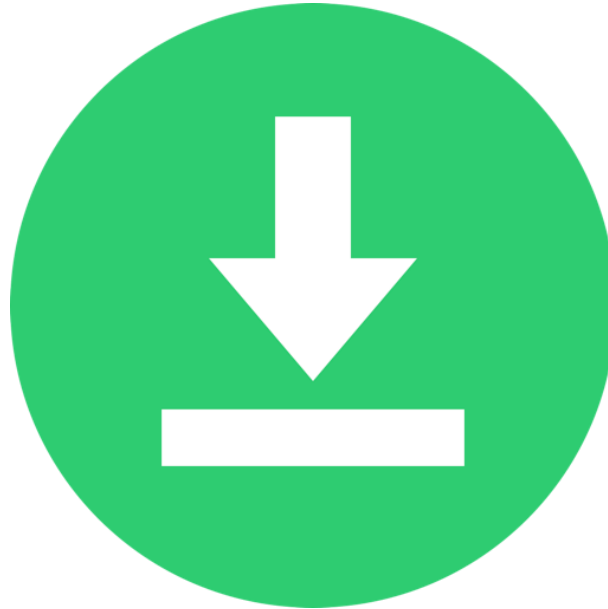
---

**WEB  
COURSE  
ORT DNIPRO**

---

**ORT**[DNIPRO.ORG/WEB](https://ORTDNIPRO.ORG/WEB)

# Скачайте шаблоны



Скачайте архив с шаблонами которые нам понадобятся:

<https://github.com/filebase-xyz/flexbox/archive/refs/heads/templates.zip>

# **1. Flexbox - размещение элементов вдоль оси**

A

B

C

D

E

F

G

H

I

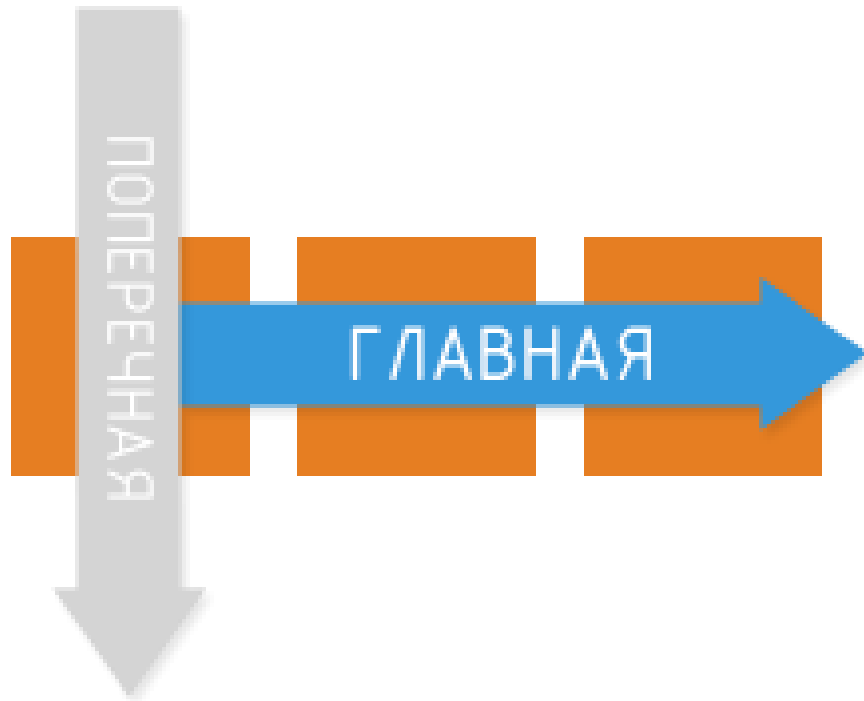
J

# Flexbox - герой эпохи до CSS Grid

Откройте проект из каталога (из архива):

[./flexbox-master/demo-template](#)

# Flexbox – управление размещением элементов

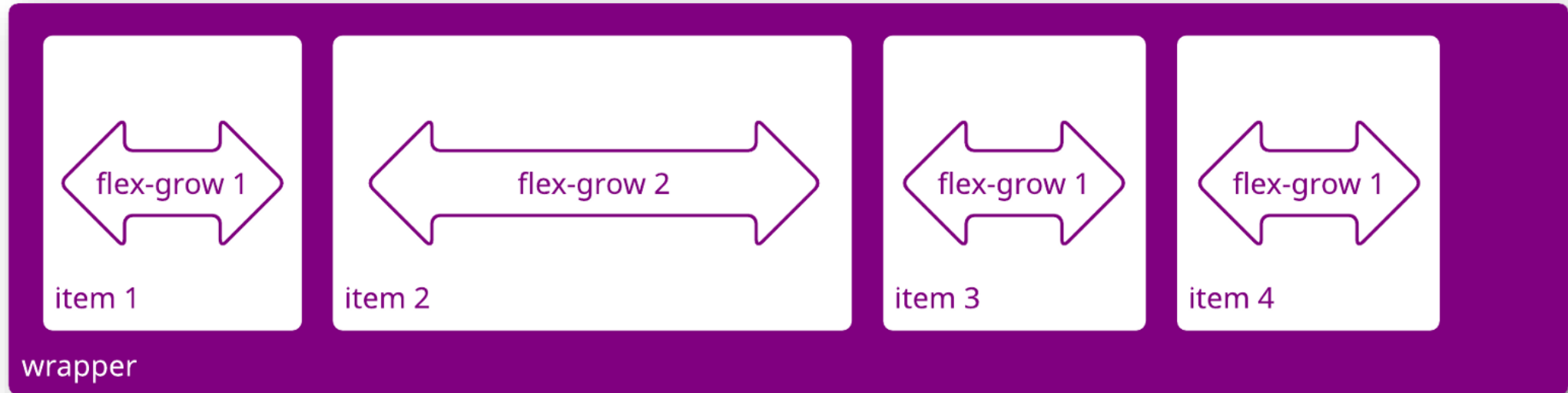


Идея **flexbox**'а проста: все элементы располагаются в ряд, для этого ряда есть главная и поперечная оси. Как будут элементы выравниваться по осям задаётся при помощи свойств **justify-content** и **align-items** (для главной и поперечной осей соответственно).

Включается режим **flexbox** при помощи свойства **display: flex**, а перенос элементов разрешается при помощи **flex-wrap: wrap**.

При помощи свойства **flex-direction** возможно указать направление размещения строка или колонка (**row**, **row-reverse**, **column**, **column-reverse**).

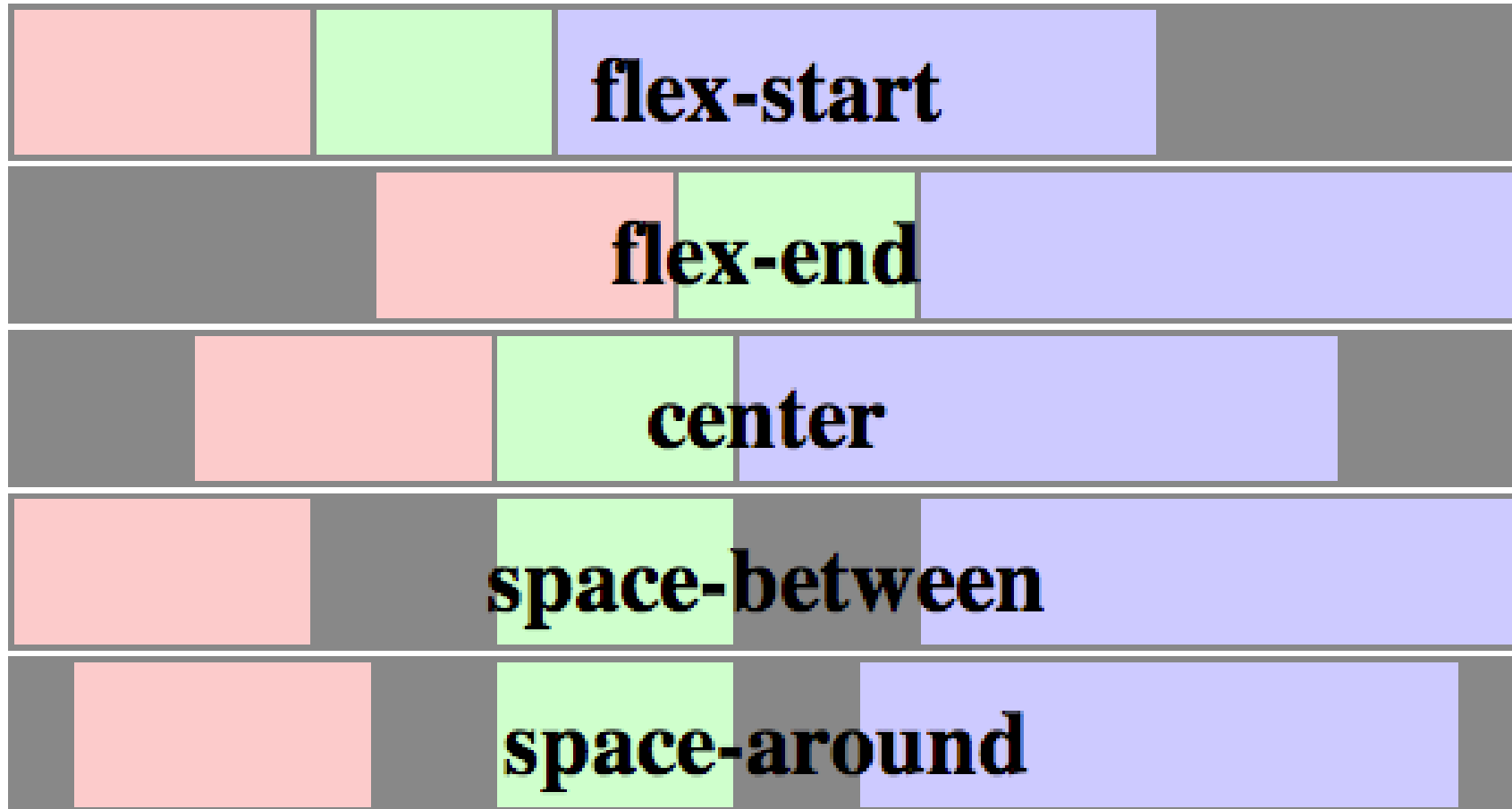
# Flexbox – управление размещением элементов



При помощи **flex-grow** (задаётся не для flex-контейнера, а для вложенных элементов) мы можем управлять тем как элементы будут занимать доступное пространство при его увеличении. Близкое по смыслу свойство **flex-shrink** отвечает за то как элемент будет сжиматься. Свойство **flex-basis** – позволяет задать базовый размер элемента вдоль главной оси. Также все три значения задаются при помощи общего свойства с названием flex (например: **flex: 1 0 auto**; в котором указываются соответственно **flex-grow**, **flex-shrink**, **flex-basis**).

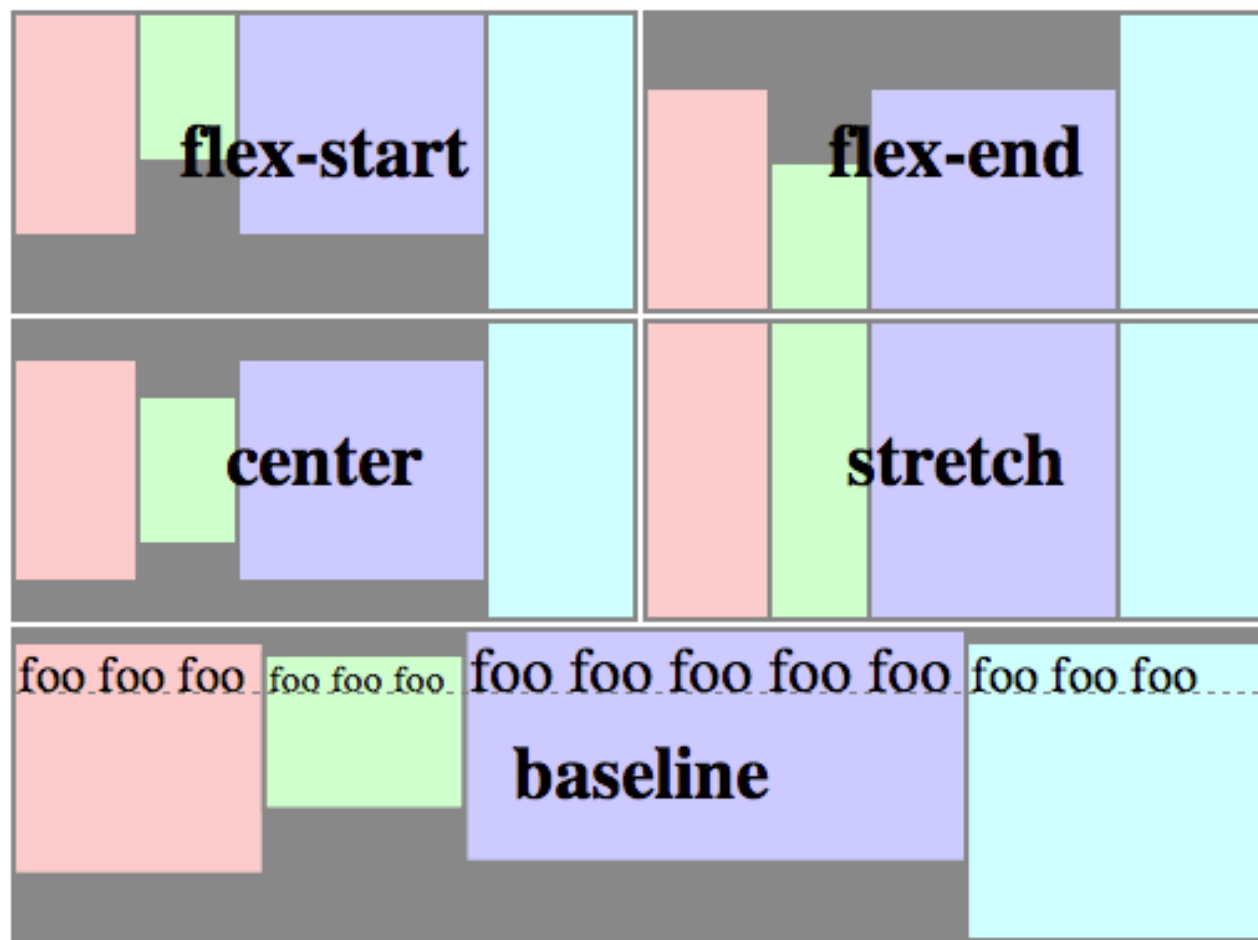
<https://webref.ru/css/flex>

## Flexbox – управление размещением элементов



**justify-content:** размещение элементов относительно главной оси.

# Flexbox – управление размещением элементов



**align-items:** размещение элементов относительно поперечной оси.




## **2. Построение шаблона страницы и компонентов**

## Lorem ipsum dolor sit amet consectetur adipisicing elit.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Itaque fuga incidunt, illo ducimus ea dolor quia molestias porro ratione quisquam voluptates placeat unde ex assumenda provident necessitatibus obcaecati molestiae nesciunt. Explicabo quisquam, nam dolorem voluptatem, architecto adipisci doloribus fuga error id nostrum ab nulla ipsam. Atque eum doloribus, at dolore commodi aliquid quas. Amet illo pariatur quod? Accusantium doloremque porro consequuntur soluta. Sit dolor deleniti nesciunt mollitia corporis pariatur veniam molestias magnam dolores, repudiandae vitae aliquam fuga vero perspiciatis nostrum et accusantium, ad labore minima. Ex expedita alias nulla tenetur similique. Quisquam, eveniet inventore quis quas rem eaque molestiae quidem autem porro non nihil qui cumque odit enim eum sunt fugit quo!


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart


Apple iPhone

799\$

 Add to Cart

Apple iPhone

799\$

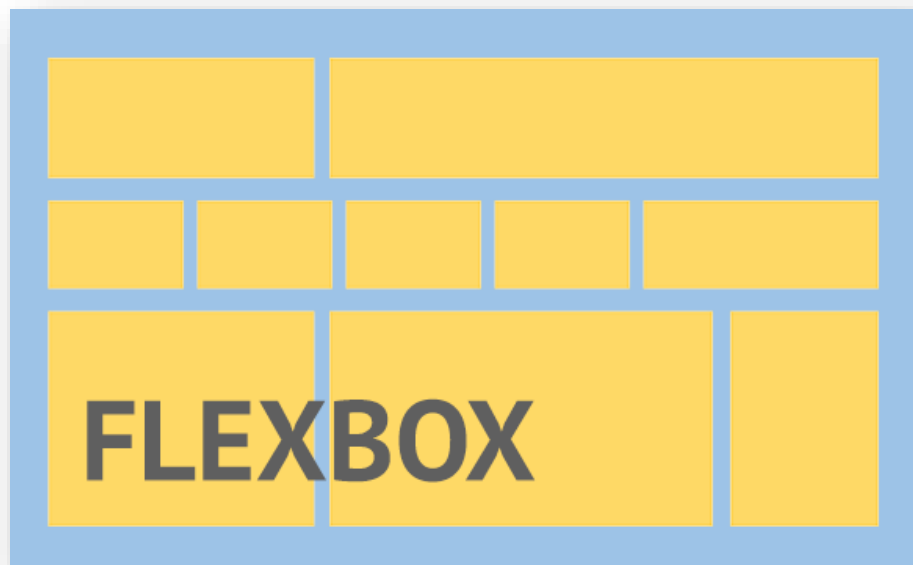
 Add to Cart

**Flexbox** может помочь и в формировании как шаблона страницы в целом, так и отдельных компонентов этой страницы (*одно другому не мешает*)

Откройте проект из каталога (из архива):

[./flexbox-master/full-page-template](#)

**Будет полезным**



## Как работает Flexbox

Вам будет полезным узнать детали работы Flexbox'a



<https://tproger.ru/translations/how-css-flexbox-works/>

<https://habr.com/ru/company/ruvds/blog/515298/>

# Домашнее задание

# Пройти игру по Flexbox!

## FLEXBOX FROGGY

Допоможіть жабеняттам потрапити на свої вертикальні листки латаття, використав `flex-direction`. Ця CSS властивість визначає напрямок елементів в контейнері та може набувати наступних значень:

- `row`: елементи розташовані так само, як напрямок тексту.
- `row-reverse`: елементи розташовані протилежно напрямку тексту.
- `column`: елементи розташовані з гори до низу.
- `column-reverse`: елементи розташовані з низу до гори.

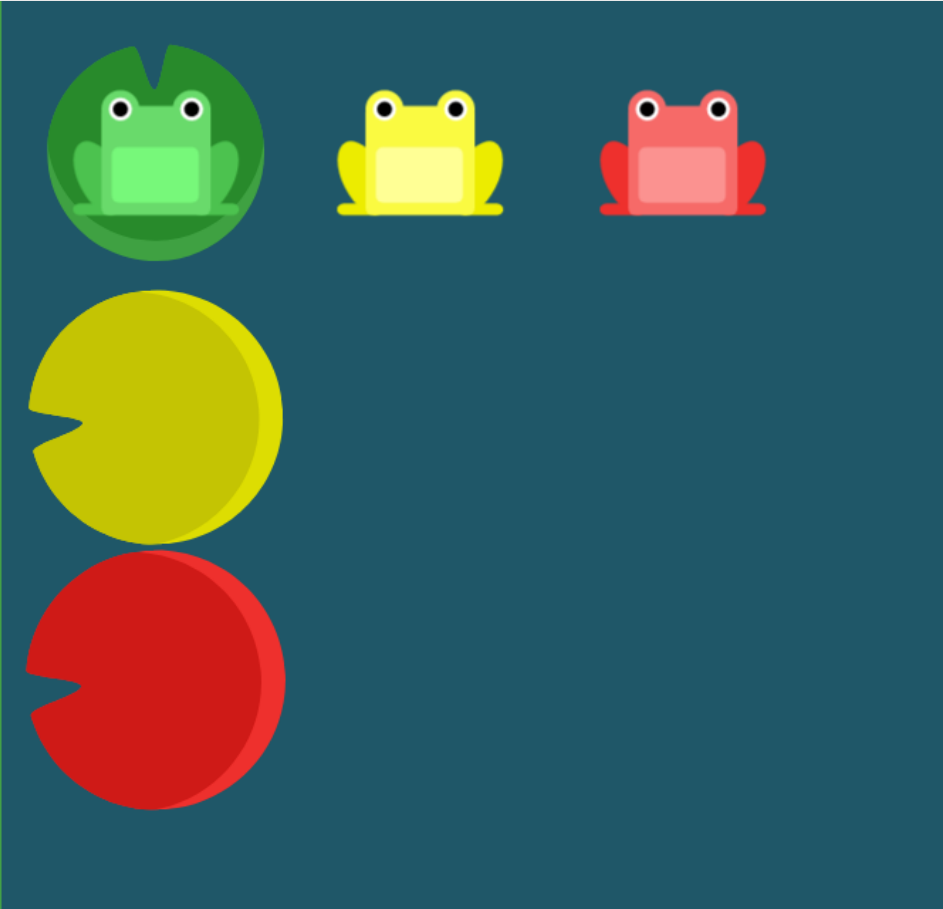
```
1 #pond {
2   display: flex;
3   |
4 }
5
6
7
8
9
10
```

Наступний

Гру Flexbox Froggy створено [Codepip](#) • [GitHub](#) • [Twitter](#) • [Settings](#)

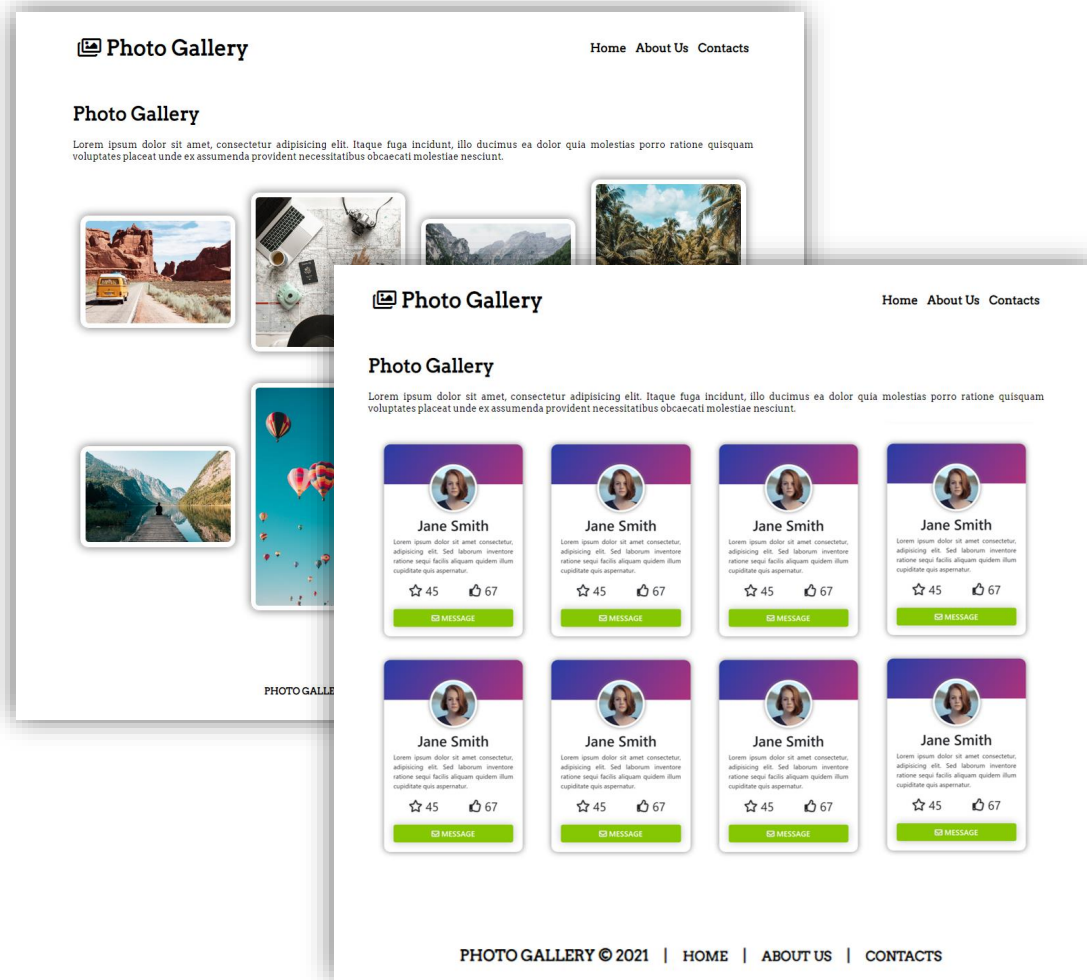
Want to learn CSS grid? Play [Grid Garden](#).

Рівень 9 з 24



<https://flexboxfroggy.com/>

# Домашнее задание | в 2 этапа



Создайте адаптивный «контейнер» в **70vw** от ширины видимой области браузера, но в рамках от **576px** до **1200px**. Внутри него разместите контент как на макете.

## Photo Gallery

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Itaque fuga incidunt, illo ducimus ea dolor quia molestias porro ratione quisquam voluptates placeat unde ex assumenda provident necessitatibus obcaecati molestiae nesciunt.



## Домашнее задание

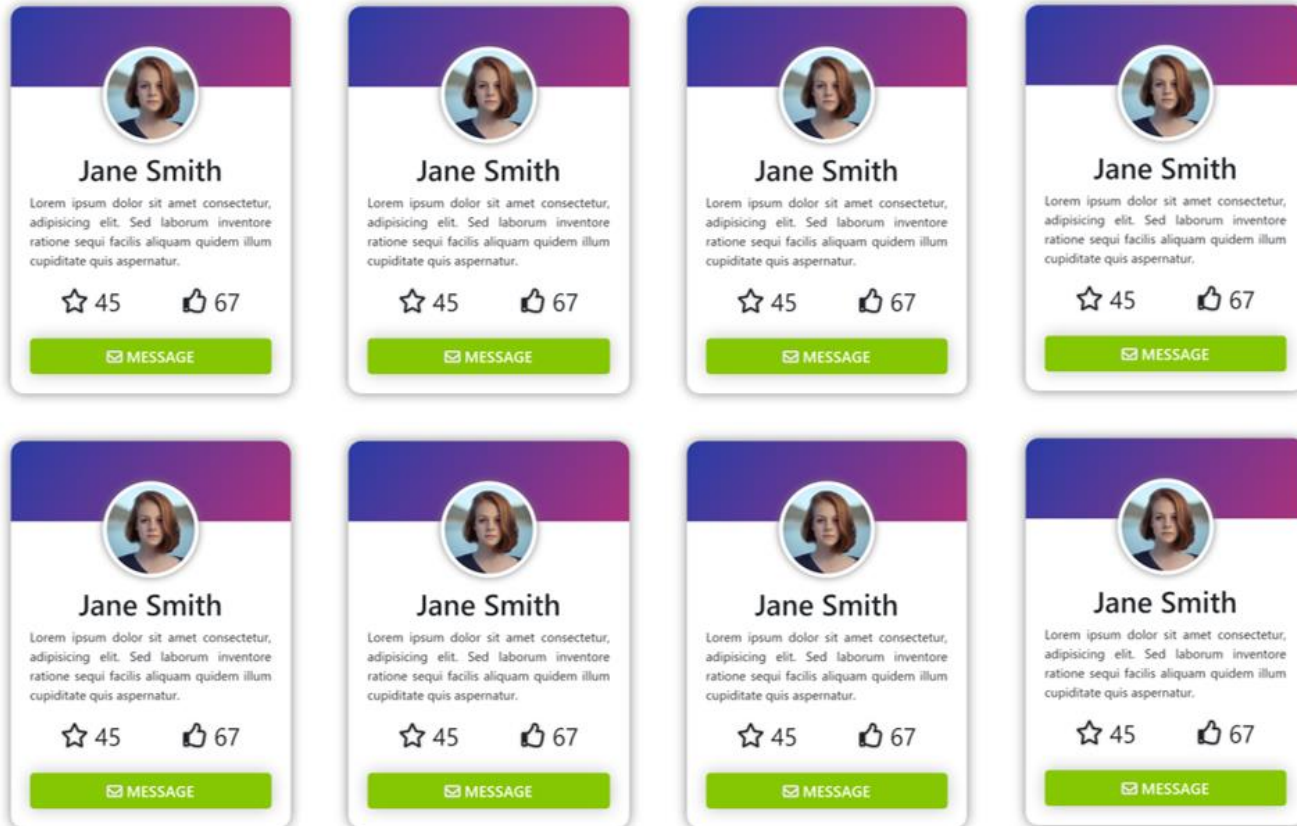
Реализуйте страницу фото галереи (**4 фото в ряд, всего 25 фоток**, с зазором между фото), **Flexbox** вам в этом поможет.

Этап #1



## Photo Gallery

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Itaque fuga incidunt, illo ducimus ea dolor quia molestias porro ratione quisquam voluptates placeat unde ex assumenda provident necessitatibus obcaecati molestiae nesciunt.



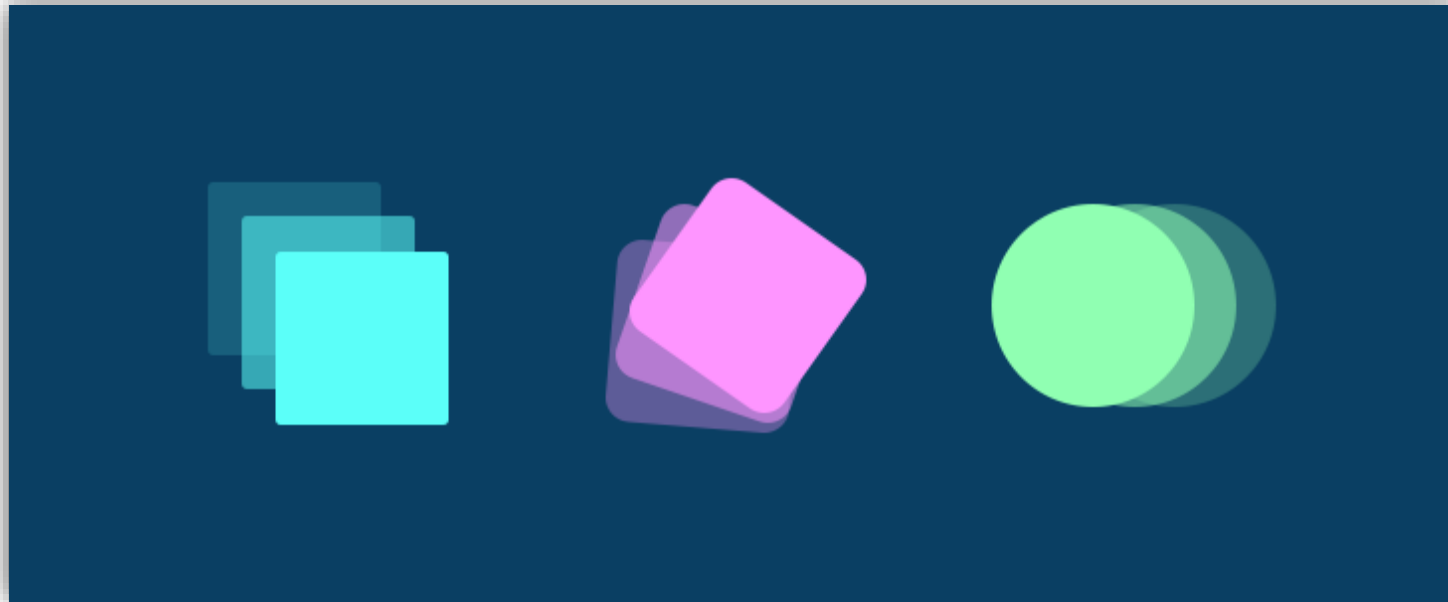
# Домашнее задание

Вместо фотографий,  
разместите на странице  
экземпляры компонента из  
предыдущего ДЗ.

Этап #2

**К следующему  
занятию...**

# CSS transition – анимация переходов



Предварительные знания – лучший помощник в обучении, поэтому к следующему занятию жду, что **посмотрите небольшой ролик о анимации переходов.**

<https://youtu.be/Nloq6uzF8RQ>