

FLEXBOX FROGGY

Уровень 1 из 24

Добро пожаловать в Flexbox Froggy. Игра, в которой тебе нужно помочь лягушонку Фроги и его друзьям, напиконтейнер CSS код! Направь этого лягушонка на лилию справа, используя свойство `justify-content`, которое выравнивает элементы горизонтально и принимает следующие значения:

- `flex-start`: элементы выравниваются по левой стороне контейнера.
- `flex-end`: элементы выравниваются по правой стороне контейнера.
- `center`: элементы выравниваются по центру контейнера.
- `space-between`: элементы отображаются с одинаковыми отступами между ними.
- `space-around`: элементы отображаются с одинаковыми отступами вокруг них.

Например, `justify-content: flex-end;` сдвинет лягушонка вправо.

```
1 #pond {
2   display: flex;
3   justify-content: flex-end
4 }
5
6
7
8
9
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 2 из 24

Используй `justify-content` ещё раз, чтобы помочь этим лягушатам попасть на их лилии. Помни, что это свойство CSS выравнивает элементы горизонтально и принимает следующие значения:

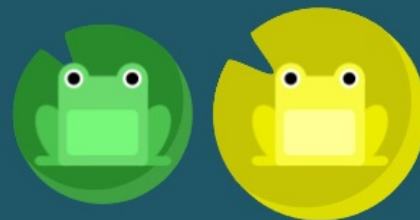
- `flex-start`: элементы выравниваются по левой стороне контейнера.
- `flex-end`: элементы выравниваются по правой стороне контейнера.
- `center`: элементы выравниваются по центру контейнера.
- `space-between`: элементы отображаются с одинаковыми отступами между ними.
- `space-around`: элементы отображаются с одинаковыми отступами вокруг них.

```
1 #pond {  
2   display: flex;  
3   justify-content: center;  
4 }
```

Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).

FLEXBOX FROGGY

Уровень 3 из 24

Помоги всем трём лягушатам найти их лилии, просто используя `justify-content`. В этот раз у лилий много пространства вокруг.

Если ты чувствуешь, что забыл возможные значения свойства, ты можешь навести курсор на название свойства, чтобы посмотреть их. Попробуй навести курсор на `justify-content`.

```
1 #pond {  
2   display: flex;  
3   justify-content: space-around;  
4 }
```

Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).

FLEXBOX FROGGY

Уровень 4 из 24

Теперь лилии по краям уплыли к берегам, увеличив пространство между ними. Используй `justify-content`. В этот раз у лилий одинаковое расстояние между ними.

```
1 #pond {  
2   display: flex;  
3   justify-content: space-between;  
4 }  
5  
6  
7  
8  
9  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 5 из 24

Теперь используйте **align-items**, чтобы помочь лягушатам добраться до нижней части пруда. Это свойство CSS выравнивает элементы вертикально и принимает следующие значения:

- **flex-start**: элементы выравниваются по верхнему краю контейнера.
- **flex-end**: элементы выравниваются по нижнему краю контейнера.
- **center**: элементы выравниваются вертикально по центру контейнера.
- **baseline**: элементы отображаются на базовой линии контейнера.
- **stretch**: элементы растягиваются, чтобы заполнить контейнер.

```
1 #pond {  
2   display: flex;  
3   align-items: flex-end;  
4 }
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 6 из 24

Направь лягушонка в центр пруда, используя `justify-content` и `align-items` вместе.

```
1 #pond {  
2   display: flex;  
3   justify-content: center;  
4   align-items: center;  
5 }
```

Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 7 из 24

Лягушатам снова нужно пересечь пруд. В этот раз к лилиям, с достаточно большим пространством вокруг них. Используй комбинацию `justify-content` и `align-items`.

```
1 #pond {  
2   display: flex;  
3   justify-content: space-around;  
4   align-items: flex-end;  
5 }
```

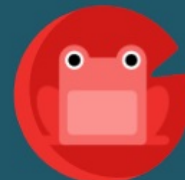
Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 8 из 24

Лягушатам нужно выстроиться в том же порядке, что и лилии, используя `flex-direction`. Это свойство CSS задает направление, в котором будут расположены элементы в контейнере, и принимает следующие значения:

- `row`: элементы размещаются по направлению текста.
- `row-reverse`: элементы отображаются в обратном порядке к направлению текста.
- `column`: элементы располагаются сверху вниз.
- `column-reverse`: элементы располагаются снизу вверх.

```
1 #pond {  
2   display: flex;  
3   flex-direction: row-reverse;  
4 }
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Помоги лягушатам расположиться на своих лилиях, используя `flex-direction`. Это свойство CSS задает направление, в котором располагаются элементы в контейнере, и принимает следующие значения:

- `row`: элементы размещаются по направлению текста.
- `row-reverse`: элементы отображаются в обратном порядке к направлению текста.
- `column`: элементы располагаются сверху вниз.
- `column-reverse`: элементы располагаются снизу вверх.

```
1 #pond {
2   display: flex;
3   flex-direction: column;
4 }
```

Следующий

SPONSOR Frontend Masters – Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 10 из 24

Помоги лягушатам попасть на свои лилии. Хотя и кажется, что они почти на своих местах, всё же придётся применить и `flex-direction`, и `justify-content`, чтобы поместить их на свои лилии.

Заметь, что когда ты устанавливаешь направление в обратном порядке для ряда или колонки, начало (start) и конец (end) тоже меняются местами.

```
1 #pond {  
2   display: flex;  
3   flex-direction: row-reverse;  
4   justify-content: flex-end;  
5 }
```

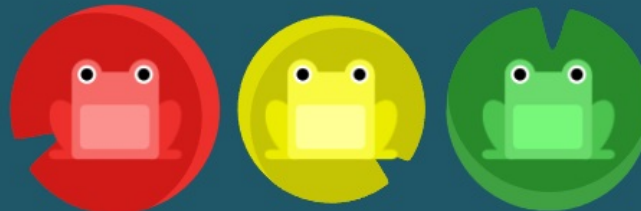
Следующий

SPONSOR

Frontend Masters – Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 11 из 24

Помоги лягушатам найти их лилии с помощью `flex-direction` и `justify-content`.

Заметь, когда в качестве направления выбрана колонка, то `justify-content` влияет на вертикальное выравнивание, а `align-items` — на горизонтальное.

```
1 #pond {  
2   display: flex;  
3   flex-direction: column;  
4   justify-content: flex-end;  
5 }
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 12 из 24

Помоги лягушатам найти их лилии с помощью `flex-direction` и `justify-content`.

```
1 #pond {  
2   display: flex;  
3   justify-content: space-between;  
4   flex-direction: column-reverse;  
5 }
```

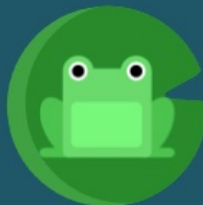
Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 13 из 24

Помоги лягушатам найти их лилии с помощью `flex-direction`, `justify-content` и `align-items`.

```
1 #pond {  
2   display: flex;  
3   flex-direction: row-reverse;  
4   align-items: flex-end;  
5   justify-content: center;  
6 }
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 14 из 24

Иногда изменения порядка отображения элементов в контейнере недостаточно. В таких случаях мы можем применить свойство `order` для конкретных элементов. По умолчанию, значение этого свойства у элементов равно 0, но мы можем задать положительное или отрицательное целое число этому свойству.

Используя свойство `order`, чтобы разместить лягушат на своих лилиях.

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

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 15 из 24

Используй свойство **order**, чтобы отправить красного лягушонка на его лилию.

```
1 #pond {  
2   display: flex;  
3 }  
4  
5 .red {  
6   order: -1;  
7 }  
8  
9  
10
```

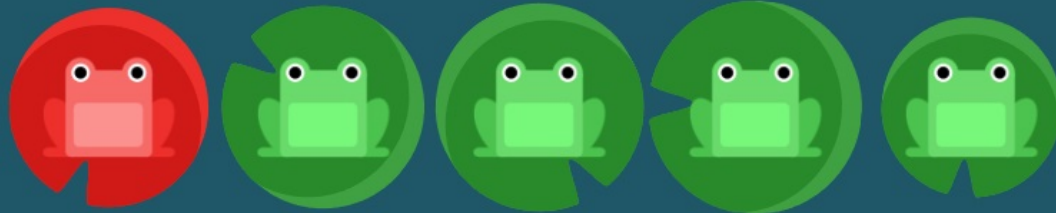
Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 16 из 24

Ещё одно свойство, которое ты можешь применить к определенному элементу – это `align-self`. Это свойство принимает те же значения, что и `align-items`.

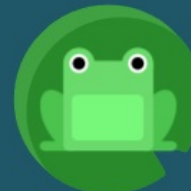
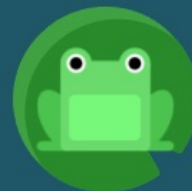
```
1 #pond {  
2   display: flex;  
3   align-items: flex-start;  
4 }  
5  
6 .yellow {  
7   align-self: flex-end;  
8 }  
9  
10
```

Следующий

SPONSOR Frontend Masters – Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 17 из 24

Используйте `order` и `align-self` вместе, чтобы помочь лягушатам добраться к своим целям.

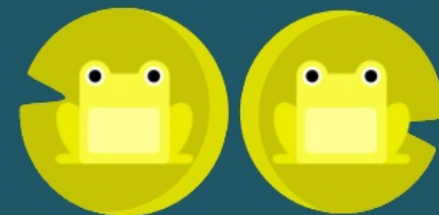
```
1 #pond {  
2   display: flex;  
3   align-items: flex-start;  
4 }  
5  
6 .yellow {  
7   order: 1;  
8   align-self: flex-end;  
9 }  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 18 из 24

О нет! Лягушат сплющило на одном ряду лилий. Раздвинь их с помощью свойства `flex-wrap`, которое принимает следующие значения:

- `nowrap`: размеры элементов устанавливаются автоматически, чтобы они поместились в один ряд.
- `wrap`: элементы автоматически переносятся на новую строку.
- `wrap-reverse`: элементы автоматически переносятся на новую строку, но строки расположены в обратном порядке.

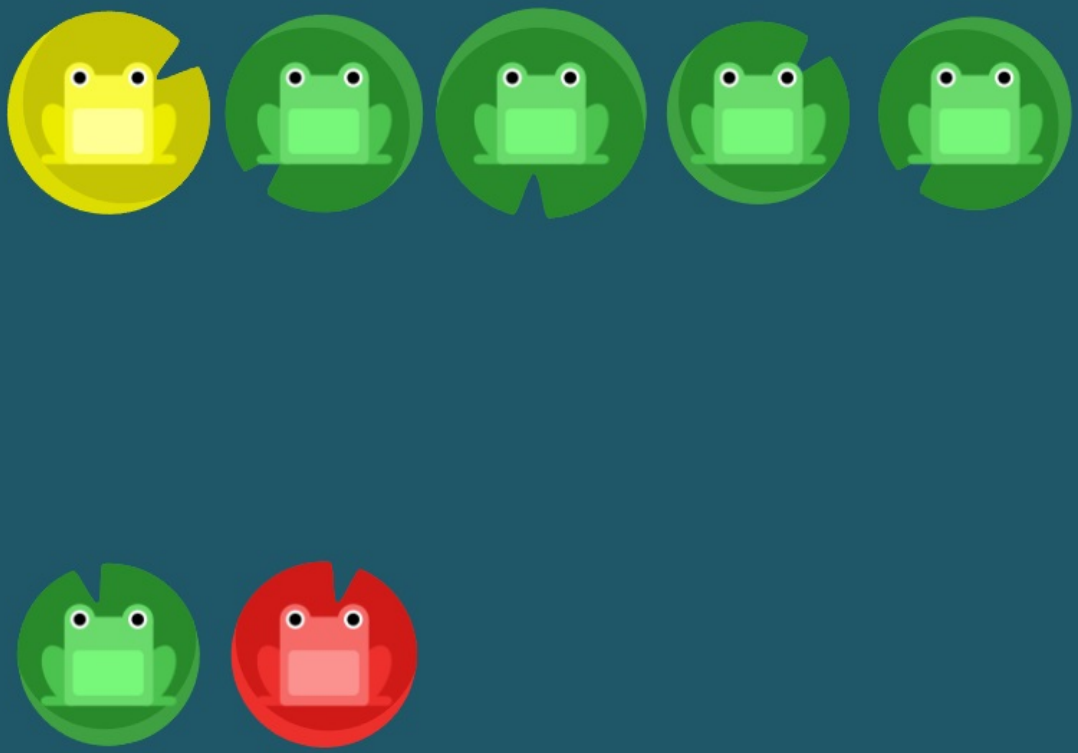
```
1 #pond {
2   display: flex;
3   flex-wrap: wrap;
4 }
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 19 из 24

Помоги этой армии лягушат выстроиться в три колонки с помощью комбинации `flex-direction` и `flex-wrap`.

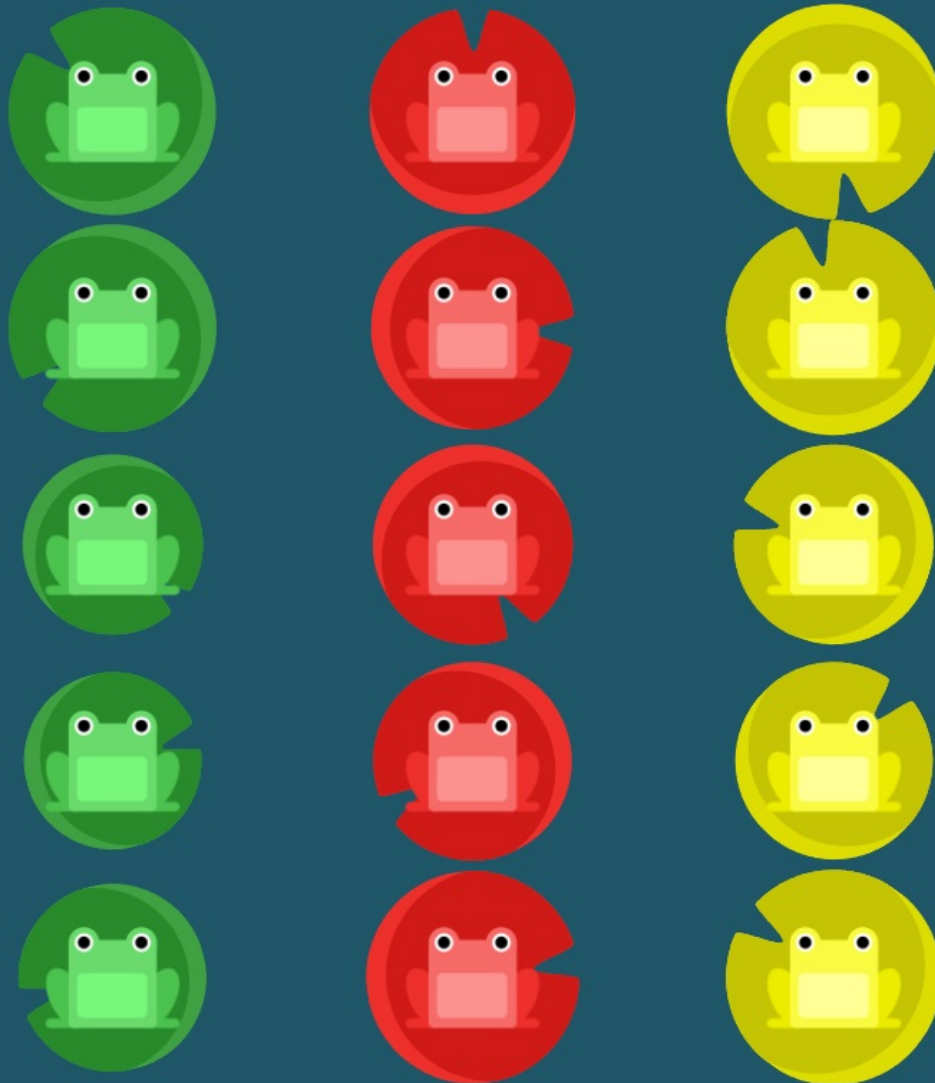
```
1 #pond {  
2   display: flex;  
3   justify-content: space-between;  
4   flex-direction: column;  
5 }  
6  
7  
8  
9  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 20 из 24

Два свойства `flex-direction` и `flex-wrap` используются так часто вместе, что было создано свойство `flex-flow` для их комбинирования. Это свойство принимает их значения, разделённые пробелом.

Например, ты можешь использовать `flex-flow: row wrap`, чтобы элементы располагались в ряд и автоматически переносились на новую строку.

Попробуй использовать `flex-flow`, чтобы повторить предыдущий уровень.

```
1 #pond {  
2   display: flex;  
3   justify-content: space-between;  
4 }
```

Следующий

SPONSOR

Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 21 из 24

Лягушат раскидало по всему пруду, но лилии сгруппированы в верхней части. Ты можешь использовать **align-content**, чтобы указать, как несколько рядов должны отделяться друг от друга. Данное свойство принимает следующие значения:

- **flex-start**: ряды группируются в верхней части контейнера.
- **flex-end**: ряды группируются в нижней части контейнера.
- **center**: ряды группируются вертикально по центру контейнера.
- **space-between**: ряды отображаются с одинаковыми расстояниями между ними.
- **space-around**: ряды отображаются с одинаковыми расстояниями вокруг них.
- **stretch**: ряды растягиваются, чтобы заполнить контейнер равномерно.

Это может запутать, но **align-content** отвечает за расстояние между рядами, в то время как **align-items** отвечает за то, как элементы в целом будут выровнены в контейнере. Когда только один ряд, **align-content** ни на что не влияет.

```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap;  
4   align-content: flex-start;  
5 }  
6  
7  
8  
9  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).

FLEXBOX FROGGY

Уровень 22 из 24

Теперь течение сгруппировало лилии в нижней части. Используйте `align-content`, чтобы направить лягушат туда.

```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap;  
4   align-content: flex-end;  
5 }  
6  
7  
8  
9  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 23 из 24

Лягушата были на вечеринке, но уже пора возвращаться домой. Используй комбинацию `flex-direction` и `align-content`, чтобы отправить их к своим лилиям.

```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap;  
4   align-items: center;  
5   align-content: center;  
6 }  
7  
8  
9  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



FLEXBOX FROGGY

Уровень 24 из 24

Доставь лягушат домой в последний раз, используя свойства CSS, которые ты выучил:

- `justify-content`
- `align-items`
- `flex-direction`
- `order`
- `align-self`
- `flex-wrap`
- `flex-flow`
- `align-content`

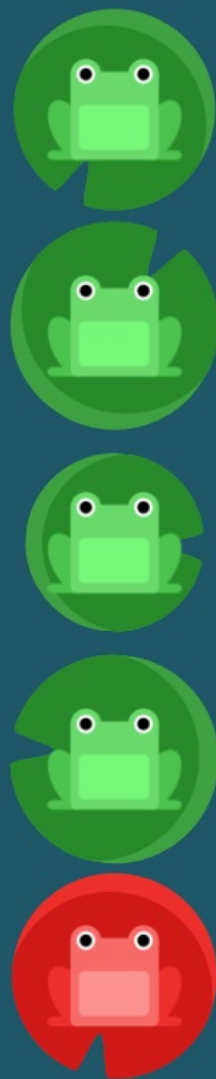
```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap-reverse;  
4   align-content: space-between;  
5   justify-content: center;  
6   align-items: flex-start;  
7 }  
8  
9  
10
```

Следующий

SPONSOR Frontend Masters — Check out the new Complete Intro to React course and learn to build real-world apps.

Игра Flexbox Froggy создана [Codepip](#) • [GitHub](#) • [Twitter](#) • [Настройки](#)

Хотите изучить CSS grid? Сыграйте в [Grid Garden](#).



1 justify-content: flex-end	15 order: -1;	23 flex-direction: column-reverse;
2 justify-content: center;	16 align-self: flex-end;	flex-wrap: wrap;
3 justify-content: space-around;	17 order: 1;	align-items: center;
4 justify-content: space-between;	align-self: flex-end;	align-content: center;
5 align-items: flex-end;	18 flex-wrap: wrap;	24 flex-direction: column-reverse;
6 justify-content: center;	19 flex-direction: row;	flex-wrap: wrap-reverse;
align-items: center;	flex-wrap: wrap;	align-content: space-between;
7 justify-content: space-around;	justify-content: space-between;	justify-content: center;
align-items: flex-end;	flex-direction: column;	align-items: flex-start;
8 flex-direction: row-reverse;	20 flex-direction: row;	
9 flex-direction: column;	flex-wrap: wrap;	
10 flex-direction: row-reverse;	justify-content: space-between;	
justify-content: flex-end;	flex-direction: column;	
11 flex-direction: column;	21 flex-direction: row;	
justify-content: flex-end;	flex-wrap: wrap;	
align-items: flex-start;	align-items: flex-start;	
12 justify-content: space-between;	align-content: flex-start;	
flex-direction: column-reverse;	22 flex-direction: row;	
13 flex-direction: row-reverse;	flex-wrap: wrap;	
align-items: flex-end;	align-items: flex-start;	
justify-content: center;	align-content: flex-end;	
14 order: 1;		