

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Домашняя работа 1

Выполнил:

Жаров Александр Павлович

Группа:

K33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2023 г.

Задача

Игра на флексбоксы

Ход работы

1.

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

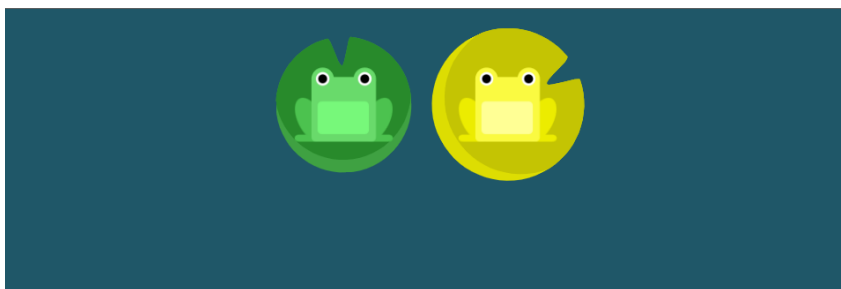
Следующий



2.

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

Следующий



3.

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

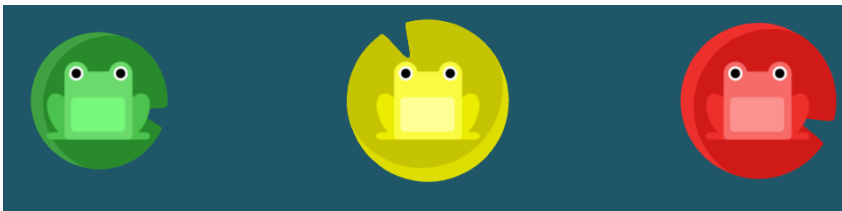
Следующий



4.

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

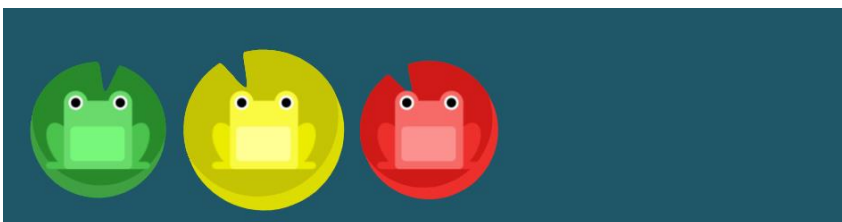
Следующий



5.

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

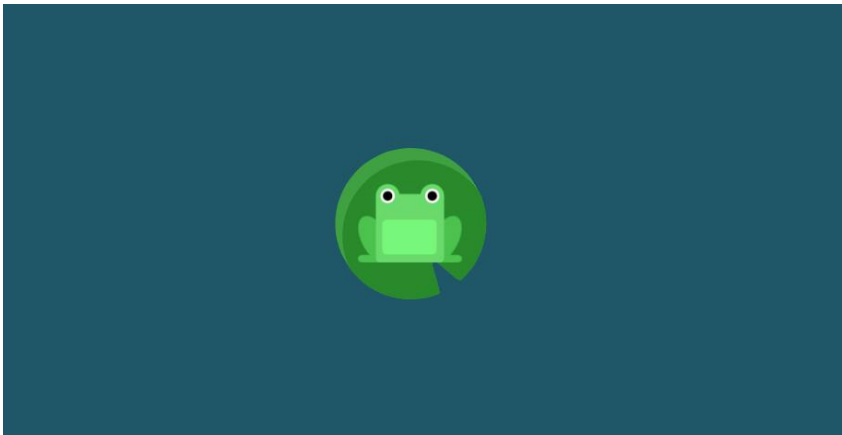
Следующий



6.

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

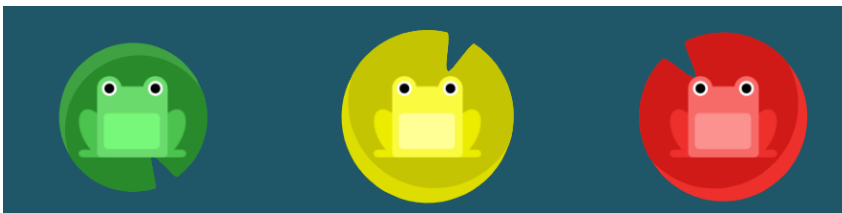
Следующий



7.

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

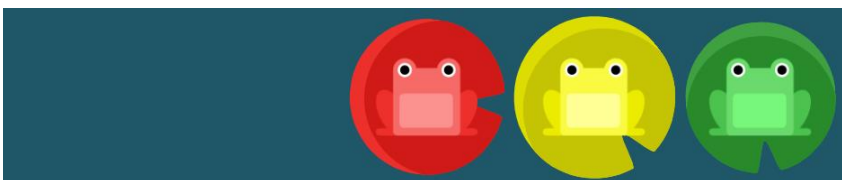
Следующий



8.

```
1 #pond {  
2   display: flex;  
3   flex-direction: row-reverse;  
4 }  
5  
6  
7  
8  
9  
10
```

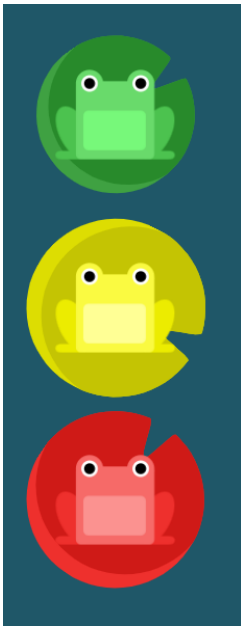
Следующий



9.

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

Следующий



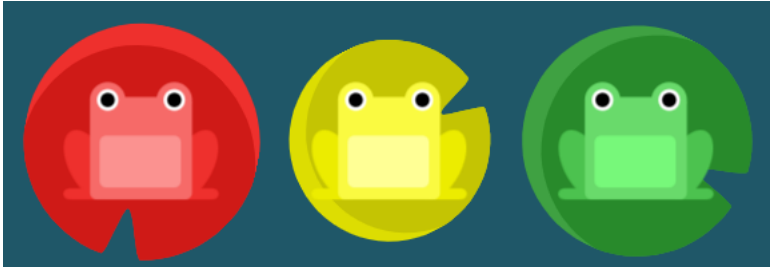
10.

```

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

```

Следующий



11.

FLEXBOX FROGGY

Помогите лягушкам найти их лисичку с помощью **flex-direction** и **justify-content**.

Запомните, 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 }
6
7
8
9
10

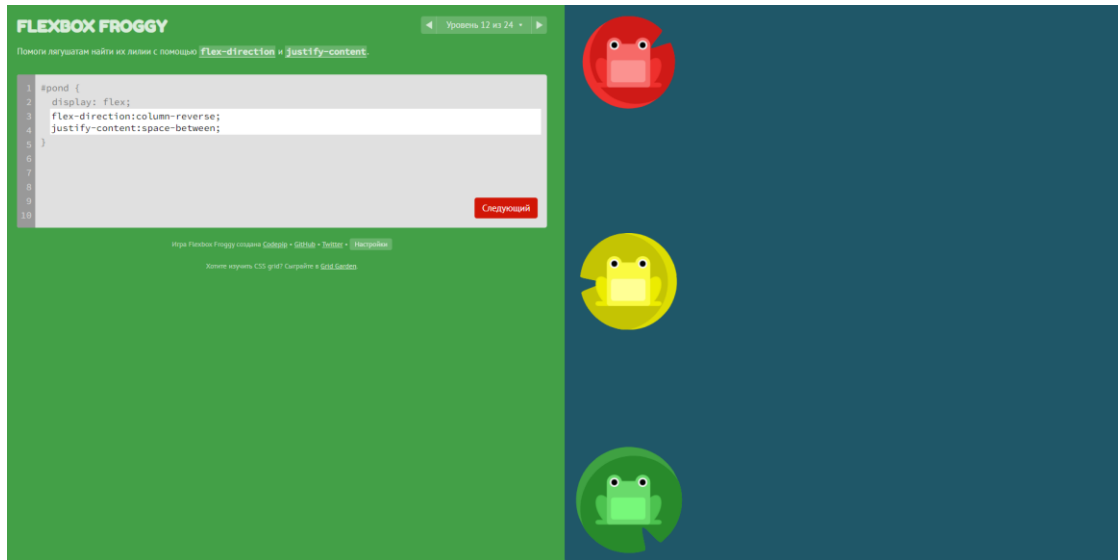
```

Следующий

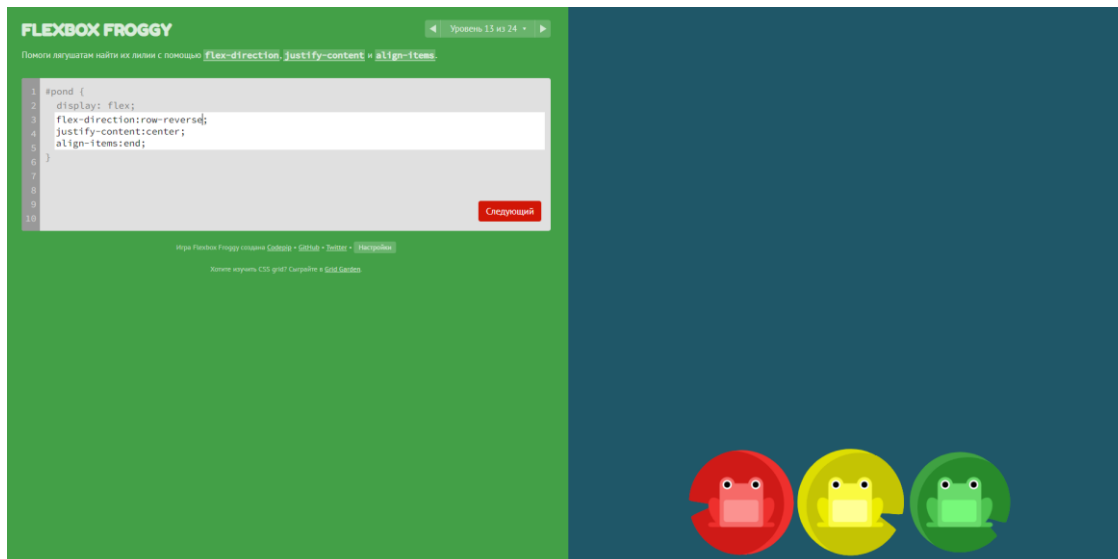
Игра Flexbox Froggy создана [Colton](#) • [GitHub](#) • [Twitter](#) • [YouTube](#)

Хотите научиться CSS grid? [Смотрите в Grid Garden](#)

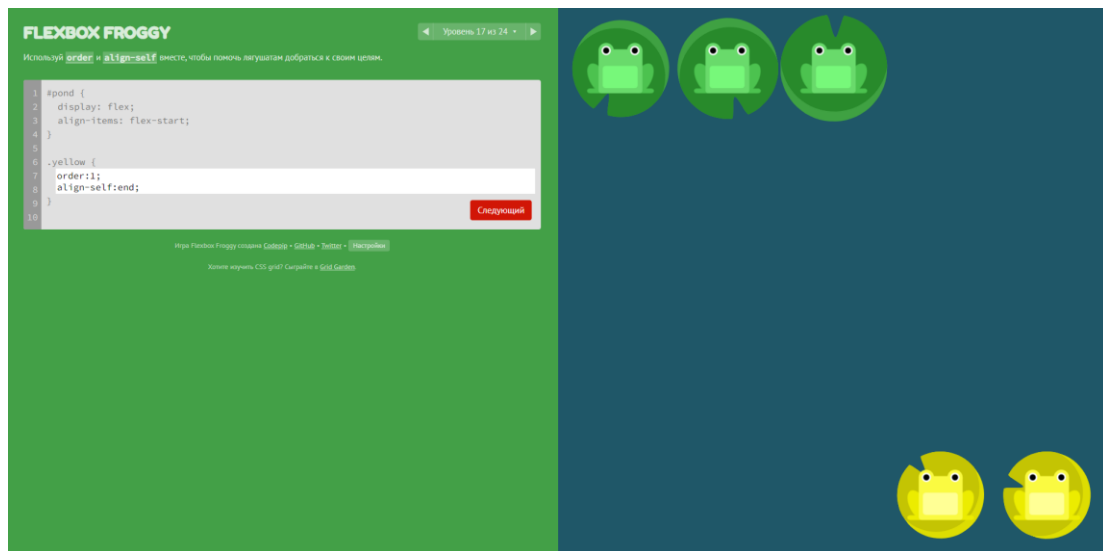
12.



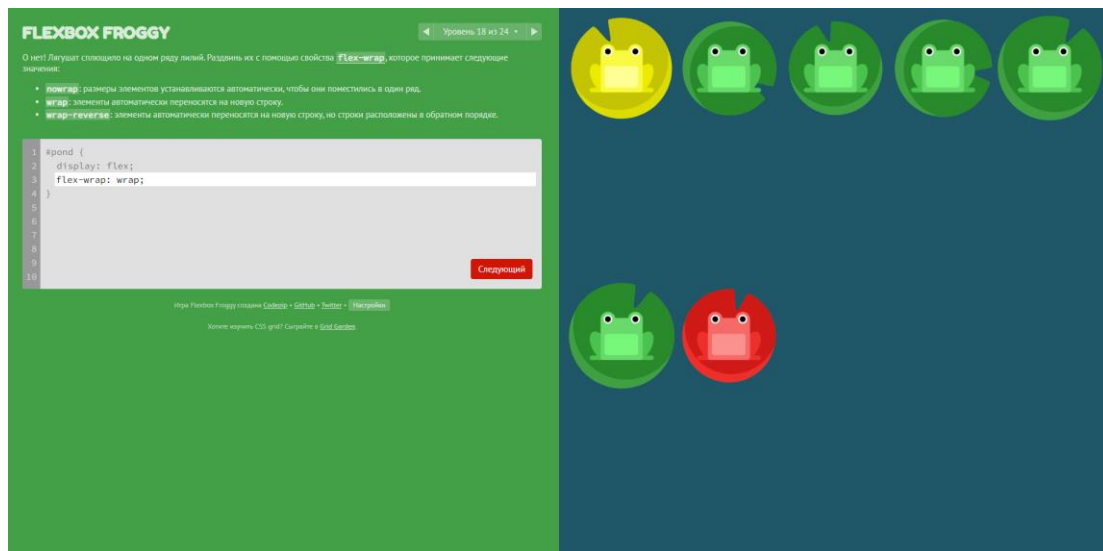
13.



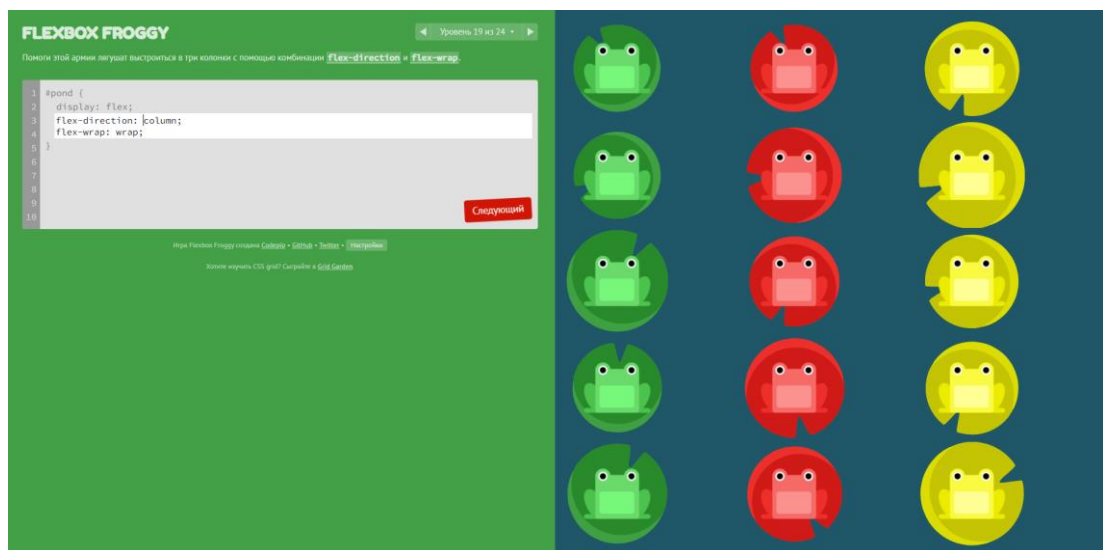
14.



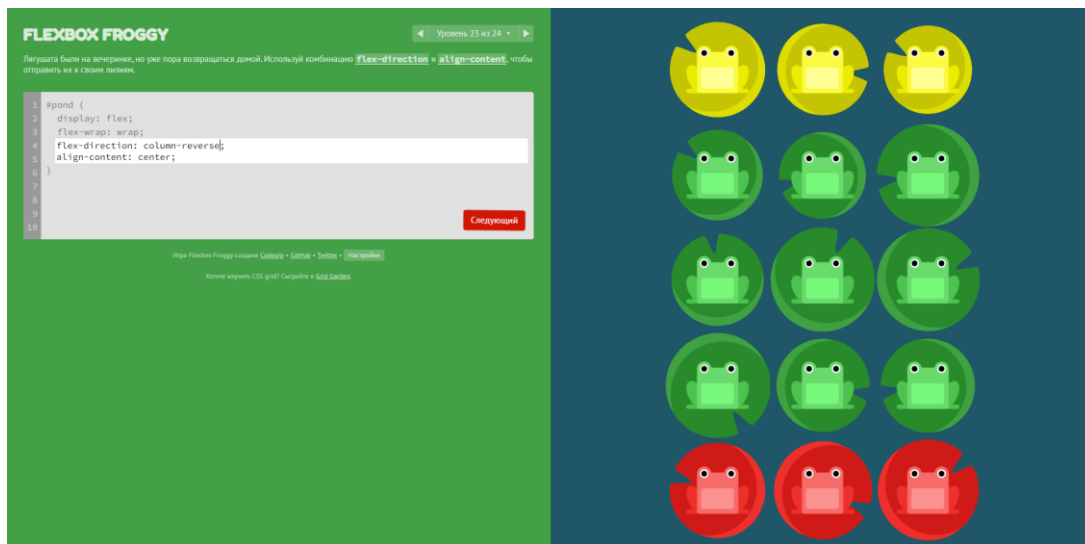
18.



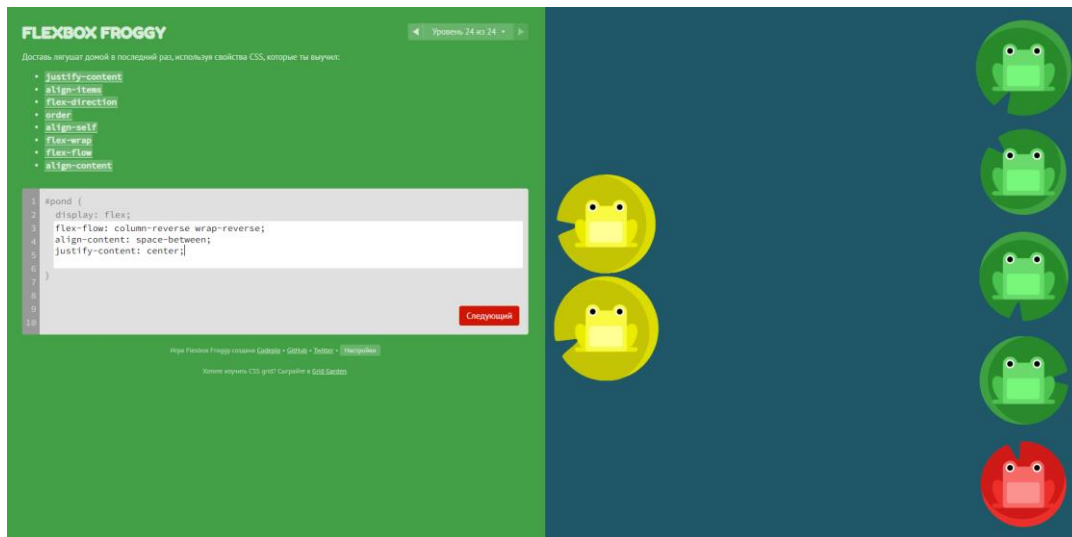
19.



20.



24.



Задание 2:

Игра на гриды

Ход выполнения:

1.

GRID GARDEN

Уровень 1 из 28

Добро пожаловать в Grid Garden — место, где вы напишете CSS-код, чтобы вырастить морковный сад. Поливайте только те зоны, на которых есть морковь, используя свойство `grid-column-start`.

Например, `grid-column-start: 3` покрывает водой зону, начинающуюся на третьей grid-линии по вертикали, то как сказать "третья вертикальная граница на grid-сетке слева".

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: 3
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • Русский

Want to learn CSS Flexbox? Play Flexbox Froggy

2.

GRID GARDEN

Уровень 2 из 28

Оу, кажется, сорняки растут прямо в углу сада. Используйте `grid-column-start`, чтобы отправить их. Помните: сорняки начинаются на первой вертикальной grid-линии.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #poison {
8   grid-column-start: 1
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • Русский

Want to learn CSS Flexbox? Play Flexbox Froggy

3.

GRID GARDEN

Уровень 3 из 28

Когда вы используете только `grid-column-start`, grid-элемент по умолчанию "захватит" только один столбец. Однако вы можете увеличить элемент между несколькими столбцами, если добавите свойство `grid-column-end`.

Используйте `grid-column-end`, полейте всю морковь, избегая пустые участки. Ведь мы не хотим использовать воду напрасно! Помните, что морковь начинается на первой вертикальной grid-линии и заканчивается на четвертой.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: 1;
9   grid-column-end: 4;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • Русский

Want to learn CSS Flexbox? Play Flexbox Froggy

4.

GRID GARDEN

Уровень 4 из 28

Когда вы соедините `grid-column-start` и `grid-column-end`, вы могли подумать, что конечное значение обязательно должно быть больше начального. Но это не так!

Попробуйте задать свойству `grid-column-end` значение меньше, чем 5, чтобы полить морковь.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: 5;
9   grid-column-end: 2;
10 }
11
12
13
14
```

Следующий

[Hype Grid Garden](#) [codepen](#) [Github](#) [Twitter](#) [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

5.

GRID GARDEN

Уровень 5 из 28

Если вы хотите посчитать grid-строки справа налево вместо слева направо, вы можете задать `grid-column-start` и `grid-column-end` отрицательные значения. Например, вы можете присвоить значение, равное `-1`, чтобы указать первую grid-строку начиная справа.

Попробуйте присвоить `grid-column-end` отрицательное значение.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: 1;
9   grid-column-end: -2;
10 }
11
12
13
14
```

Следующий

[Hype Grid Garden](#) [codepen](#) [Github](#) [Twitter](#) [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

6.

GRID GARDEN

Уровень 6 из 28

Теперь попробуйте присвоить `grid-column-start` отрицательное значение.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #poison {
8   grid-column-start: -3;
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

7.

GRID GARDEN

Уровень 7 из 28

Вместо определения grid-элемента на основе начальной и конечной позиции grid-строк вы можете задавать их с помощью необходимой вам ширины строк, используя `span`. Помните, что `span` работает только с положительными значениями. Для примера, поместите эту морковь, используя свойство `grid-column-end: span 2`.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: 2;
9   grid-column-end: span 2;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

8.

GRID GARDEN

Уровень 8 из 28

Попробуйте опять использовать `grid-column-end` вместе со `span`, чтобы полить морковь.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: 1;
9   grid-column-end: span 5;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

9.

GRID GARDEN

Уровень 9 из 28


Вы также можете использовать `span` вместе с `grid-column-start`, чтобы присвоить значение ширины grid-элемента относительно конечной позиции.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 28% 28% 28% 28% 28%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column-start: span 3;
9   grid-column-end: 6;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • Русский

Want to learn CSS flexbox? Play Flexbox Zoo

10.

GRID GARDEN

Уровень 10 из 28

Печать каждый раз `grid-column-start` и `grid-column-end` может быть немного утомительно. К счастью, есть краткая форма `grid-column`, которая принимает оба значения сразу через косую черту `/`.


Например, `grid-column: 2 / 4` скажет grid-элементу начаться на второй вертикальной grid-линии и закончиться на четвёртой.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 28% 28% 28% 28% 28%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column: 4 / 6
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • Русский

Want to learn CSS flexbox? Play Flexbox Zoo

11.

GRID GARDEN

Уровень 11 из 28





Попробуйте использовать `grid-column`, чтобы полить эту морковку. `span` также работает с этим сокращённым свойством, так что попробуйте!

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column: 2 / span 3;
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

12.

GRID GARDEN

Уровень 12 из 28

Одна из вещей, которая отличает CSS Grid Layout от Flexbox'a — это то, что вы можете легко позиционировать элементы в двух направлениях: в столбцах и в строках. `grid-row-start` работает почти как `grid-column-start`, но только по вертикальной оси. Используйте `grid-row-start`, чтобы полить эту морковку.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-row-start: 3;
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

13.

GRID GARDEN

Уровень 13 из 28

А теперь попробуйте использовать сокращённое свойство `grid-row`.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-row: 3 / 6;
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана [Codecademy](#) • [GitHub](#) • [Twitter](#) • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

14.

GRID GARDEN

Уровень 14 из 28

Используйте `grid-column` и `grid-row` одновременно, чтобы разместить элемент в двух направлениях.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #poison {
8   grid-column: 2;
9   grid-row: 4;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

15.

GRID GARDEN

Уровень 15 из 28

Вы также можете использовать `grid-column` и `grid-row` вместе, чтобы охватить более крупные зоны внутри grid-сетки. Попробуйте!

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column: 2 / span 5;
9   grid-row: span 5;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

16.

GRID GARDEN

Уровень 16 из 28

Если вы также уделите время печати `grid-column` и `grid-row`, есть сокращенное свойство и для этого. `grid-area` принимает 4 значения, разделенные косой чертой: `grid-row-start`, `grid-column-start`, `grid-row-end` и `grid-column-end`.
Пример: `grid-area: 1 / 1 / 3 / 6;`

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-area: 1 / 2 / 4 / 6;
9 }
10
11
12
13
14
```

Следующий

Игра Grid Garden создана Codecademy • GitHub • Twitter • [Русский](#)

Want to learn CSS Flexbox? Play [Flexbox Froggy](#)

17.

GRID GARDEN

Уровень 17 из 28

Как насчет множества элементов? Вы можете накладывать их друг на друга без проблем. Используйте `grid-area`, чтобы определить вторую зону, которая покрывает всю не политую морковь.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water-1 {
8   grid-area: 1 / 4 / 6 / 5;
9 }
10
11 #water-2 {
12   grid-area: 2 / 3 / 5 / 6;
13 }
14
```

Следующий

Игра Grid Garden создана Gaborio • GitHub • Twitter • [Русский](#)

Want to learn CSS Gridbox? Play [Gridbox.Froggy](#)

18.

GRID GARDEN

Уровень 18 из 28

Если `grid`-элементы не имеют конкретного расположения с `grid-area`, `grid-column`, `grid-row` и т.д., они автоматически размещаются сверху по порядку, написанному в коде. Мы можем изменить это с помощью свойства `order`, которое является одним из преимуществ CSS Grid Layout перед табличной разметкой.

Изначально все `grid`-элементы имеют `order`, равный 0, но этому свойству можно присвоить любое положительное или отрицательное значение, примерно как у `z-index`.

Сейчас морковь во втором столбце отравлена и сорняки в последнем столбце появляются. Измените значение свойства `order` для кда, чтобы исправить это прямо сейчас!

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 .water {
8   order: 0;
9 }
10
11 #poison {
12   order: 1
13 }
14
```

Следующий

Игра Grid Garden создана Gaborio • GitHub • Twitter • [Русский](#)

Want to learn CSS Gridbox? Play [Gridbox.Froggy](#)

19.

GRID GARDEN

Уровень 19 из 28

Теперь вода и кда меняются местами, хотя все сорняки находятся в начале сада. Присвойте правильное значение `order` для кда, чтобы исправить это.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 .water {
8   order: 0;
9 }
10
11 .poison {
12   order: -1
13 }
14
```

Следующий

Игра Grid Garden создана Gaborio • GitHub • Twitter • [Русский](#)

Want to learn CSS Gridbox? Play [Gridbox.Froggy](#)

20.

GRID GARDEN

Уровень 20 из 28

До этого момента сад имел grid-сетку с пятью столбцами по 20% ширины и пятью строками, каждая по 20% высоты.

Это было сделано свойствами `grid-template-columns: 20% 20% 20% 20% 20%;` и `grid-template-rows: 20% 20% 20% 20% 20%;`. Каждое свойство имеет пять значений, которые создают пять столбцов, где ширина каждого равна 20% от общей ширины сада.

Но вы можете изменить grid-сетку как вам вздумается. Присвойте `grid-template-columns` новое значение, чтобы полить морковь. Вам необходимо поставить значение ширины первого столбца равным 50%.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 50% 20% 20% 20%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column: 1;
9   grid-row: 1;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Gabbay • GitHub • Twitter • YouTube

Want to learn CSS Flexbox? Play Flexbox Games

21.

GRID GARDEN

Уровень 21 из 28

Определить несколько столбцов с одинаковой шириной может быть утомительно. К счастью, есть функция `repeat`, которая призвана помочь в этом.

Например, вместо того чтобы определить пять столбцов по 20% ширины с помощью `grid-template-columns: 20% 20% 20% 20% 20%;`, это можно упростить до `grid-template-columns: repeat(5, 20%);`.

Используя `grid-template-columns` вместе с функцией `repeat`, создайте 8 столбцов по 12.5% ширины каждый. Таким образом, вы не будете использовать больше кода, чем нужно.

```
1 #garden {
2   display: grid;
3   grid-template-columns: repeat(8, 12.5%);
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7 #water {
8   grid-column: 1;
9   grid-row: 1;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Gabbay • GitHub • Twitter • YouTube

Want to learn CSS Flexbox? Play Flexbox Games

22.

GRID GARDEN

Уровень 22 из 28

`grid-template-columns` принимает значения не только в процентах, но и в единицах длины, например `200px` или `400px`. Вы даже можете комбинировать разные единицы измерения.

Присвойте трем столбцам значения `100px`, `3em` и `40%` соответственно.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 100px 3em 40%;
4   grid-template-rows: 20% 20% 20% 20% 20%;
5 }
6
7
8
9
10
11
12
13
14
```

Следующий

Игра Grid Garden создана Gabbay • GitHub • Twitter • YouTube

Want to learn CSS Flexbox? Play Flexbox Games

23.

GRID GARDEN

Уровень 23 из 28

CSS Grid Layout также вводит новую единицу измерения: дробный `fr` (англ. fraction). Каждый `fr` выделяет одну часть свободного пространства. Например, если двум элементам задать `1fr` и `3fr`, соответственно, то пространство будет поделено на 4 одинаковые части. Первый элемент займет ¼, а второй останется ¾ пространства.

Сейчас сорняки занимают левую ¼ часть вашей первой строки, а морковь останется ¾. Создайте две столбца с такой же шириной, используя `fr`.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 1fr 5fr;
4   grid-template-rows: 28% 28% 28% 28%;
5 }
6
7
8
9
10
11
12
13
14
```

Следующий

Нравится Grid Garden? Оставьте [Feedback](#) • [GitHub](#) • [Twitter](#) • [YouTube](#)

Want to learn CSS Grid? Play [Flexbox](#) • [Grid](#)

24.

GRID GARDEN

Уровень 24 из 28

Когда один столбец определен с помощью пикселей, процентов или еще, а любые другие столбцы – с помощью `fr`, то последнее просто разделит все возможное оставшееся пространство.

Сейчас морковь занимает 50 пикселей слева, а сорняки 50 пикселей справа. С помощью `grid-template-columns` создайте два столбца и используйте `fr`, чтобы сделать еще 3 столбца, которые займут оставшееся пространство между ними.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 50px 1fr 1fr 1fr 50px;
4   grid-template-rows: 28% 28% 28% 28%;
5 }
6
7 #water {
8   grid-area: 1 / 1 / 6 / 2;
9 }
10
11 #poison {
12   grid-area: 1 / 5 / 6 / 6;
13 }
14
```

Следующий

Нравится Grid Garden? Оставьте [Feedback](#) • [GitHub](#) • [Twitter](#) • [YouTube](#)

Want to learn CSS Grid? Play [Flexbox](#) • [Grid](#)

25.

GRID GARDEN

Уровень 25 из 28

Теперь у нас есть столбец сорняков с шириной 75 пикселей на левой стороне сада. ¼ оставшегося пространства занимает рисунок моркови, а ¾ заполняют сорняки.

Используйте `grid-template-columns` с комбинацией `px` и `fr`, чтобы создать необходимые столбцы.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 75px 3fr 2fr;
4   grid-template-rows: 188px;
5 }
6
7
8
9
10
11
12
13
14
```

Следующий

Нравится Grid Garden? Оставьте [Feedback](#) • [GitHub](#) • [Twitter](#) • [YouTube](#)

Want to learn CSS Grid? Play [Flexbox](#) • [Grid](#)

26.

GRID GARDEN

Уровень 26 из 28

grid-template-rows

работает точно так же, как и grid-template-columns.

Используйте **grid-template-rows**, чтобы полить все, кроме верхних 50 пикселей сада. Помните, что вода заполняет только левую сторону, так что вам нужно создать именно пять строк.

```
1 #garden {
2   display: grid;
3   grid-template-columns: 20% 20% 20% 20% 20%;
4   grid-template-rows: 50px 0fr 0fr 0fr 1fr
5 }
6
7 #water {
8   grid-column: 1 / 6;
9   grid-row: 5 / 6;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Gabbay • GitHub • Twitter • YouTube

Want to learn CSS Flexbox? Play Flexbox Tropy

27.

GRID GARDEN

Уровень 27 из 28

grid-template

— сокращенный вариант комбинации **grid-template-rows** и **grid-template-columns**.

Например, **grid-template: 50% 50% / 200px** создаст grid-сетку с двумя строками по 50% каждая и одним столбцом шириной 200 пикселей.

Попробуйте использовать **grid-template**, чтобы полить зону, включающую в себя верхние 60% и левые 200 пикселей сада.

```
1 #garden {
2   display: grid;
3   grid-template: 60% / 200px;
4 }
5
6
7 #water {
8   grid-column: 1;
9   grid-row: 1;
10 }
11
12
13
14
```

Следующий

Игра Grid Garden создана Gabbay • GitHub • Twitter • YouTube

Want to learn CSS Flexbox? Play Flexbox Tropy

28.

GRID GARDEN

Уровень 28 из 28

Сад выглядит прекрасно. Здесь вы оставили 50-пиксельную дорожку снизу и заполнили все остальное пространство морковью.

К сожалению, левые 20% сада заполнили сорняки. Используйте CSS Grid Layout в последний раз, чтобы очистить сад.

```
1 #garden {
2   display: grid;
3   grid-template: 100% / 20% 80%;
4 }
5
6
7
8
9
10
11
12
13
14
```

Следующий

Игра Grid Garden создана Gabbay • GitHub • Twitter • YouTube

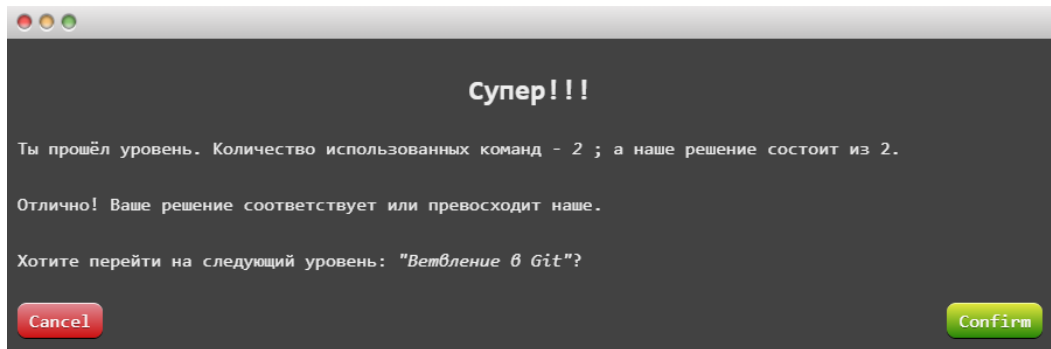
Want to learn CSS Flexbox? Play Flexbox Tropy

Задание 3:

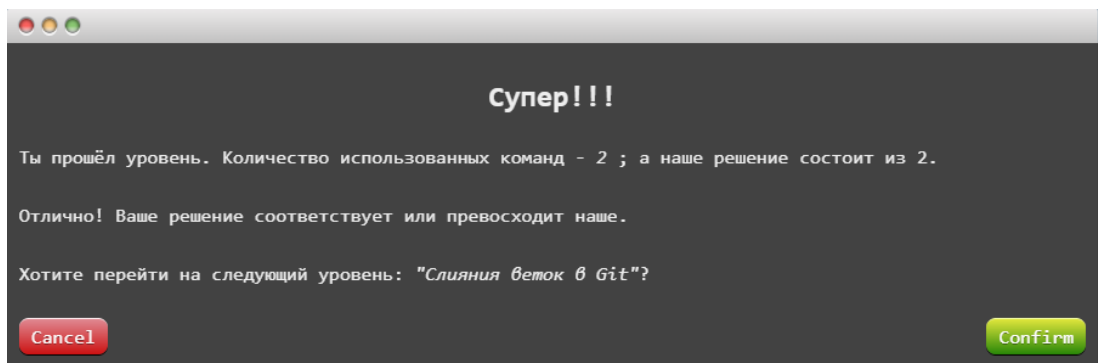
Игра на знание Git

Ход выполнения:

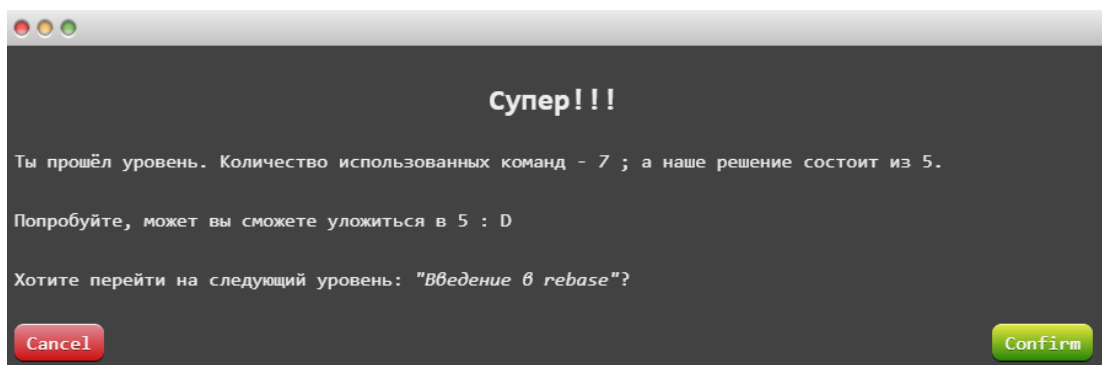
1.1



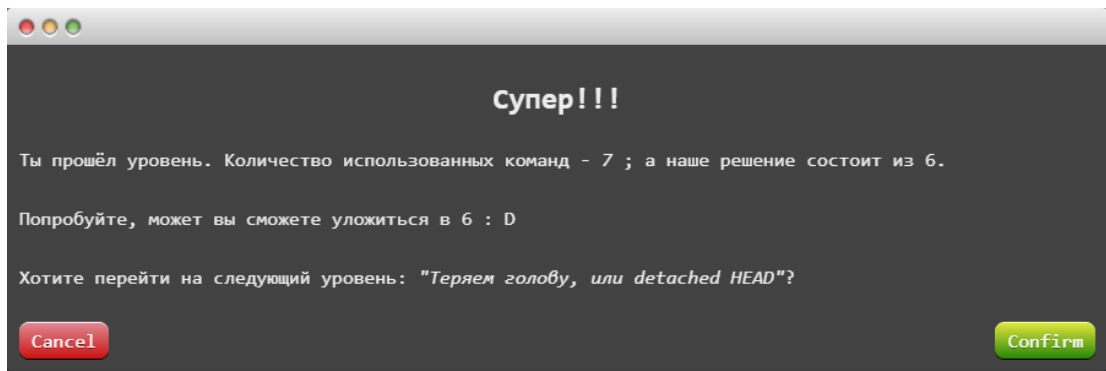
1.2



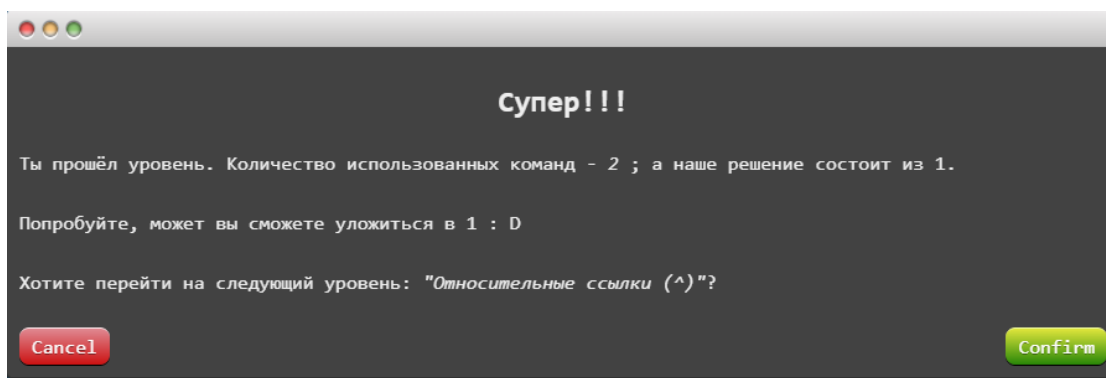
1.3



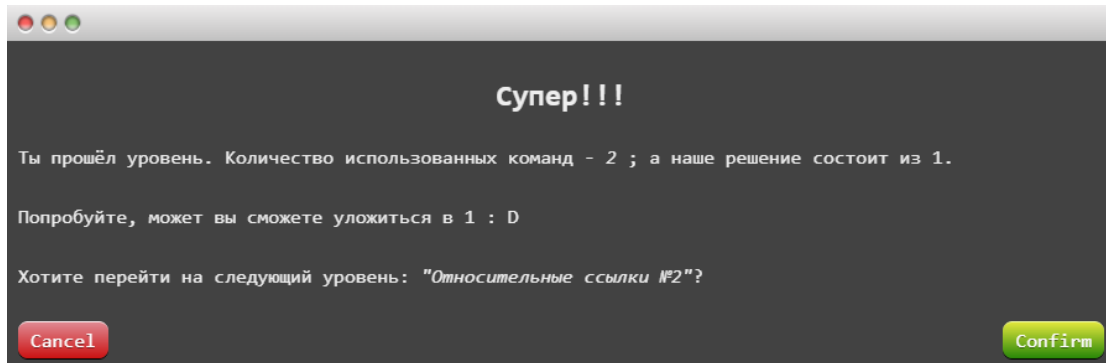
1.4



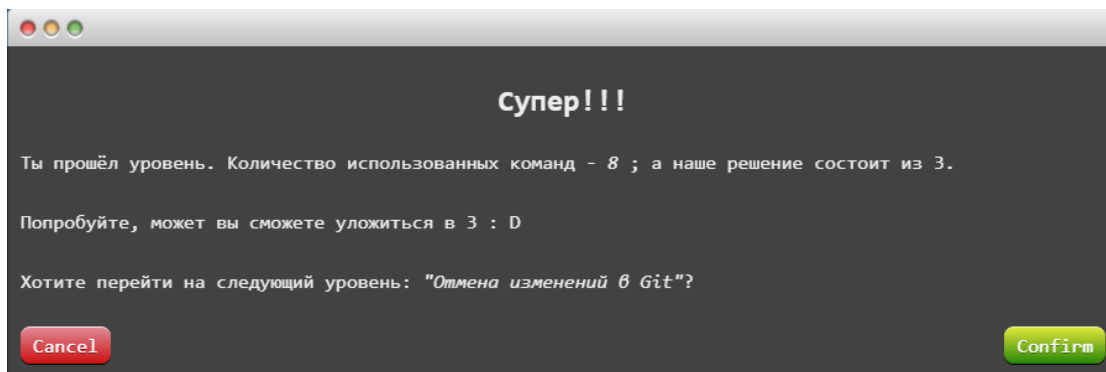
2.1



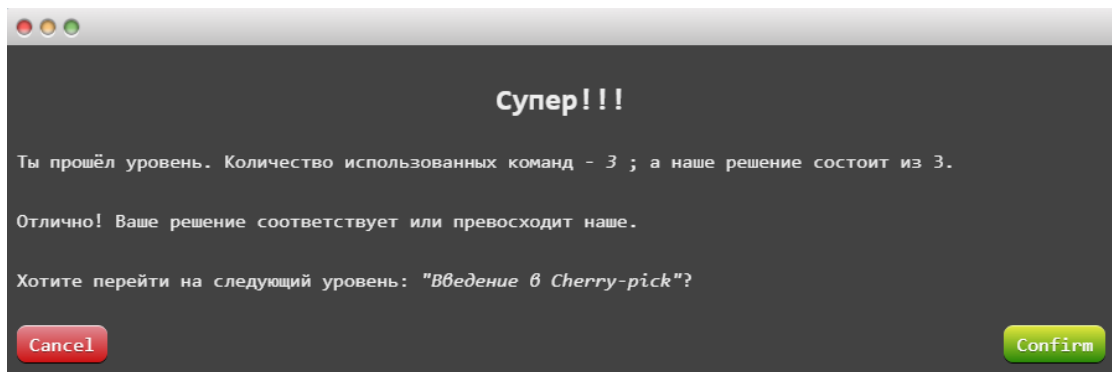
2.2



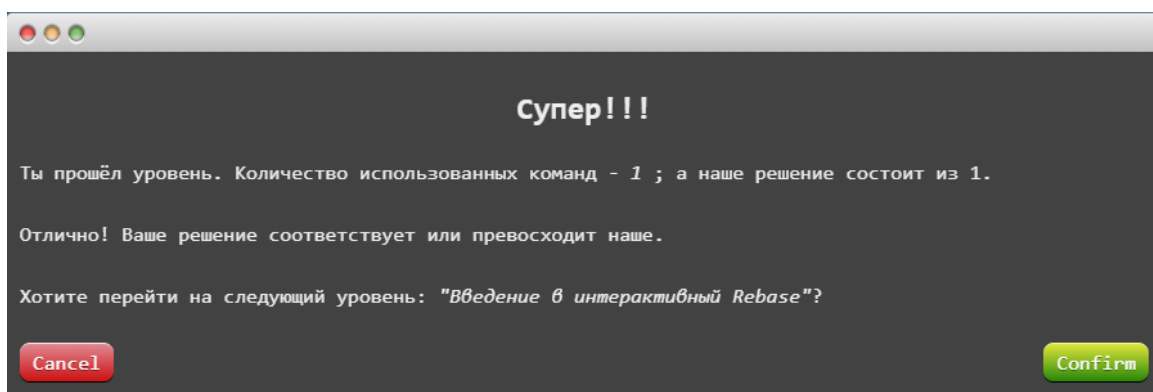
2.3



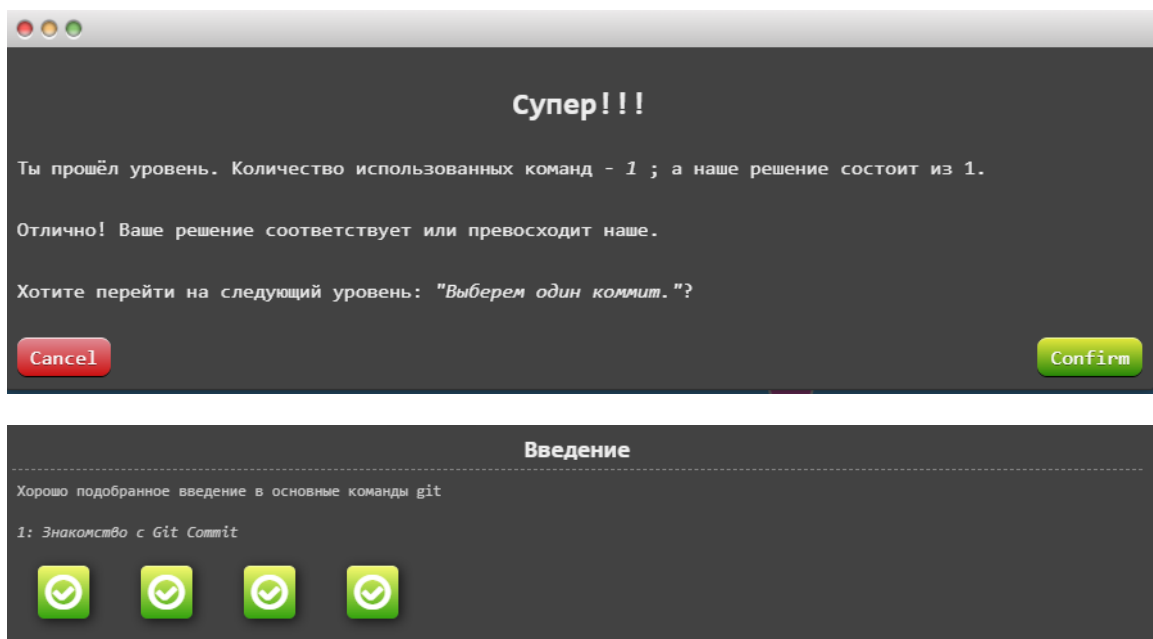
2.4



3.1



3.2



Вывод

В ходе выполнения работы я повторил основы верстки на гридах и флексбоксах, а также основные команды для работы с git.