

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Домашняя работа 1

Выполнил:

Косарев Илья

Группа

J3210

Проверил:

Добряков Д. И.

Санкт-Петербург

2026 г.

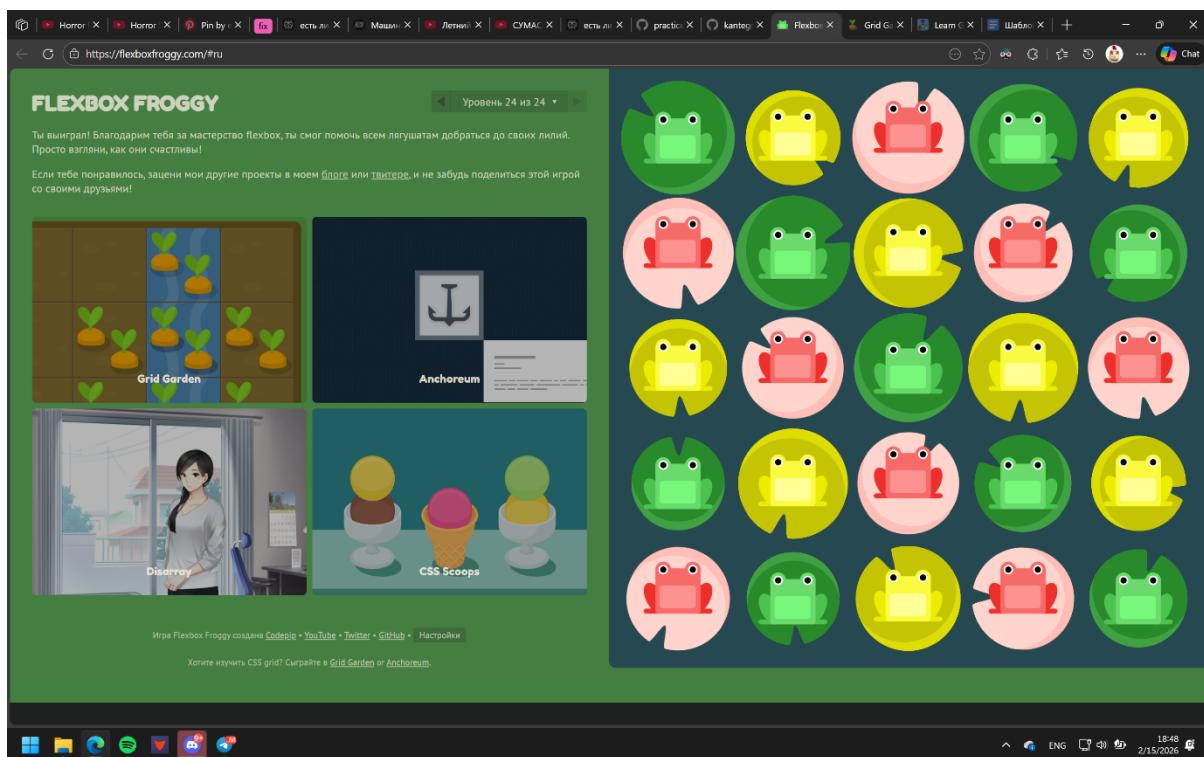
Задача

Изучить CSS Flexbox и Grid, Git, пройдя онлайн-игры:

- <https://flexboxfroggy.com/#ru>
- <https://cssgridgarden.com/#ru>
- https://learngitbranching.js.org/?locale=ru_RU

Ход работы

В ходе прохождения первой игры были изучены синтаксис свойств в CSS и применение Flexbox.



Были изучены свойства:

- justify-content управляет выравниванием элементов вдоль главной оси. Значение flex-start выравнивает элементы по началу контейнера; flex-end выравнивает элементы по концу; center выравнивает элементы по центру; space-between распределяет элементы так, что первый и последний прижаты к краям, а отступы между остальными равны; space-around распределяет элементы так, что отступы вокруг каждого элемента равны; space-evenly распределяет элементы так,

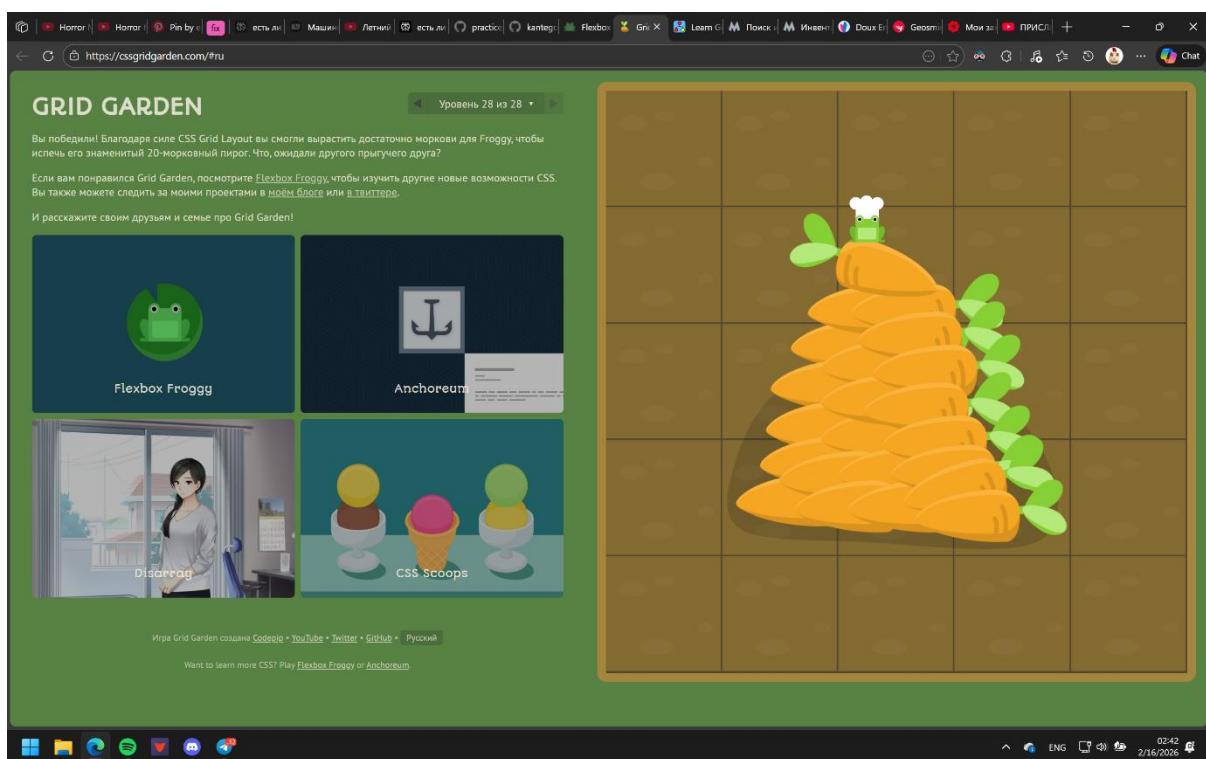
что отступы между всеми элементами и краями контейнера абсолютно равны.

- align-items управляет выравниванием элементов вдоль поперечной оси в пределах одной строки. Значение flex-start выравнивает элементы по началу поперечной оси; flex-end выравнивает элементы по концу; center выравнивает элементы по центру; baseline выравнивает элементы по базовой линии их текста; stretch растягивает элементы на всю доступную высоту или ширину строки, заполняя пространство.
- flex-direction управляет направлением главной оси, определяя линию построения элементов. Значение row выстраивает элементы горизонтально слева направо; row-reverse выстраивает элементы горизонтально справа налево; column выстраивает элементы вертикально сверху вниз; column-reverse выстраивает элементы вертикально снизу вверх.
- order управляет визуальным порядком отдельного элемента. Значение 0 сохраняет исходный порядок элемента (по умолчанию); положительные значения перемещают элемент ближе к концу списка; отрицательные значения перемещают элемент ближе к началу списка.
- align-self управляет выравниванием отдельного элемента вдоль поперечной оси, переопределя общее правило align-items. Значение flex-start выравнивает элемент по началу оси; flex-end выравнивает элемент по концу оси; center выравнивает элемент по центру оси; baseline выравнивает элемент по базовой линии текста; stretch растягивает элемент на всю доступную высоту или ширину.
- flex-wrap управляет правилами переноса элементов при нехватке места в контейнере. Значение nowrap запрещает перенос, выстраивая элементы строго в одну строку; wrap разрешает перенос элементов на новые строки сверху вниз; wrap-reverse разрешает перенос элементов на новые строки снизу вверх.
- flex-flow является кратким свойством, объединяющим flex-direction и flex-wrap. Оно позволяет одновременно задать и направление главной оси, и правила переноса элементов.
- align-content управляет выравниванием целых строк вдоль поперечной оси при многострочной раскладке. Значение flex-start выравнивает строки по началу контейнера; flex-end выравнивает

строки по концу контейнера; center выравнивает строки по центру; space-between распределяет строки так, что крайние прижаты к краям контейнера; space-around создает равные отступы вокруг каждой строки; space-evenly создает абсолютно равные отступы между всеми строками и краями; значение stretch растягивает строки на всю высоту контейнера.

Эта игра была полезна как первое знакомство с объектами и свойствами CSS (в частности, Flexbox). Трудностей не возникло.

В ходе прохождения второй игры были изучены свойства CSS Grid.



Были изучены свойства:

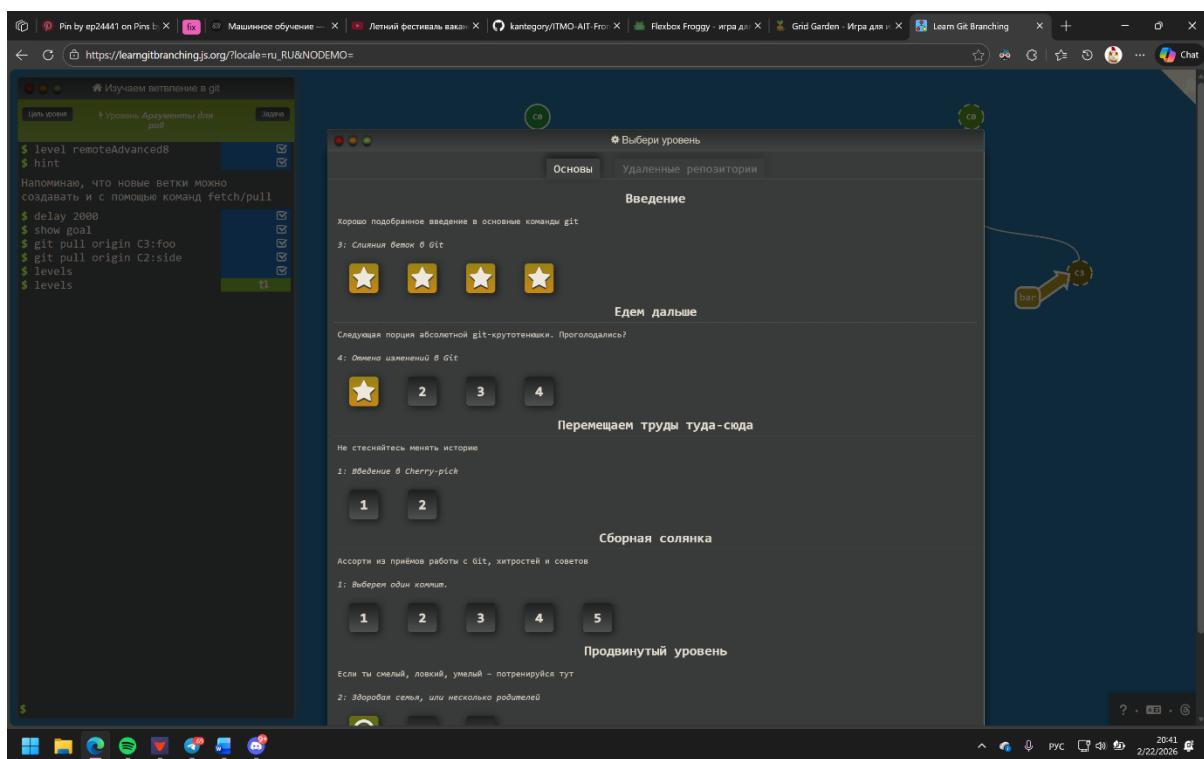
- grid-template-columns и grid-template-rows определяют структуру сетки, задавая количество и размеры колонок или строк. Значение в пикселях (px) или процентах устанавливает фиксированный размер полосы; fr (фракция) распределяет свободное пространство пропорционально; auto подстраивает размер под содержимое; repeat() автоматически дублирует полосы заданное количество раз.
- grid-column-start и grid-column-end определяют начальную и конечную позиции элемента по горизонтали. Значение номер линии указывает конкретную вертикальную линию сетки, у которой

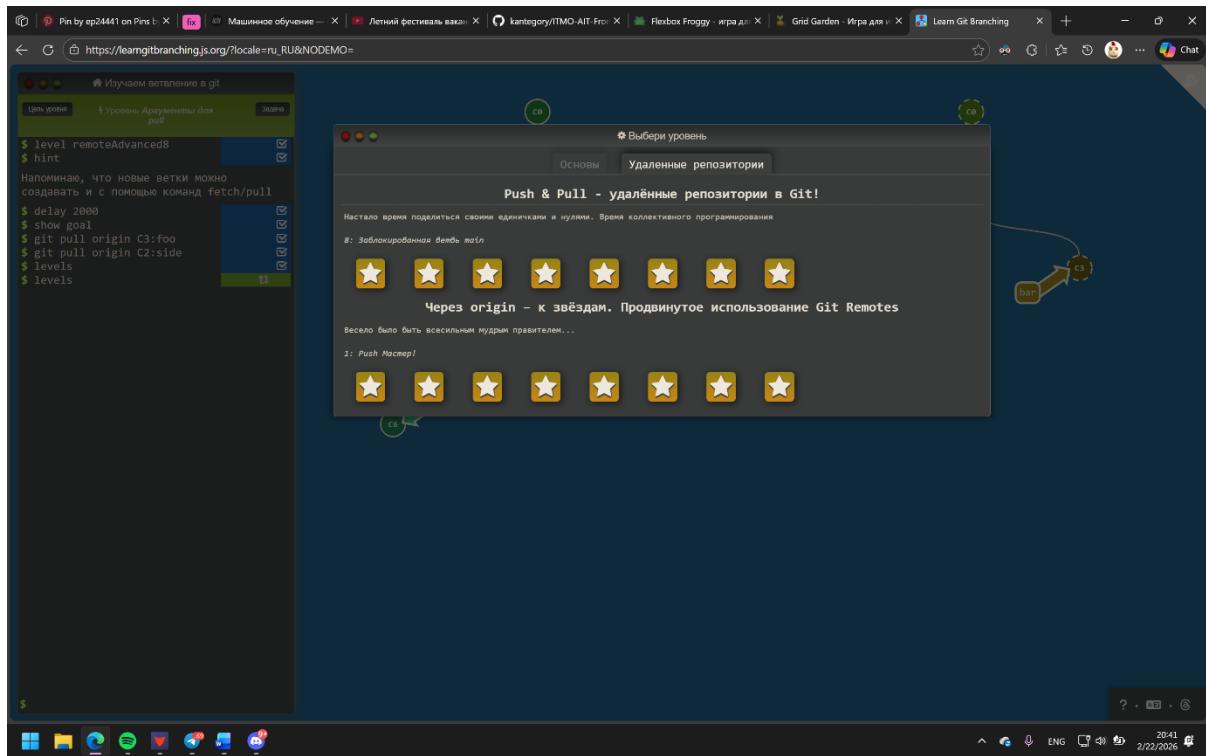
элемент начинается или заканчивается; span указывает количество колонок, которые элемент должен занять.

- grid-column является кратким свойством, объединяющим grid-column-start и grid-column-end.
- grid-row-start и grid-row-end определяют начальную и конечную позиции элемента по вертикали. Значение номер линии указывает конкретную горизонтальную линию сетки для начала или конца элемента; span указывает количество строк, которые элемент должен охватить.
- grid-row является кратким свойством, объединяющим grid-row-start и grid-row-end.
- grid-area управляет комплексным размещением элемента в сетке. Значение row-start / col-start / row-end / col-end задает все четыре границы элемента одновременно.

В ходе игры я познакомился с CSS Grid и закрепил работу со свойствами в CSS.

В ходе прохождения третьей игры я отточил работу с Git и узнал новые свойства знакомых команд.





Единственное, что вызвало затруднение – это блок с `git pull`, где отрабатывается свойство `<источник>:<целевая ветка>`, так как приходилось учитывать, что команда заключает функционал `git fetch` и `git merge` одновременно.

Вывод

В ходе выполнения работы я познакомился с работой со свойствами объектов в CSS, а также с Flexbox и Grid: свойства Flexbox для форматирования и создания flex-контейнеров, свойства Grid для построения сетки и настройки размеров её строк и столбцов. Наконец, я отточил навыки работы с Git, из-за чего смогу более эффективно его использовать при организации работы с проектами.