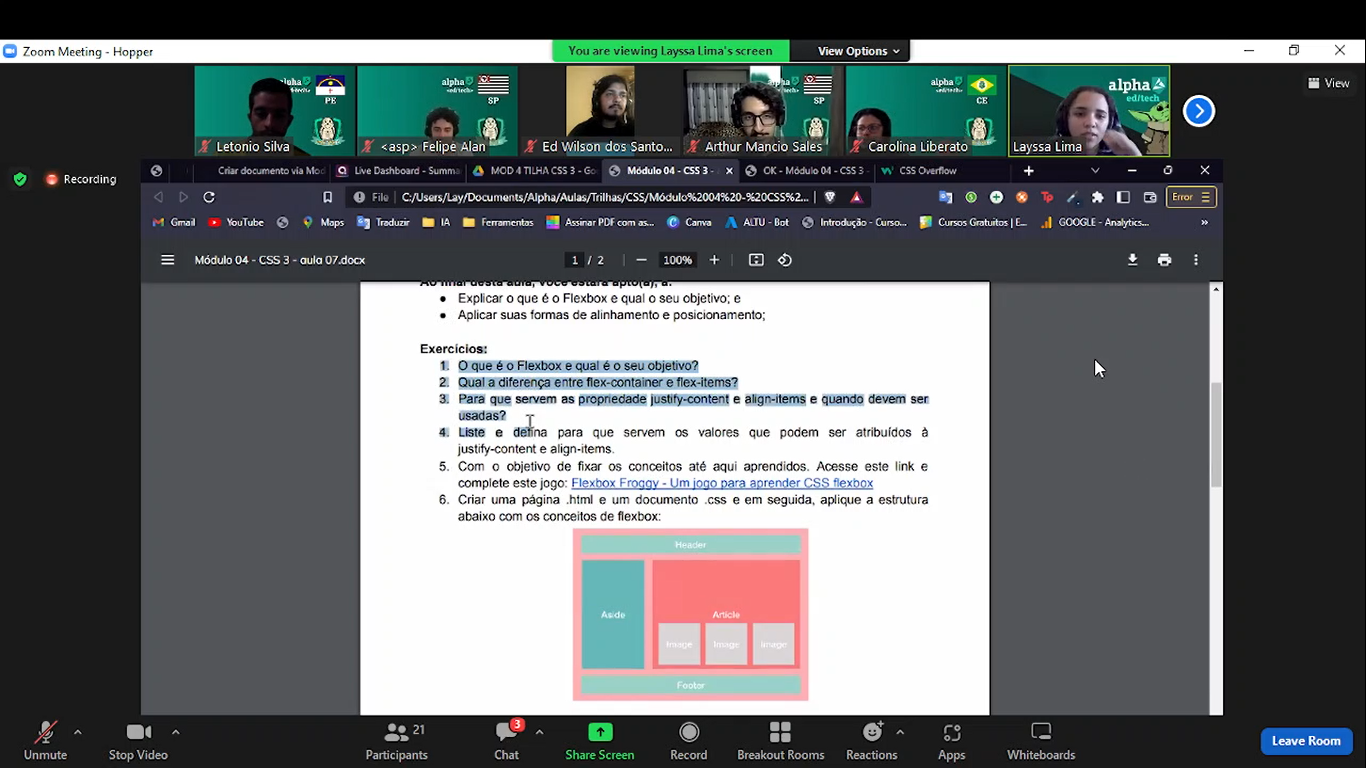
Atividade 06



Q01) O Flexible Box Module, normalmente chamado de flexbox, foi projetado tanto como um modelo de layout unidimensional quanto como um método capaz de organizar espacialmente os elementos em uma interface, além de possuir capacidades avançadas de alinhamento.Quando se descreve o flexbox como sendo unidimensional, enfatiza-se o fato de que ele lida com o layout em uma dimensão de cada vez - seja uma linha ou uma coluna. Isto pode ser comparado com o modelo bidimensional de CSS - Layout de Grade, que permite o controle simultâneo das colunas e linhas. Ao se utilizar o flexbox, é preciso ter em mente que todas as operações realizadas relacionam-se a dois eixos: o eixo principal e o eixo transversal. O eixo principal é definido através da propriedade flex-direction e o eixo transversal encontra-se na direção perpendicular a ele.

Q02)

A área de um documento que faz uso do flexbox é chamada de contêiner flex. Para criar essa estrutura, define-se o valor da propriedade display do elemento representado pelo contêiner como flex ou inline-flex. Desse modo, os elementos-filhos desse contêiner tornar-se-ão do tipo flex. Os elementos filhos são chamados de flex-items. Alguns tipos de propriedades devem ser declaradas no container, enquanto outras precisam ser declaradas no elementos filhos.

Q03) Justify-content->Esta propriedade define o alinhamento dos itens ao longo do eixo principal. Ajuda a distribuir o espaço livre que sobrar no container tanto se todos os flex items em uma linha são inflexíveis, ou são flexíveis mas já atingiram seu tamanho máximo. Também exerce algum controle sobre o alinhamento de itens quando eles ultrapassam o limite da linha. Pode receber os valores: flex-start, flex-end, space-between, space-around e space-evenl.

No align-items->define o comportamento padrão de como flex items são alinhados de acordo com o eixo transversal (cross axis). De certa forma, funciona de forma similar ao justify-content, porém no eixo transversal (perpendicular ao eixo principal). Pode receber os valores: stretch, flex-start, flex-end, center e baseline.

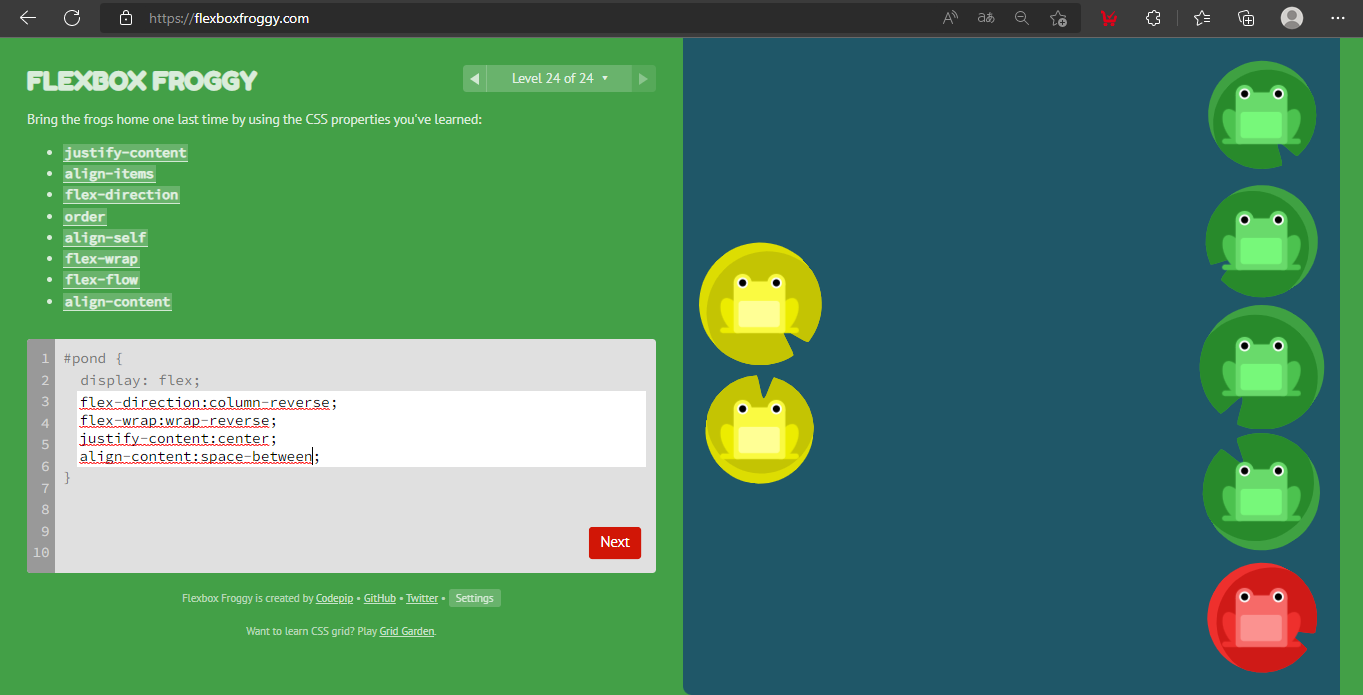
Q4) Começando com justify-content:

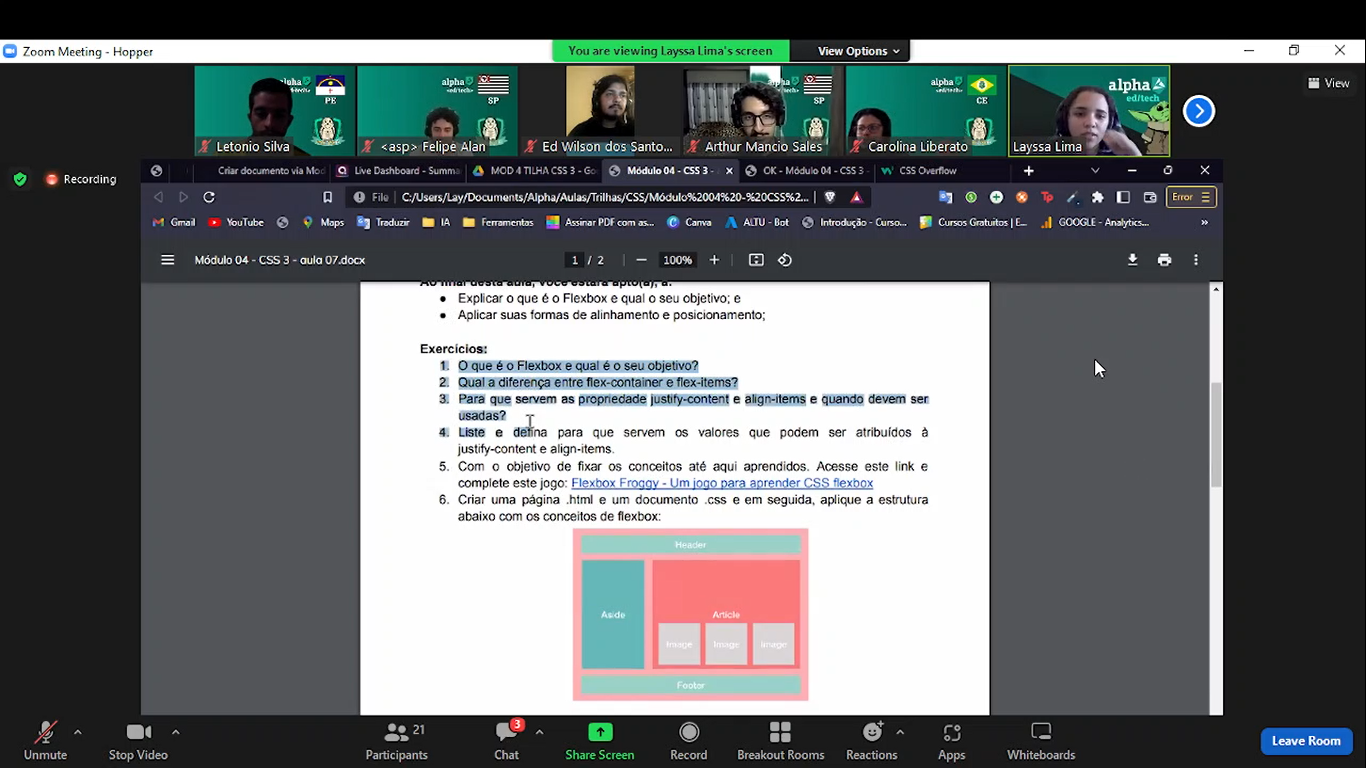
* flex-start (padrão): os ítens são alinhados junto à borda de início (start) de acordo com qual for a flex-direction do container.
* flex-end: os ítens são alinhados junto à borda final (end) de acordo com qual for a flex-direction do container.
* start: os ítens são alinhados junto à borda de início da direção do writing-mode (modo de escrita).
* end: os ítens são alinhados junto à borda final da direção do writing-mode (modo de escrita).
* left: os ítens são alinhados junto à borda esquerda do container, a não ser que isso não faça sentido com o flex-direction que estiver sendo utilizado. Nesse caso, se comporta como start.
* right: os ítens são alinhados junto à borda direita do container, a não ser que isso não faça sentido com o flex-direction que estiver sendo utilizado. Nesse caso, se comporta como start.
* center: os ítens são centralizados na linha.
* space-between: os ítens são distribuídos de forma igual ao longo da linha; o primeiro ítem junto à borda inicial da linha, o último junto à borda final da linha.
* space-around: os ítens são distribuídos na linha com o mesmo espaçamento entre eles. Note que, visualmente, o espaço pode não ser igual, uma vez que todos os ítens tem a mesma quantidade de espaço dos dois lados: o primeiro item vai ter somente uma unidade de espaço junto à borda do container, mas duas unidades de espaço entre ele e o próximo ítem, pois o próximo ítem também tem seu próprio espaçamento que está sendo aplicado.
* space-evenly: os ítens são distribuídos de forma que o espaçamento entre quaisquer dois itens da linha (incluindo entre os ítens e as bordas) seja igual.

Sobre o align-items:

* stretch (padrão): estica os ítens para preencher o container, respeitando o min-width/max-width).
* flex-start/ start / self-start: ítens são posicionados no início do eixo transversal. A diferença entre eles é sutil e diz respeito às regras de flex-direction ou writing-mode.
* center: ítens são centralizados no eixo transversal.
* baseline: ítens são alinhados de acordo com suas baselines.

Q05) Entrei no link para completar o game:





Q06)