**CSS CODE**

**justify-content** property, which aligns items horizontally and accepts the following values:

* **flex-start**: Items align to the left side of the container.
* **flex-end**: Items align to the right side of the container.
* **center**: Items align at the center of the container.
* **space-between**: Items display with equal spacing between them.
* **space-around**: Items display with equal spacing around them.

**align-items** to help the frogs get to the bottom of the pond. This CSS property aligns items vertically and accepts the following values:

* **flex-start**: Items align to the top of the container.
* **flex-end**: Items align to the bottom of the container.
* **center**: Items align at the vertical center of the container.
* **baseline**: Items display at the baseline of the container.
* **stretch**: Items are stretched to fit the container.

**flex-direction**. This CSS property defines the direction items are placed in the container, and accepts the following values:

* **row**: Items are placed the same as the text direction.
* **row-reverse**: Items are placed opposite to the text direction.
* **column**: Items are placed top to bottom.
* **column-reverse**: Items are placed bottom to top.

Sometimes reversing the row or column order of a container is not enough. In these cases, we can apply the **order** property to individual items. By default, items have a value of 0, but we can use this property to also set it to a positive or negative integer value (-2, -1, 0, 1, 2).

**align-self**. This property accepts the same values as **align-items** and its value for the specific item.

**flex-wrap** property, which accepts the following values:

* **nowrap**: Every item is fit to a single line.
* **wrap**: Items wrap around to additional lines.
* **wrap-reverse**: Items wrap around to additional lines in reverse.

**flex-direction** and **flex-wrap** are used so often together that the shorthand property **flex-flow** was created to combine them. This shorthand property accepts the value of the two properties separated by a space.

For example, you can use **flex-flow: row wrap** to set rows and wrap them.

Try using **flex-flow** to repeat the previous level.

**align-content** to set how multiple lines are spaced apart from each other. This property takes the following values:

* **flex-start**: Lines are packed at the top of the container.
* **flex-end**: Lines are packed at the bottom of the container.
* **center**: Lines are packed at the vertical center of the container.
* **space-between**: Lines display with equal spacing between them.
* **space-around**: Lines display with equal spacing around them.
* **stretch**: Lines are stretched to fit the container.

This can be confusing, but **align-content** determines the spacing between lines, while **align-items** determines how the items as a whole are aligned within the container. When there is only one line, **align-content** has no effect.