**justify-content**, qui aligne les éléments horizontalement et accepte les valeurs suivantes :

* **flex-start** : Les éléments s'alignent au côté gauche du conteneur.
* **flex-end** : Les éléments s'alignent au côté droit du conteneur.
* **center** : Les éléments s'alignent au centre du conteneur.
* **space-between** : Les éléments s'affichent avec un espace égal entre eux.
* **space-around** : Les éléments s'affichent avec un espacement égal à l'entour d'eux.

#pond {

display: flex;

justify-content: space-around;

}

**align-items** pour aider les grenouilles à se rendre au bas de l'étang. Cette propriété CSS aligne les éléments verticalement et accepte les valeurs suivantes :

* **flex-start** : Les éléments s'alignent au haut du conteneur.
* **flex-end** : Les éléments s'alignent au bas du conteneur.
* **center** : Les éléments s'alignent au centre vertical du conteneur.
* **baseline** : Les éléments s'alignent à la ligne de base du conteneur.
* **stretch** : Les éléments sont étirés pour s'adapter au conteneur.

#pond {

display: flex;

align-items:flex-end;

}

**flex-direction**. Cette propriété CSS définit la direction dans laquelle les éléments sont placés dans le conteneur, et accepte les valeurs suivantes :

* **row** : Les éléments sont disposés dans la même direction que le texte.
* **row-reverse** : Les éléments sont disposés dans la direction opposée au texte. Remarquez que quand vous mettez la direction inversée, le début et la fin sont aussi inversés.
* **column** : Les éléments sont disposés de haut en bas.
* **column-reverse** : Les éléments sont disposés de bas en haut.

#pond {

display: flex;

flex-direction: row-reverse;

}

Parfois, inverser l'ordre de la rangée ou la colonne ne suffit pas. Dans ces cas, on peut appliquer la propriété **order** à des éléments individuels. Par défaut, les éléments ont une valeur de 0, mais on peut utiliser cette propriété pour changer la valeur à un entier positif ou négatif.

#pond {

display: flex;

}

.yellow {

order:3;

}

Une autre propriété que vous pouvez appliquer sur des éléments individuels est align-self. Cette propriété accepte les mêmes valeurs que align-items, mais s'applique seulement à l'élément ciblé

#pond {

display: flex;

align-items: flex-start;

}

.yellow {

align-self:flex-end;

}

 Distribuez-les en utilisant la propriété **flex-wrap**, qui accepte les valeurs suivantes :

* **nowrap** : Tous les éléments sont tenus sur une seule ligne.
* **wrap** : Les éléments s'enveloppent sur plusieurs lignes au besoin.
* **wrap-reverse** : Les éléments s'enveloppent sur plusieurs lignes dans l'ordre inversé.

#pond {

display: flex;

flex-wrap:wrap;

}

Les deux propriétés **flex-direction** et **flex-wrap** sont utilisées tellement souvent ensembles que le raccourci **flex-flow** a été créé pour les combiner. Ce raccourci accepte les valeurs des deux propriétés séparées par un espace.

Par exemple, vous pouvez utiliser **flex-flow: row wrap** pour configurer les colonnes et les faire s'envelopper.

#pond {

display: flex;

flex-flow:column wrap;

}

Vous pouvez utiliser **align-content** pour définir comment plusieurs lignes sont espacées de l'une à l'autre. Cette propriété prend les valeurs suivantes :

* **flex-start** : Les lignes sont amassées dans le haut du conteneur.
* **flex-end**: Les lignes sont amassées dans le bas du conteneur.
* **center** : Les lignes sont amassées dans le centre vertical du conteneur.
* **space-between** : Les lignes s'affichent avec un espace égal entre eux.
* **space-around** : Les lignes s'affichent avec un espace égal autour d'eux.
* **stretch** : Les lignes sont étirées pour s'adapter au conteneur.

Ceci peut être déroutant, mais **align-content** détermine l'espace entre les lignes, alors que **align-items** détermine comment les éléments dans leur ensemble sont alignées à l'intérieur du conteneur. Quand il n'y a qu'une seule ligne, **align-content** n'a aucun effet.

#pond {

display: flex;

flex-wrap: wrap;

align-content:flex-start;

}