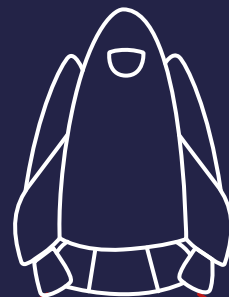


Programa acad mico CAMPUS



**Ciclo 2:
FlexBox**



CSS



CSS the Skin



Flex

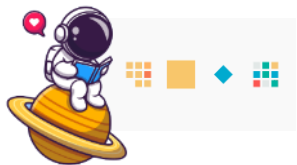


- **Flex** (*también llamado flexbox*) es un sistema de **elementos flexibles** que llega con la idea de olvidar anteriores mecanismos y acostumbrarnos a una mecánica más potente, limpia y personalizable, en la que los elementos HTML se adaptan y colocan automáticamente y es más fácil personalizar los diseños de una página web

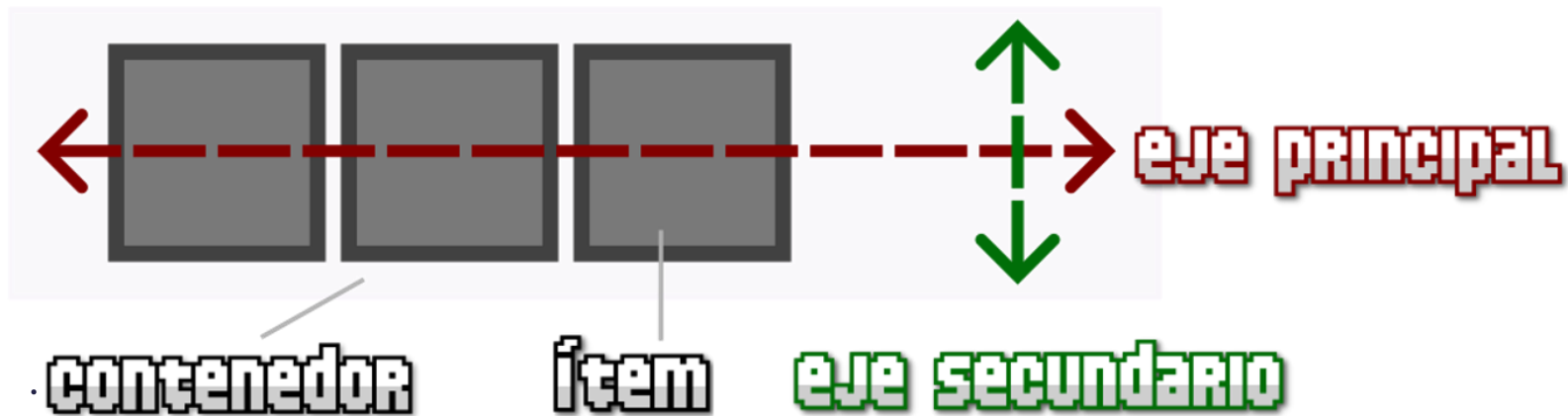
CSS



CSS the Skin



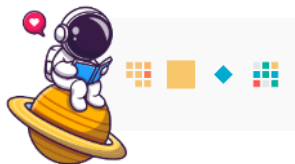
Flex - Conceptos



CSS



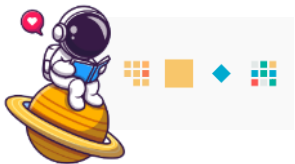
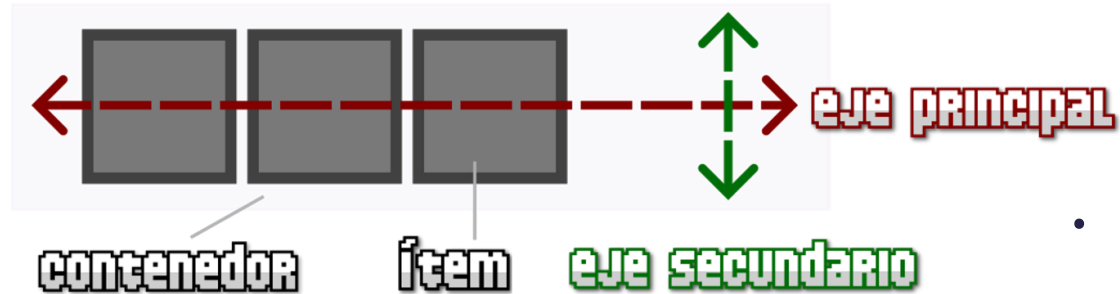
CSS the Siskin



Flex - Conceptos

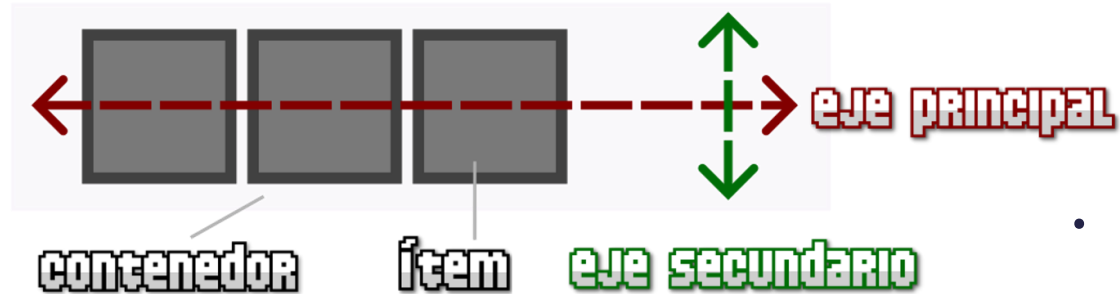
• **Contenedor:** Es el elemento padre que tendrá en su interior cada uno de los ítems flexibles.

• Observa que al contrario que muchas otras estructuras CSS, por norma general, en Flex establecemos las propiedades al elemento padre.



Flex - Conceptos

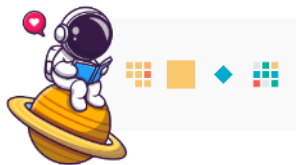
- **Eje principal:** Los contenedores flexibles tendrán una orientación principal específica. Por defecto, el eje principal del contenedor flex es en **horizontal** (*en fila*).



CSS

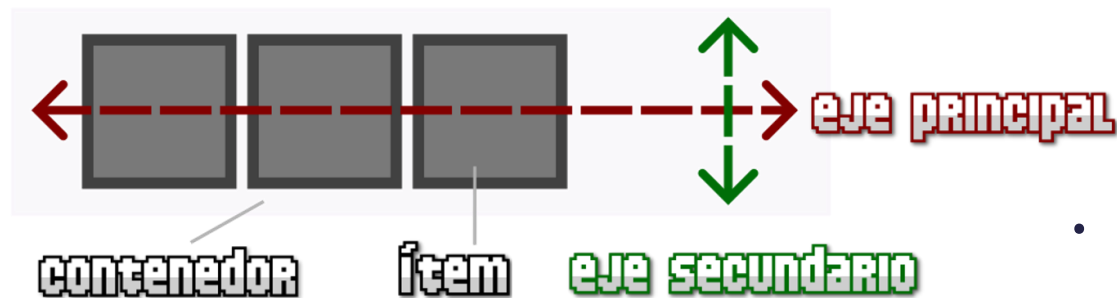


CSS the Siskin



Flex - Conceptos

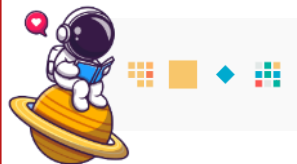
- **Eje secundario:** De la misma forma, los contenedores flexibles tendrán una orientación secundaria, perpendicular a la principal. Si la principal es en horizontal, la secundaria será en **vertical (y viceversa)**.



CSS

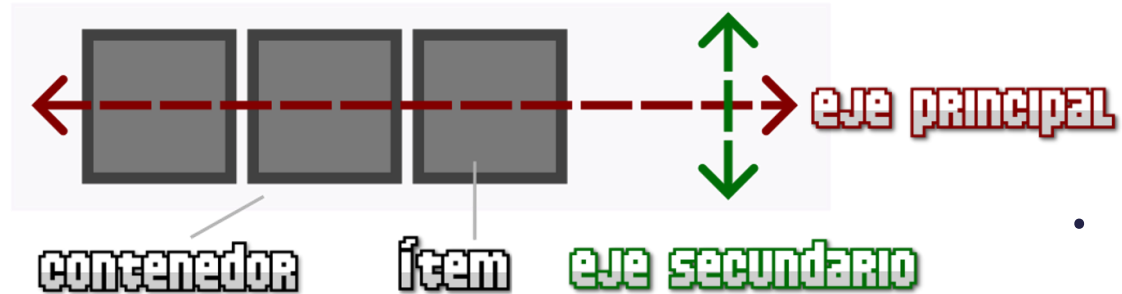


CSS the Siskin



Flex - Conceptos

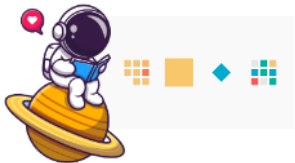
- **Ítem:** Cada uno de los **hijos** que tendrá el contenedor en su interior.



CSS



CSS the Siskin



Flex – HTML Cajas



```
<h1>Aprendiendo Flexbox</h1>
```

```
<section class="contenedor">
```

```
<div class="item">Item 1</div>
```

```
<div class="item">Item 2</div>
```

```
<div class="item">Item 3</div>
```

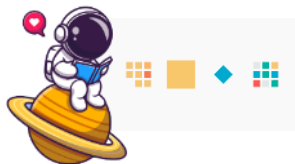
```
<div class="item">Item 4</div>
```

```
</section>
```

CSS



CSS the Siskin



Flex - CSS

```
body {  
  background-color: beige;  
}  
  
.item {  
  border: 1px solid rgb(20, 69, 69);  
  background-color: white;  
  color: black;  
  font-family: 'Courier New',  
  Courier, monospace;  
  font-weight: bold;  
  width: 120px;  
  height: 120px;  
  line-height: 120px;  
  text-align: center;  
  margin: 5px;  
}
```



Aprendiendo Flexbox

Item 1

Item 2

Item 3

Item 4



CSS



CSS the Siskin

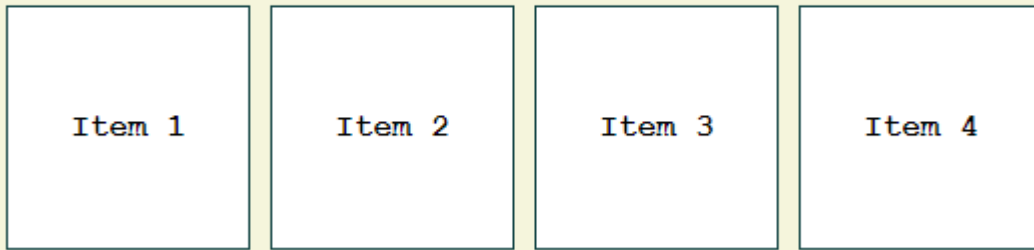


Flex – display:flex



```
.contenedor {  
  display: flex;  
}
```

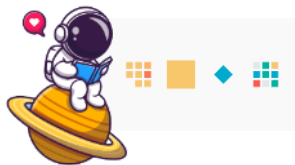
Aprendiendo Flexbox



CSS



CSS the Siskin

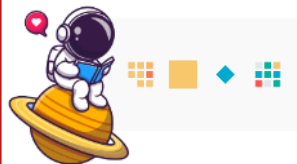
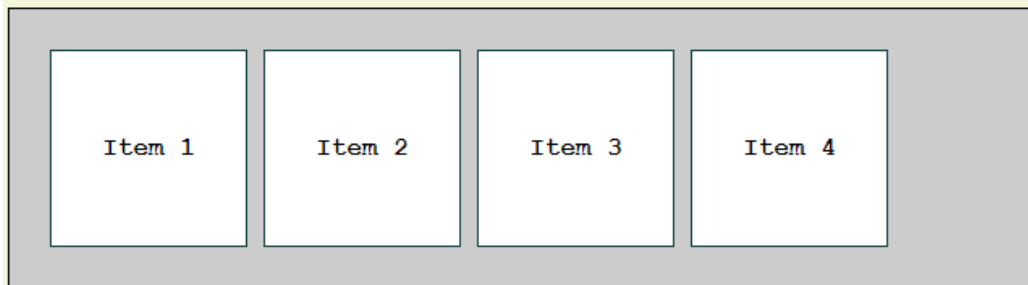


Flex – Enmarcando el Flex



```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
}
```

Aprendiendo Flexbox



CSS



CSS the Side



Flex – flex-direction

```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  flex-direction: column;  
}
```

Aprendiendo Flexbox

Item 1

Item 2

Item 3

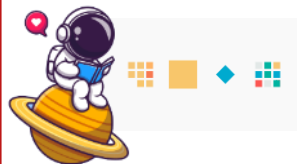
Item 4



CSS



CSS the Siskin



Flex – flex-direction

```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  flex-direction: column-reverse;  
}
```

Aprendiendo Flexbox

Item 4

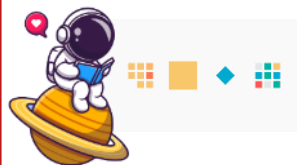
Item 3

Item 2

Item 1



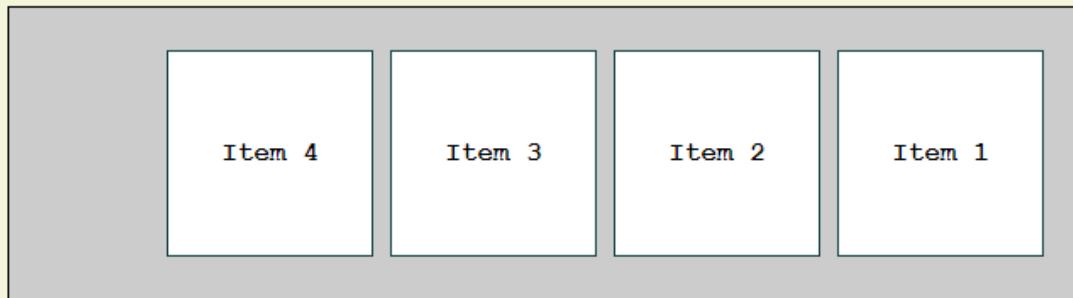
CSS the Siskin



Flex – flex-direction

```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  flex-direction: row-reverse;  
}
```

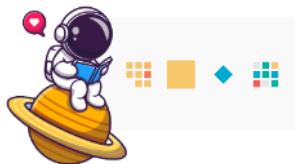
Aprendiendo Flexbox



CSS



CSS the Side



Flex – flex-direction – agregando más cajas



```
<section class="contenedor">
  <div class="item">Item 1</div>
  <div class="item">Item 2</div>
  <div class="item">Item 3</div>
  <div class="item">Item 4</div>
```

```
... <div class="item">Item 1</div>
... <div class="item">Item 2</div>
... <div class="item">Item 3</div>
... <div class="item">Item 4</div>
```

```
... <div class="item">Item 1</div>
... <div class="item">Item 2</div>
... <div class="item">Item 3</div>
... <div class="item">Item 4</div>
```

```
... <div class="item">Item 1</div>
... <div class="item">Item 2</div>
... <div class="item">Item 3</div>
... <div class="item">Item 4</div>
</section>
```

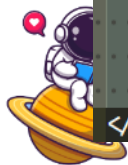
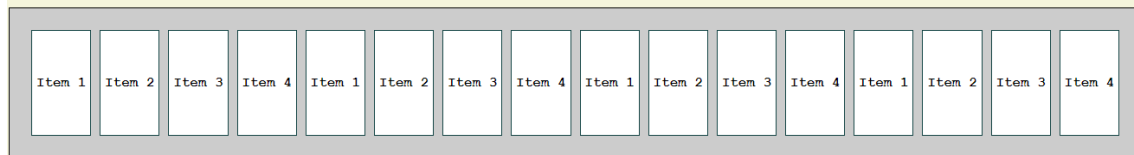
```
.contenedor {
  border: 1px solid black;
  background-color: #ccc;
  padding: 20px;
  display: flex;
  flex-direction: row;
}
```

CSS



CSS the Side

Aprendiendo Flexbox



Flex – flex-direction – redimensionando y centrando



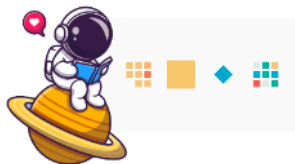
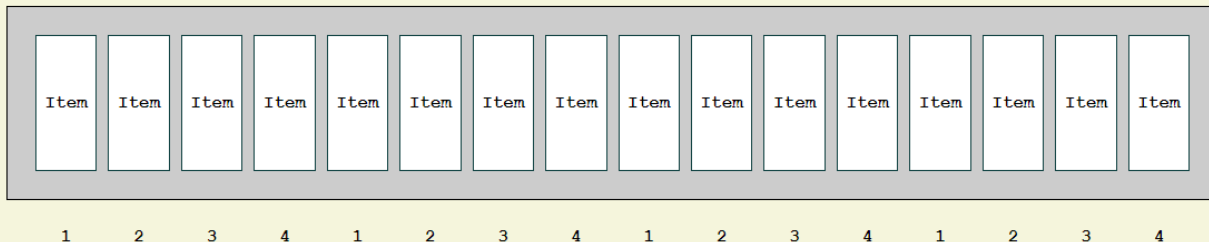
```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  flex-direction: row;  
  width: 80%;  
  margin: 0 auto;  
}
```

Elementos se reducen y se ajustan todos al mismo tamaño para caber en el contenedor.

CSS



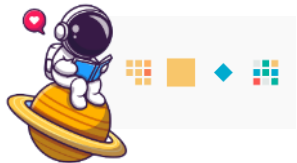
Aprendiendo Flexbox



Flex – flex-wrap

```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  flex-direction: row;  
  width: 80%;  
  margin: 0 auto;  
  flex-wrap: wrap;  
}
```

Aprendiendo Flexbox



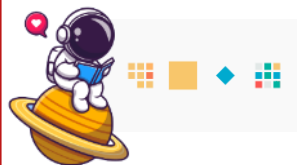
Flex – flex-flow



```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap;  
}
```

Combina el flex-direction y el flex-wrap.

flexbox



SS



the Sidn

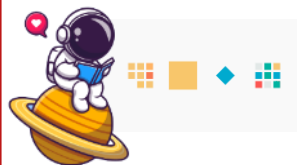
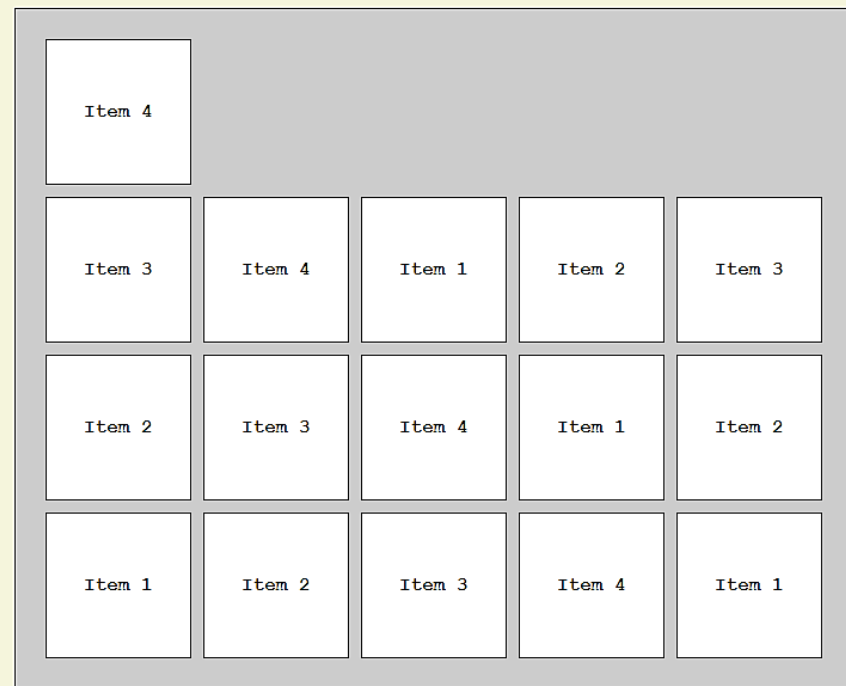


Flex – flex-flow : wrap-reverse



```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
}
```

Aprendiendo Flexbox



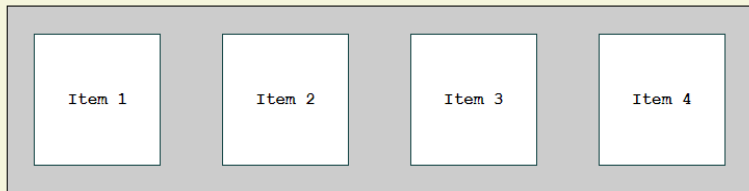
Flex – quitar cajas y justify-content:space-between



```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: space-between;  
}
```

```
<section class="contenedor">  
  <div class="item">Item 1</div>  
  <div class="item">Item 2</div>  
  <div class="item">Item 3</div>  
  <div class="item">Item 4</div>  
</section>
```

Aprendiendo Flexbox



Distribuye el espacio de manera uniforme entre los elementos de un contenedor, dejando espacio en blanco tanto antes del primer elemento como después del último elemento.

Ocupan el 100% del espacio con una separación entre ellos.

CSS



CSS the Siskin



Flex – justify-content:space-around



```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: space-around;  
}
```

Distribuye el espacio alrededor de los elementos hijos de un contenedor, creando un espaciado uniforme entre ellos y en los extremos.

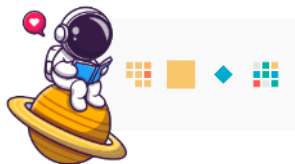
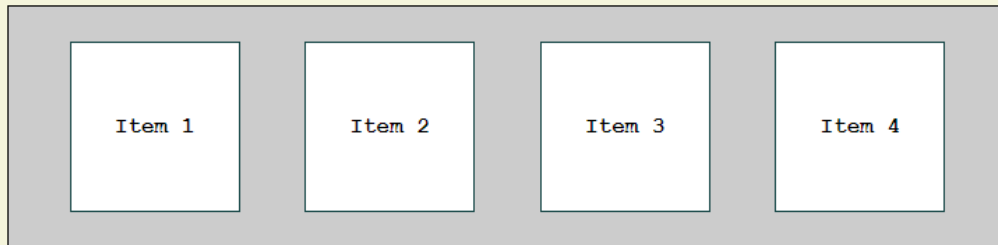
CSS



CSS the Siskin



Aprendiendo Flexbox



Flex – justify-content:space-evenly



```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: space-evenly;  
}
```

Distribuye el espacio entre los elementos de un contenedor, y alrededor del primer y último elemento, de manera uniforme, creando un espaciado equitativo.

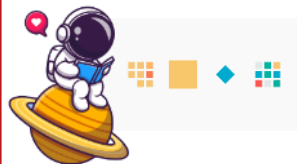
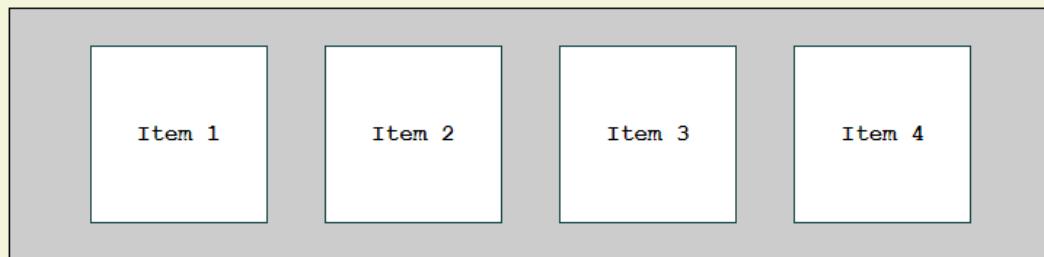
CSS



CSS the Siskin

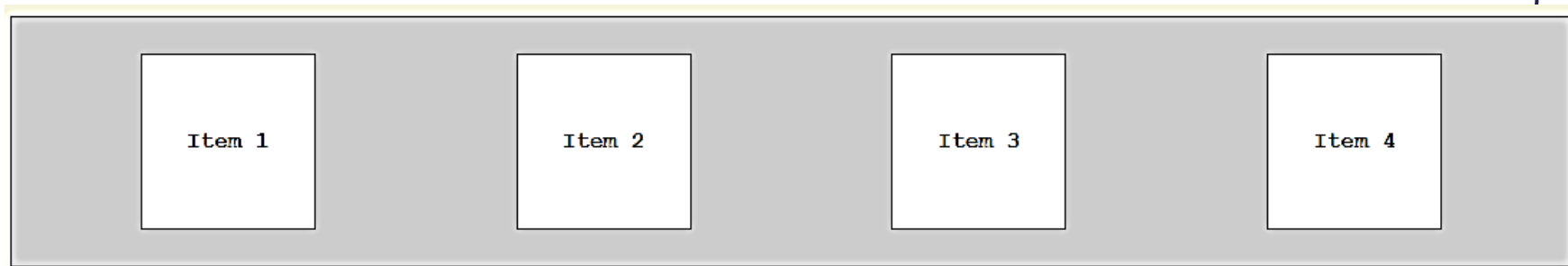


Aprendiendo Flexbox

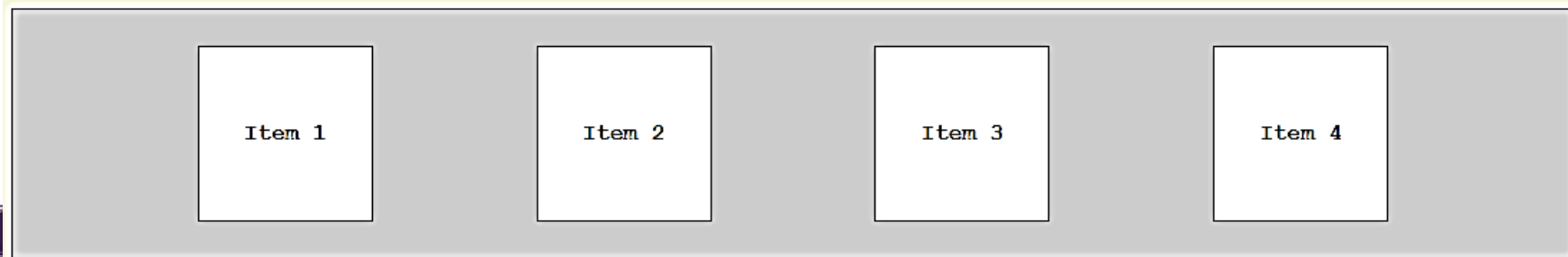


Flex – space-around vs space-evenly

space-around



space-evenly



Flex – space-around vs space-evenly

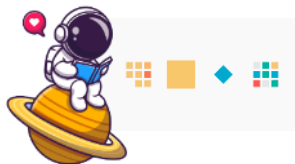


En resumen, **space-evenly** distribuye el espacio de manera uniforme entre elementos y en los extremos, mientras que **space-around** distribuye el espacio de manera uniforme alrededor de cada elemento, incluidos los extremos.

CSS



CSS the Siskin



Flex – justify-content:center

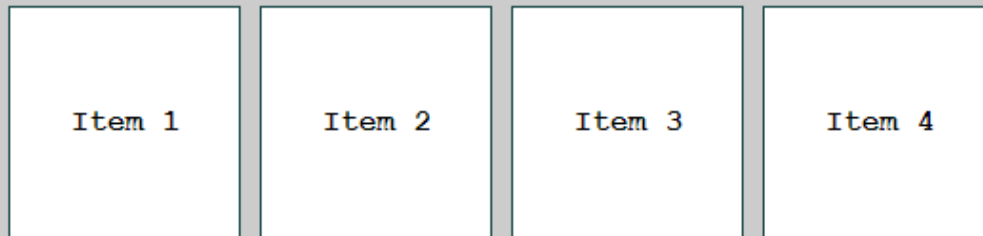
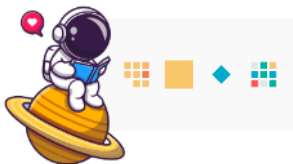


```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: center;  
}
```

CSS



CSS the Side



Flex – justify-content:flex-start

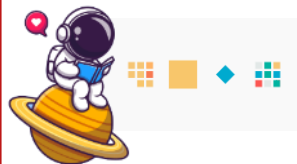


```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: flex-start;  
}
```

CSS



ne Sidn



Item 1

Item 2

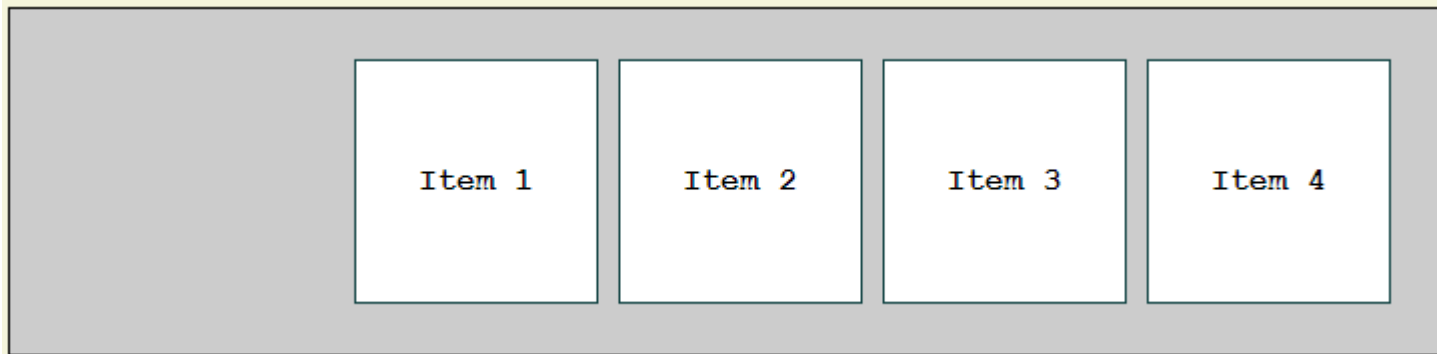
Item 3

Item 4

Flex – justify-content:flex-end



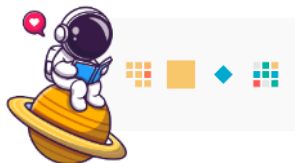
```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: flex-end;  
}
```



CSS



the Side



Flex – altura en el contenedor



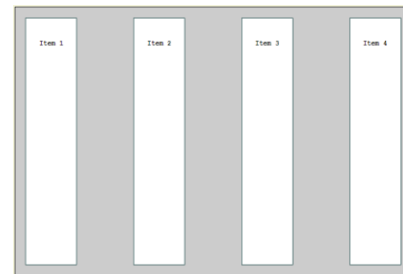
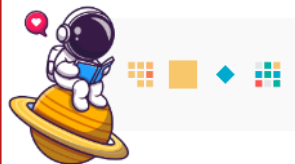
```
.item {  
  border: 1px solid rgb(20, 69, 69);  
  background-color: white;  
  color: black;  
  font-family: 'Courier New',  
  Courier, monospace;  
  font-weight: bold;  
  width: 120px;  
  /*height: 120px;*/  
  line-height: 120px;  
  text-align: center;  
  margin: 5px;  
}
```

```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: space-between;  
  height: 600px;  
}
```

CSS



CSS the Siskin



Flex – Quitar la altura en el contenedor y se ajustan las alturas de las cajas

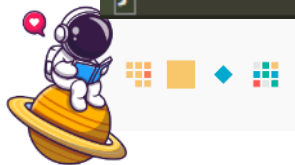
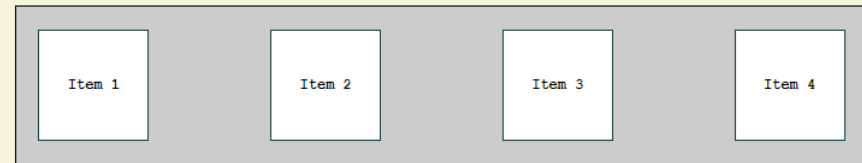


```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  flex-flow: row wrap-reverse;  
  justify-content: space-between;  
}
```

CSS



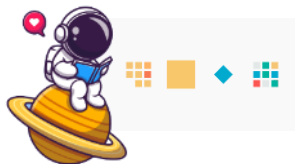
Aprendiendo Flexbox



Flex – align-items: flex-end

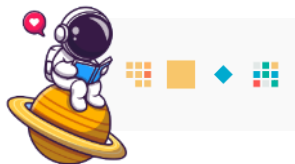


```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: space-between;  
  flex-flow: row wrap;  
  height: 600px;  
  align-items: flex-end;  
}
```



Flex – align-items: flex-start

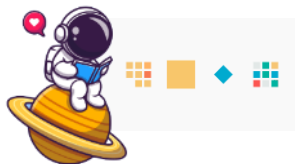
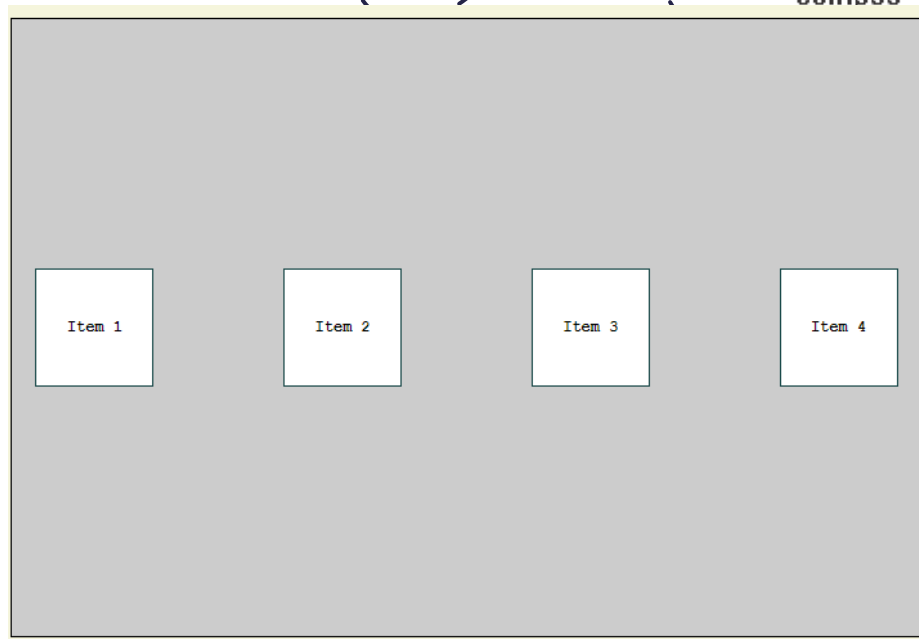
```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: space-between;  
  flex-flow: row wrap;  
  height: 600px;  
  align-items: flex-start;  
}
```



Flex – align-items: flex-center



```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: space-between;  
  flex-flow: row wrap;  
  height: 600px;  
  align-items: center;  
}
```



Flex – align-items: flex-center – agreguemos más cajas

```
<div class="item">Item 1</div>  
<div class="item">Item 2</div>  
<div class="item">Item 3</div>  
<div class="item">Item 4</div>
```

```
<div class="item">Item 1</div>  
<div class="item">Item 2</div>  
<div class="item">Item 3</div>  
<div class="item">Item 4</div>
```

```
<div class="item">Item 1</div>  
<div class="item">Item 2</div>  
<div class="item">Item 3</div>  
<div class="item">Item 4</div>
```

```
<div class="item">Item 1</div>  
<div class="item">Item 2</div>  
<div class="item">Item 3</div>  
<div class="item">Item 4</div>
```



CSS



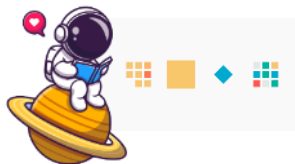
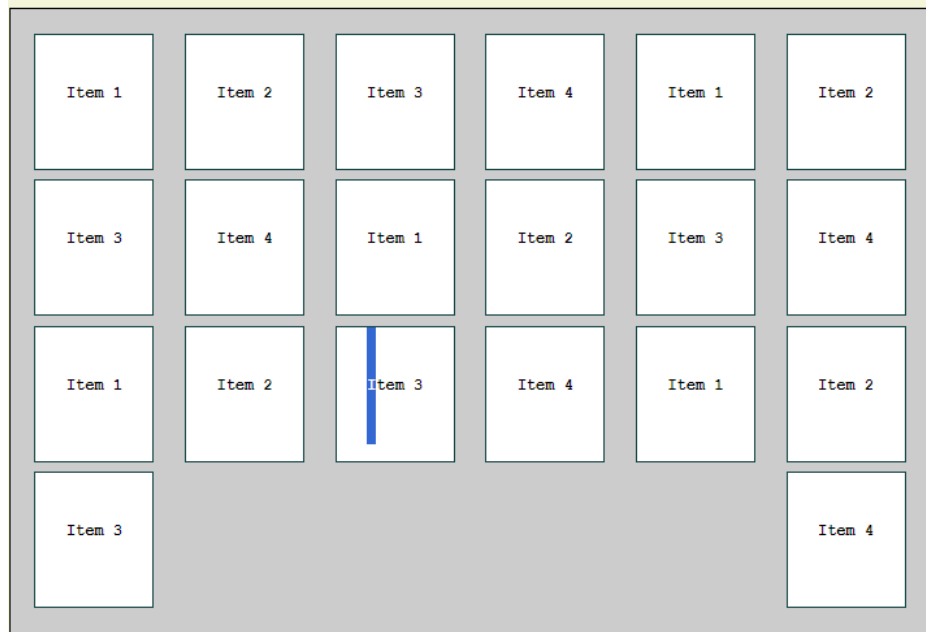
5 the Side



Flex – si quitamos el flex-center

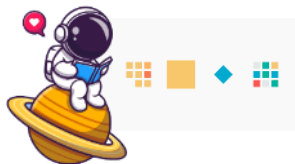
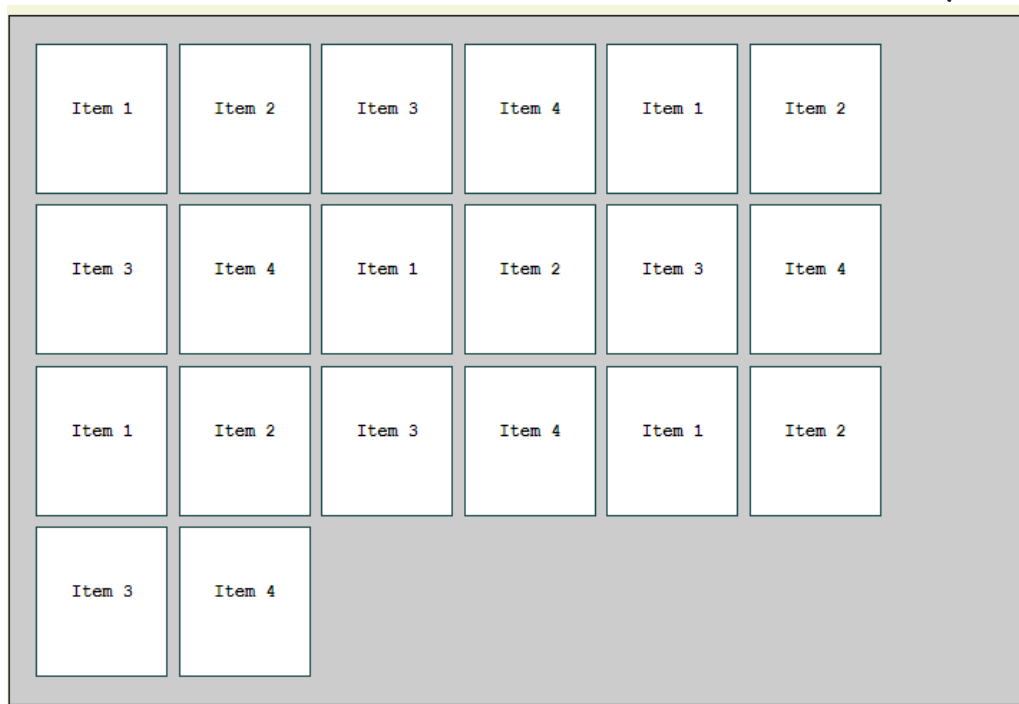


```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: space-between;  
  flex-flow: row wrap;  
  height: 600px;  
}
```



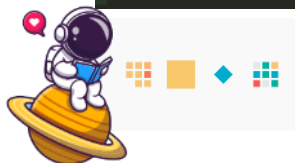
Flex – si dejamos justify-content: flex-start;

```
.contenedor {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: flex-start;  
  flex-flow: row wrap;  
  height: 600px;  
}
```



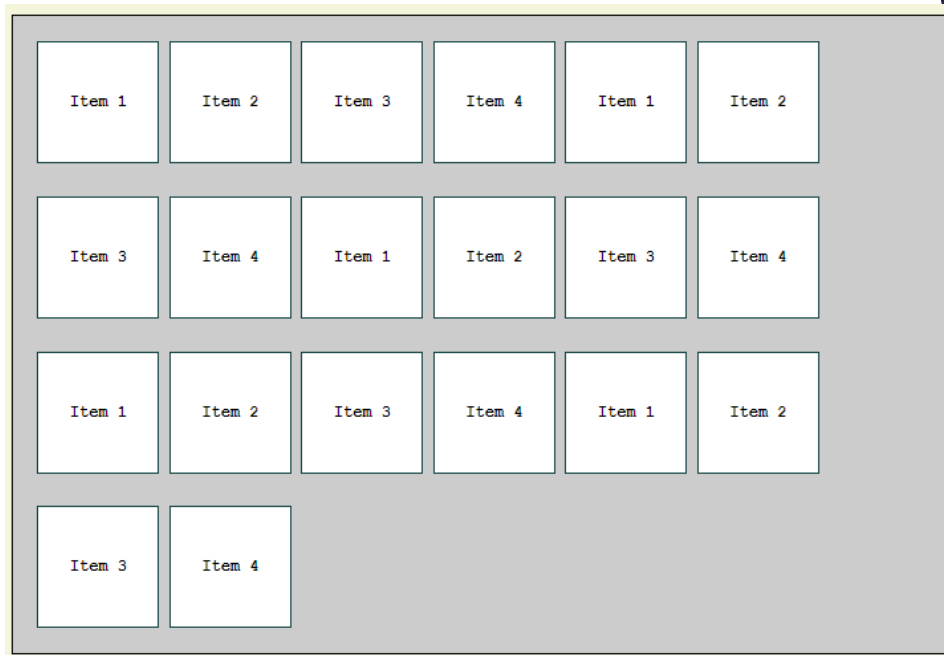
Flex – align-content:center;

```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: flex-start;  
  flex-flow: row wrap;  
  height: 600px;  
  align-content: center;  
}
```



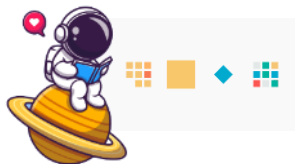
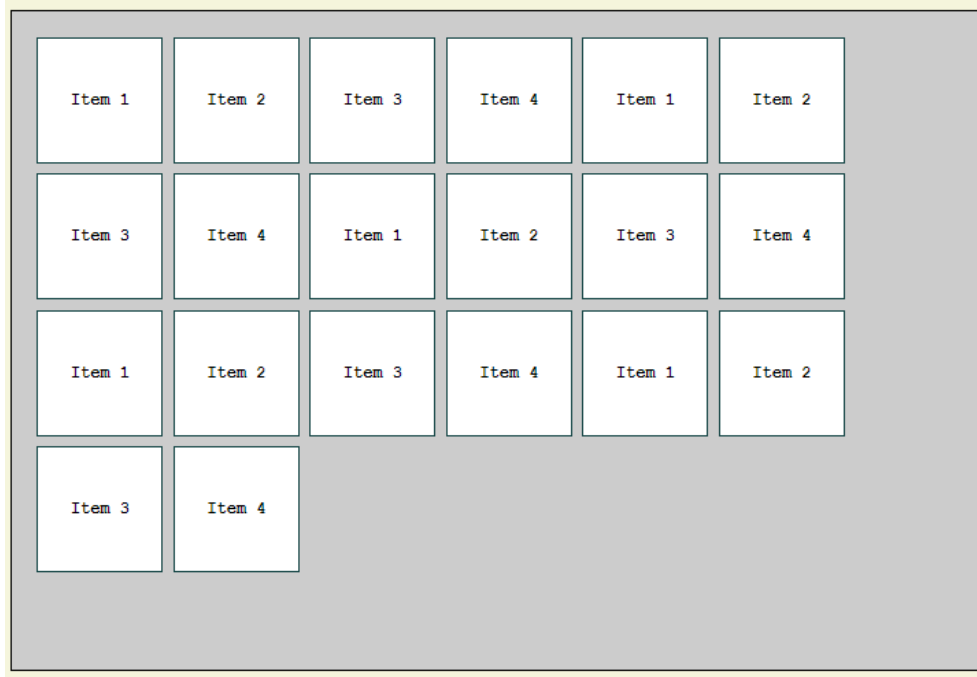
Flex – align-content: space-between

```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: flex-start;  
  flex-flow: row wrap;  
  height: 600px;  
  align-content: space-between;  
}
```



Flex – align-content: flex-start

```
.container {  
  border: 1px solid black;  
  background-color: #ccc;  
  padding: 20px;  
  display: flex;  
  width: 80%;  
  margin: 0 auto;  
  justify-content: flex-start;  
  flex-flow: row wrap;  
  height: 600px;  
  align-content: flex-start;  
}
```



Flex – borramos las cajas y dejamos 3 y colocamos clases diferentes a cada una

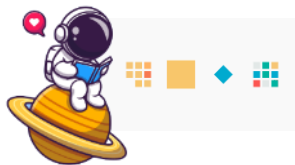
```
<section class="contenedor">
  <div class="item i1">Item 1</div>
  <div class="item i2">Item 2</div>
  <div class="item i3">Item 3</div>
  <div class="item i4">Item 4</div>
</section>
```



CSS



CSS the Siskin



Flex – Estilos a cada caja



```
flex-flow: row wrap;
```

```
height: 600px;
```

```
justify-content: center;
```

```
align-content: center;
```

```
}
```

```
.i1 {
```

```
background-color: lightblue;
```

```
}
```

```
.i2 {
```

```
background-color: lightcoral;
```

```
}
```

```
.i3 {
```

```
background-color: lightgreen;
```

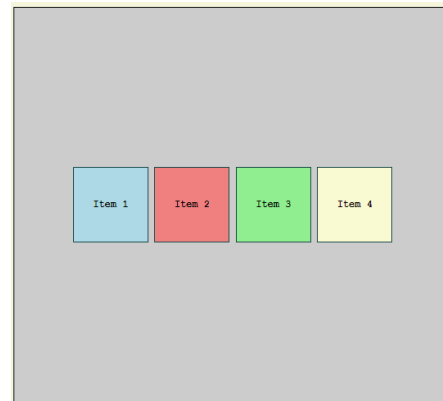
```
}
```

```
.i4 {
```

```
background-color: lightgoldenrodyellow;
```

```
}
```

```
.item {  
  border: 1px solid rgb(20, 69, 69);  
  background-color: white;  
  color: black;  
  font-family: 'Courier New', Courier, monospace;  
  font-weight: bold;  
  width: 200px;  
  /*height: 120px;*/  
  line-height: 200px;  
  text-align: center;  
  margin: 5px;  
}
```



CSS



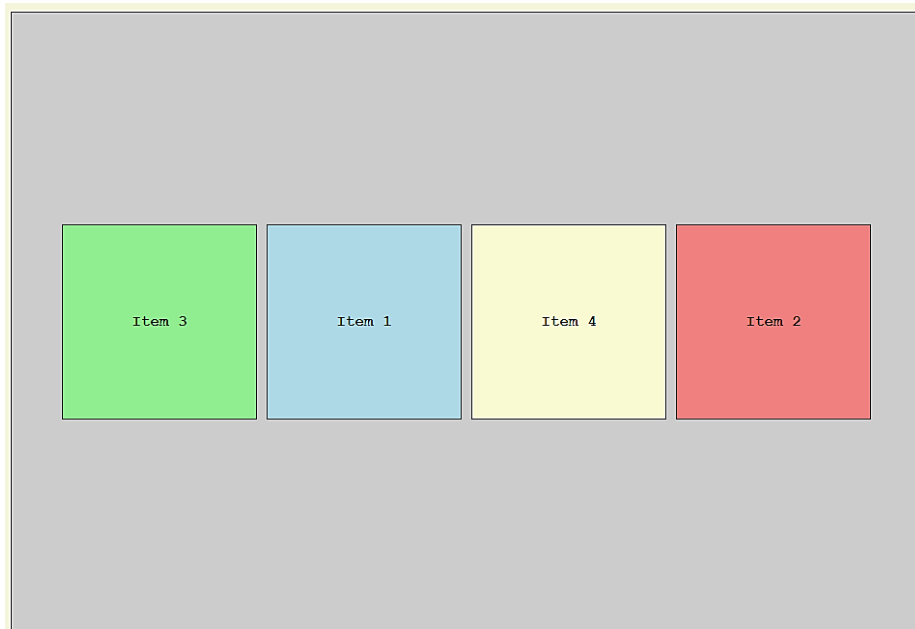
CSS the Squirrel



Flex – order - Ordenando las cajas propiedad

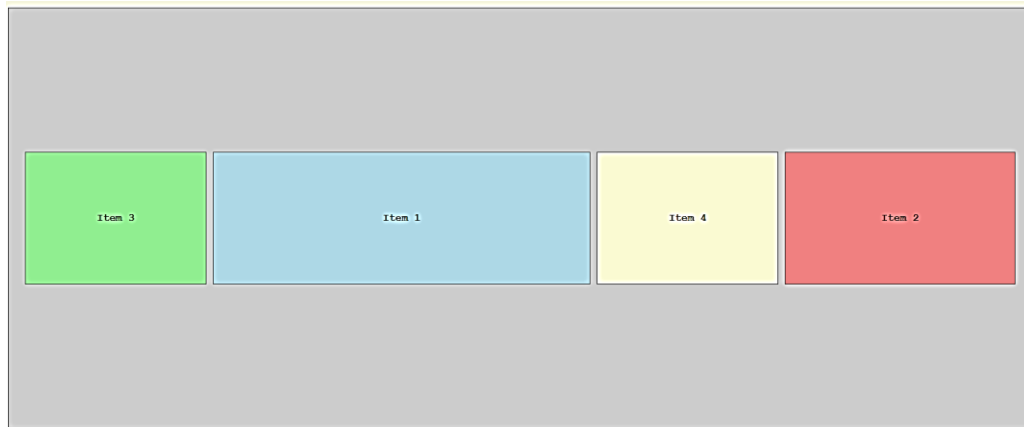


```
.i1 {  
  background-color: lightblue;  
  order: 2;  
}  
  
.i2 {  
  background-color: lightcoral;  
  order: 4;  
}  
  
.i3 {  
  background-color: lightgreen;  
  order: 1;  
}  
  
.i4 {  
  background-color: lightgoldenrodyellow;  
  order: 3;  
}
```

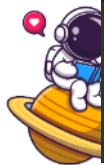


Flex – flex-grow – Priorizando el ancho de los elementos.

```
.i1 {  
  background-color: lightblue;  
  order: 2;  
  flex-grow: 5;  
}  
  
.i2 {  
  background-color: lightcoral;  
  order: 4;  
  flex-grow: 2;  
}  
  
.i3 {  
  background-color: lightgreen;  
  order: 1;  
  flex-grow: 1;  
}  
  
.i4 {  
  background-color: lightgoldenrodyellow;  
  order: 3;  
  flex-grow: 1;  
}
```



CSS the Side



Flex – flex-grow – Priorizando el ancho de los elementos.



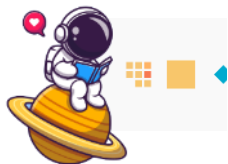
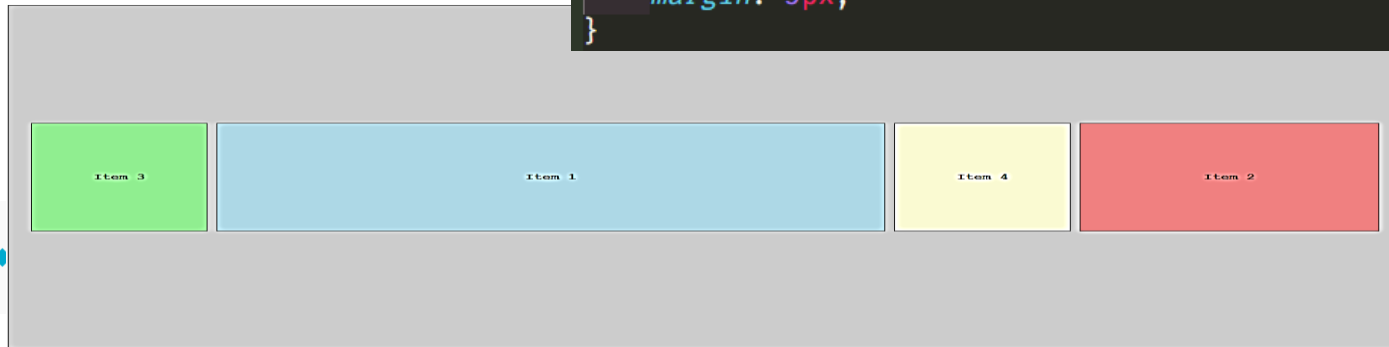
- Quita el width de los ítem y observa el resultado de afectar el ancho de los elementos con **flex-grow**.

```
.item {  
  border: 1px solid rgb(20, 69, 69);  
  background-color: white;  
  color: black;  
  font-family: 'Courier New', Courier, monospace;  
  font-weight: bold;  
  /*width: 200px;*/  
  /*height: 120px;*/  
  line-height: 200px;  
  text-align: center;  
  margin: 5px;  
}
```

CSS



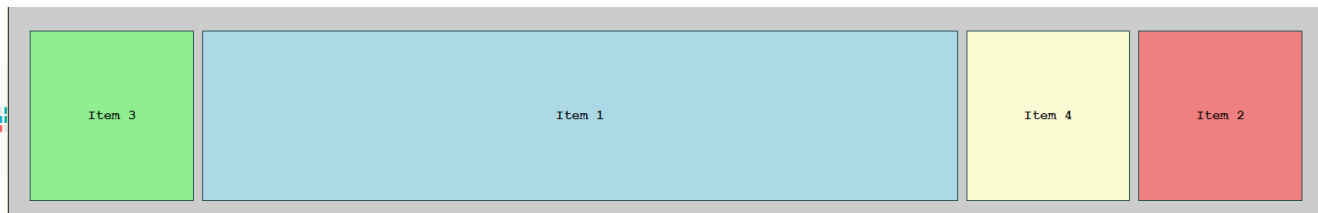
CSS the Siskin



Flex – flex-basis – ancho de los elementos.

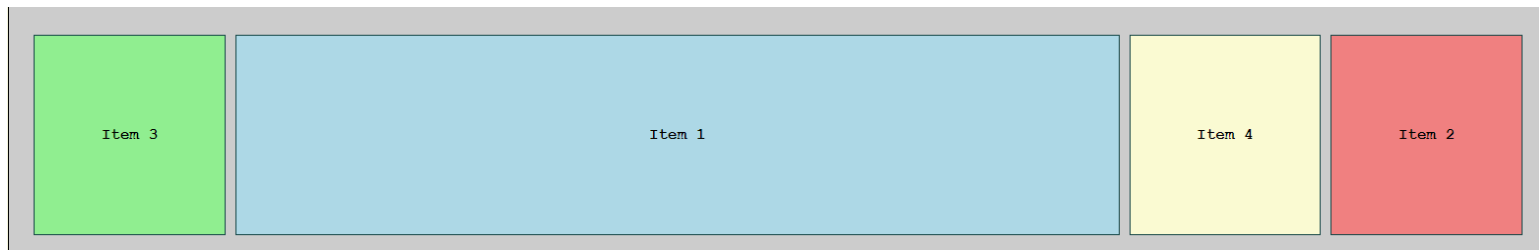
Para tener un 50% real de width, hacer flex-grow en 1, en todos los ítems, y **flex-basis de un 50%.**

```
.i1 {  
  background-color: lightblue;  
  order: 2;  
  flex-grow: 1;  
  flex-basis: 50%;  
}  
  
.i2 {  
  background-color: lightcoral;  
  order: 4;  
  flex-grow: 1;  
}
```



Flex – flex-basis – ancho de los elementos.

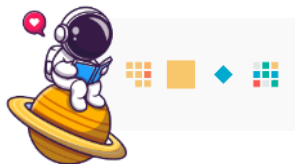
Se usa más el **flex-basis** que el **width** ya que tiene mejor resultado con las propiedades de Flexbox

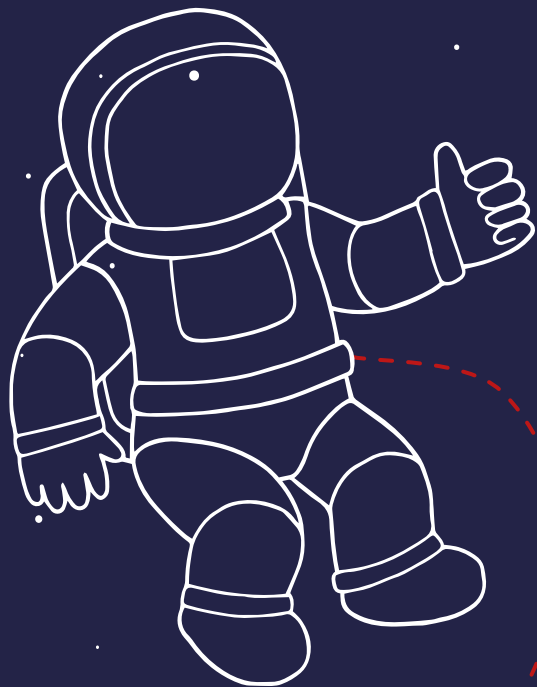


CSS



CSS the Siskin





Programa acad mico CAMPUS

