

# FLEXBOX & GRID

— GRID AREAS —



**IN A ROCKET**

*Learn front-end development at rocket speed*

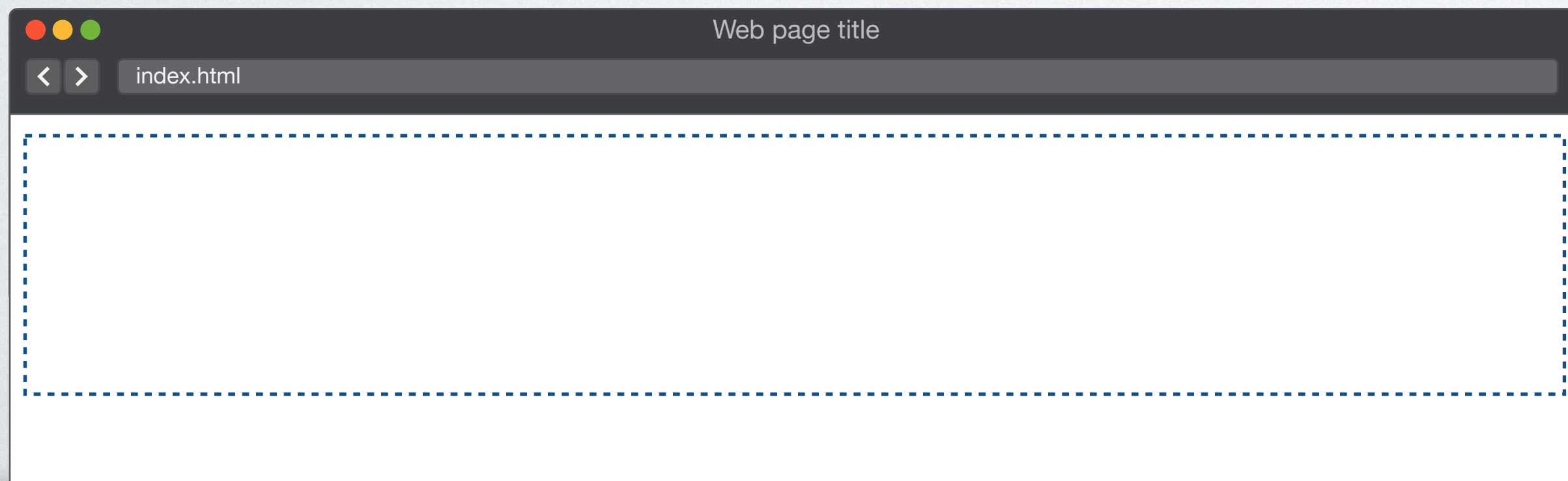
# GRID AREAS

## HTML

```
<body>
<div class="container">
  <div class="i1">Item1</div>
  <div class="i2">Item2</div>
  <div class="i3">Item3</div>
</div>
</body>
```

## CSS

## Browser



# GRID AREAS

## HTML

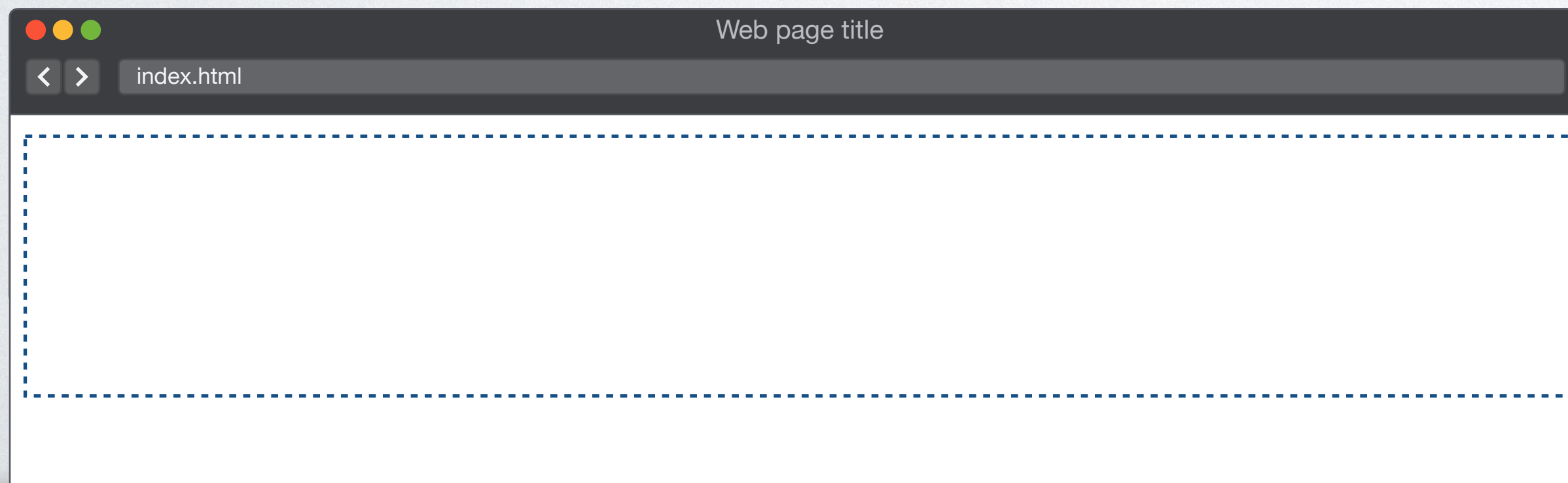
```
<body>
<div class="container">
  <div class="i1">Item1</div>
  <div class="i2">Item2</div>
  <div class="i3">Item3</div>
</div>
</body>
```

## CSS

```
.i1 { grid-area: navigation; }
.i2 { grid-area: content; }
.i3 { grid-area: promos; }

.container {
  display: grid;
  grid-template-columns: repeat(3, 1fr);
}
```

## Browser



## HTML

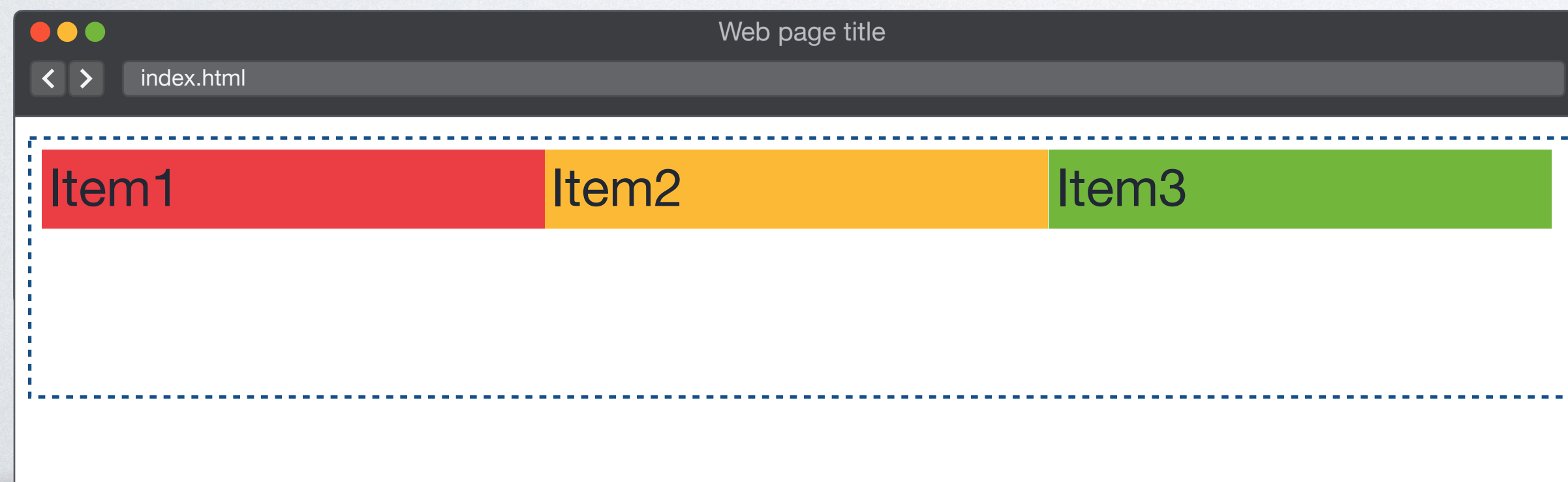
```
<body>
<div class="container">
  <div class="i1">Item1</div>
  <div class="i2">Item2</div>
  <div class="i3">Item3</div>
</div>
</body>
```

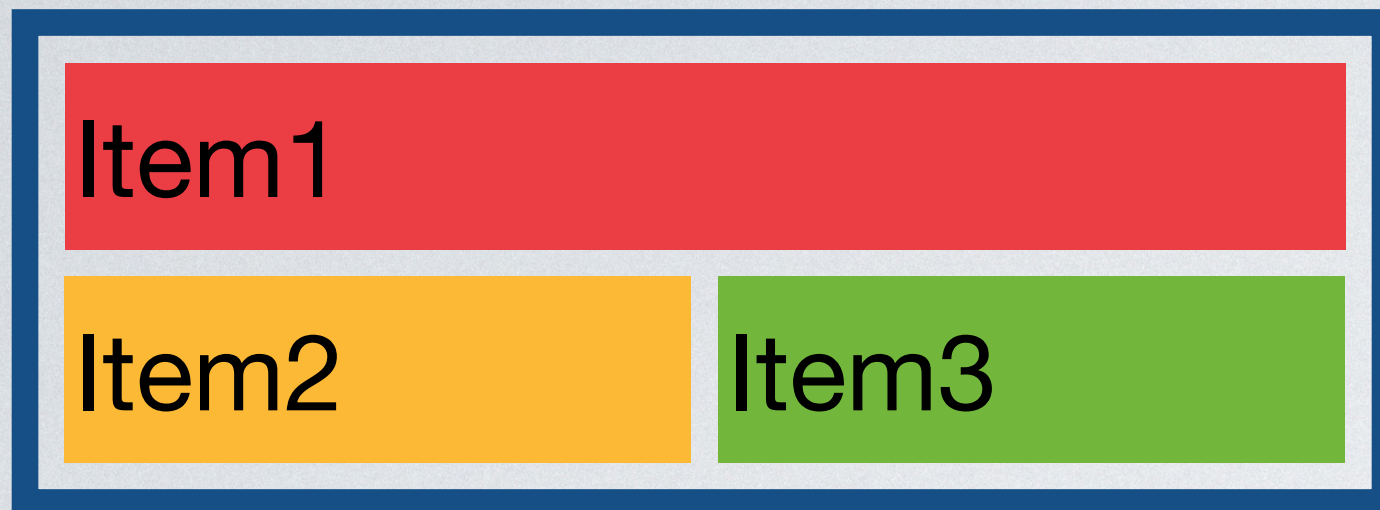
## CSS

```
.i1 { grid-area: navigation; }
.i2 { grid-area: content; }
.i3 { grid-area: promos; }

.container {
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  grid-template-areas: "navigation content promos";
}
```

## Browser





## CSS

```
.i1 { grid-area: item1; }  
.i2 { grid-area: item2; }  
.i3 { grid-area: item3; }  
  
.container {  
  display: grid;  
  grid-template: repeat(2, 1fr) / repeat(2, 1fr);  
  grid-template-areas: "item1 item1"  
                      "item2 item3";  
}
```

## HTML

```
<body>  
<div class="container">  
  <div class="i1">Item1</div>  
  <div class="i2">Item2</div>  
  <div class="i3">Item3</div>  
</div>  
</body>
```



## CSS

```
.i1 { grid-area: item1; }  
.i2 { grid-area: item2; }  
.i3 { grid-area: item3; }  
.i4 { grid-area: item3; }  
  
.container {  
  display: grid;  
  grid-template: repeat(2, 1fr) / repeat(2, 1fr);  
  grid-template-areas: "item2 item3"  
                      "item4 item1";  
}
```

## HTML

```
<body>  
<div class="container">  
  <div class="i1">Item1</div>  
  <div class="i2">Item2</div>  
  <div class="i3">Item3</div>  
  <div class="i4">Item4</div>  
</div>  
</body>
```

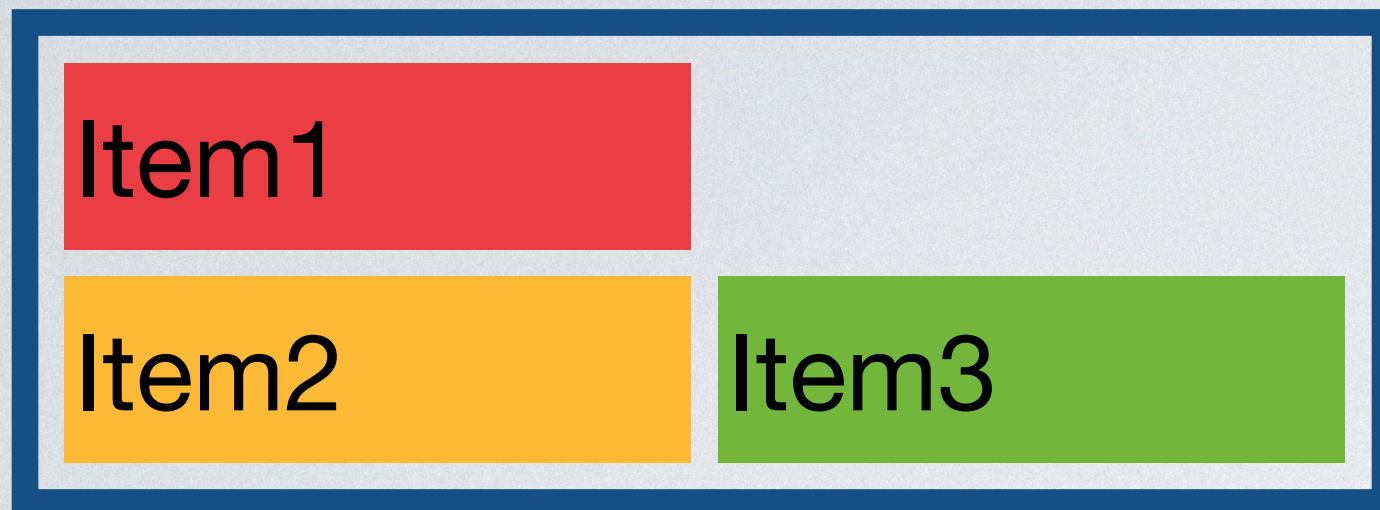


## CSS

```
.i1 { grid-area: item1; }  
.i2 { grid-area: item2; }  
.i3 { grid-area: item3; }  
  
.container {  
  display: grid;  
  grid-template: repeat(2, 1fr) / repeat(2, 1fr);  
  grid-template-areas: "item2 item3"  
                      "item1 item1";  
}
```

## HTML

```
<body>  
<div class="container">  
  <div class="i1">Item1</div>  
  <div class="i2">Item2</div>  
  <div class="i3">Item3</div>  
</div>  
</body>
```



## CSS

```
.i1 { grid-area: item1; }  
.i2 { grid-area: item2; }  
.i3 { grid-area: item3; }  
  
.container {  
  display: grid;  
  grid-template: repeat(2, 1fr) / repeat(2, 1fr);  
  grid-template-areas: "item1 ."  
                      "item2 item3";  
}
```

## HTML

```
<body>  
<div class="container">  
  <div class="i1">Item1</div>  
  <div class="i2">Item2</div>  
  <div class="i3">Item3</div>  
</div>  
</body>
```



# FLEXBOX & GRID

— GRID AREAS —



**IN A ROCKET**

*Learn front-end development at rocket speed*