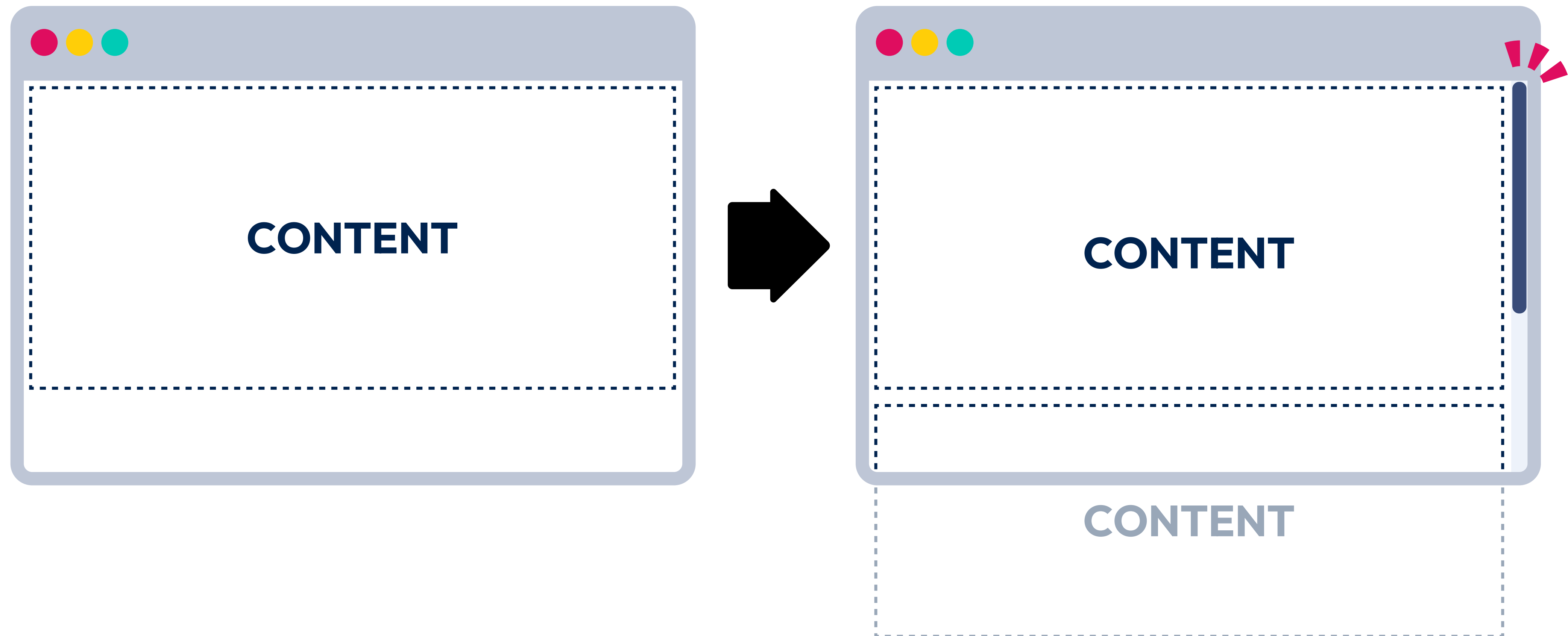


CSS *だけで実装する* パララックススクロール

【解説資料】

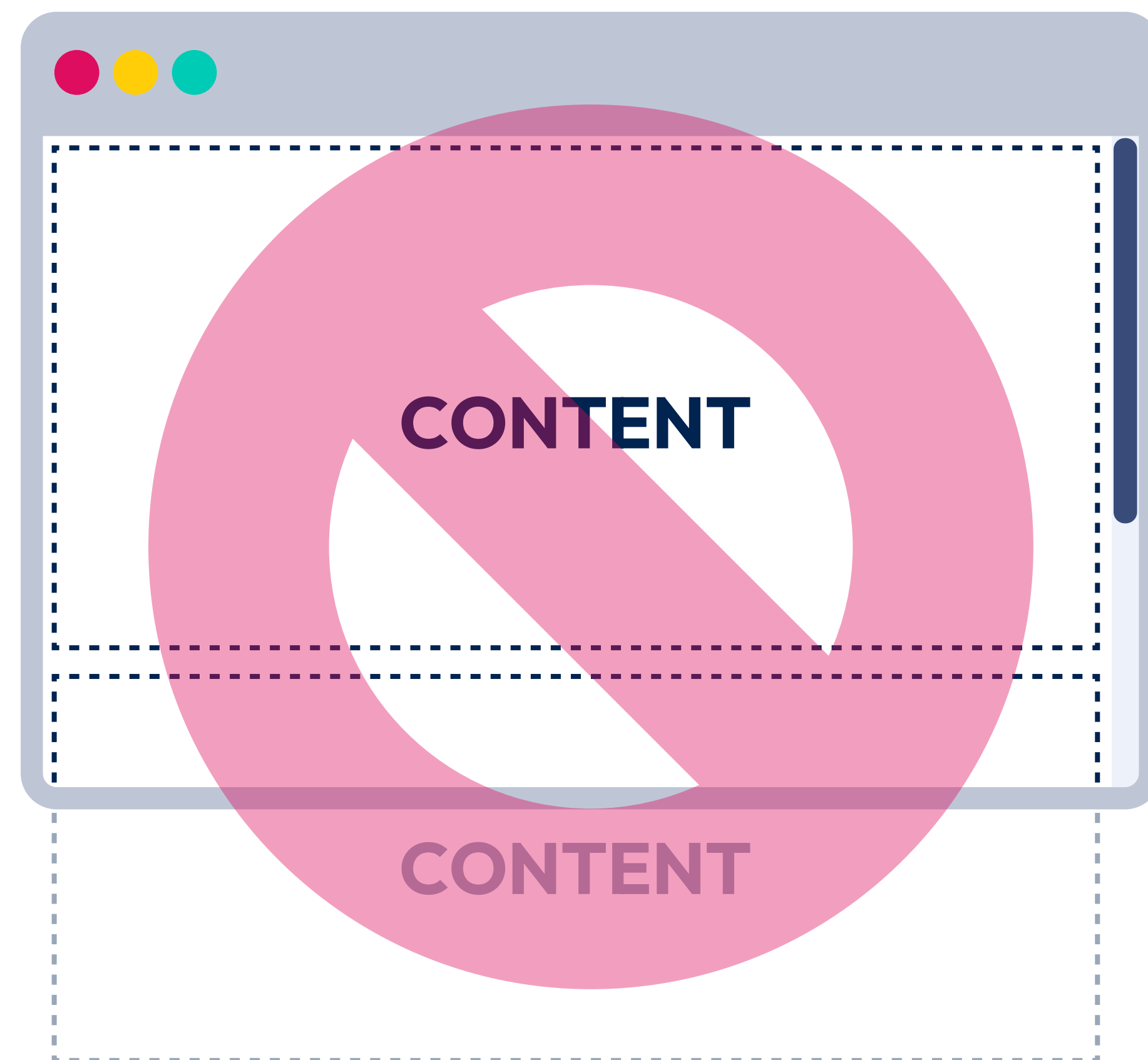
ソースコードと併せてご覧ください。

通常、コンテンツの高さがブラウザの高さを超えると、
ブラウザ（<body>）にスクロールバーが出現します。




ですが、この `<body>` のスクロールバーだと
パララックススクロールを実現できません。

理由はのちほど



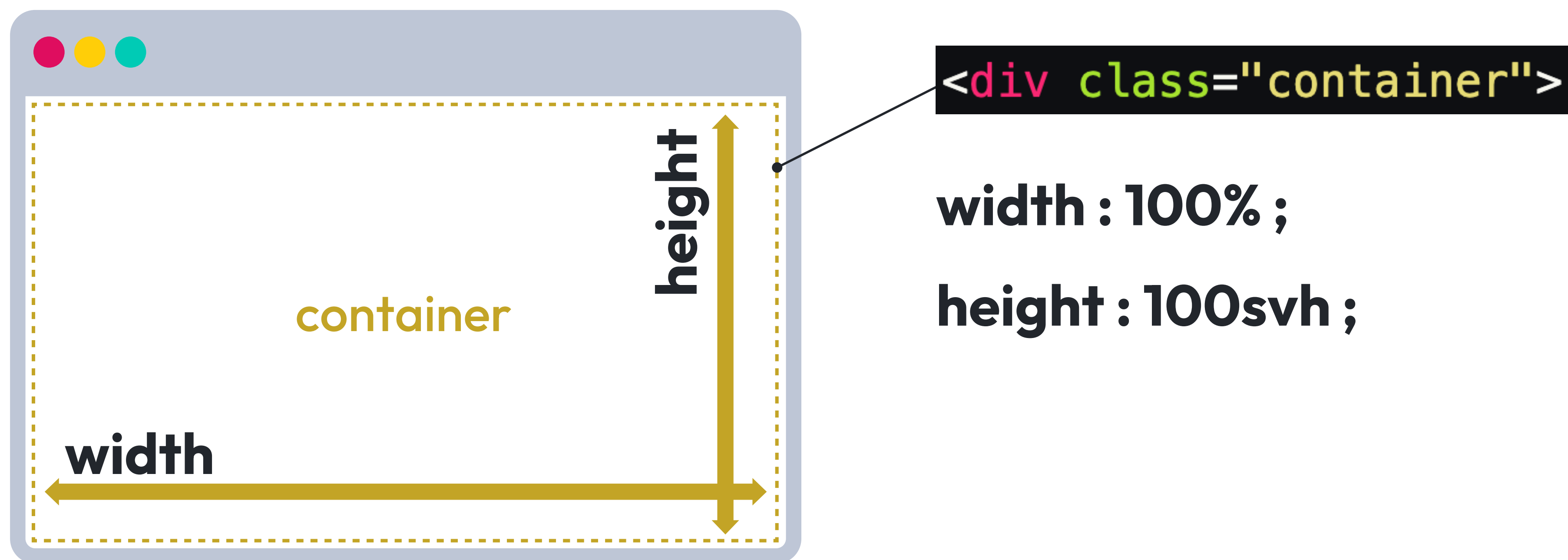
パララックス実装のポイントは、
<body> 直下にあるこの「container」です。

```
<body>  
  <div class="container">  
    <div class="parallax">  
      <div class="element_A">A</div>  
      <div class="element_B">B</div>  
      <div class="element_C">C</div>  
    </div>  
    <!-- ↓これは必須じゃない（下に続くセクション） -->  
    <div class="otherSection"></div>  
  </div>  
</body>
```

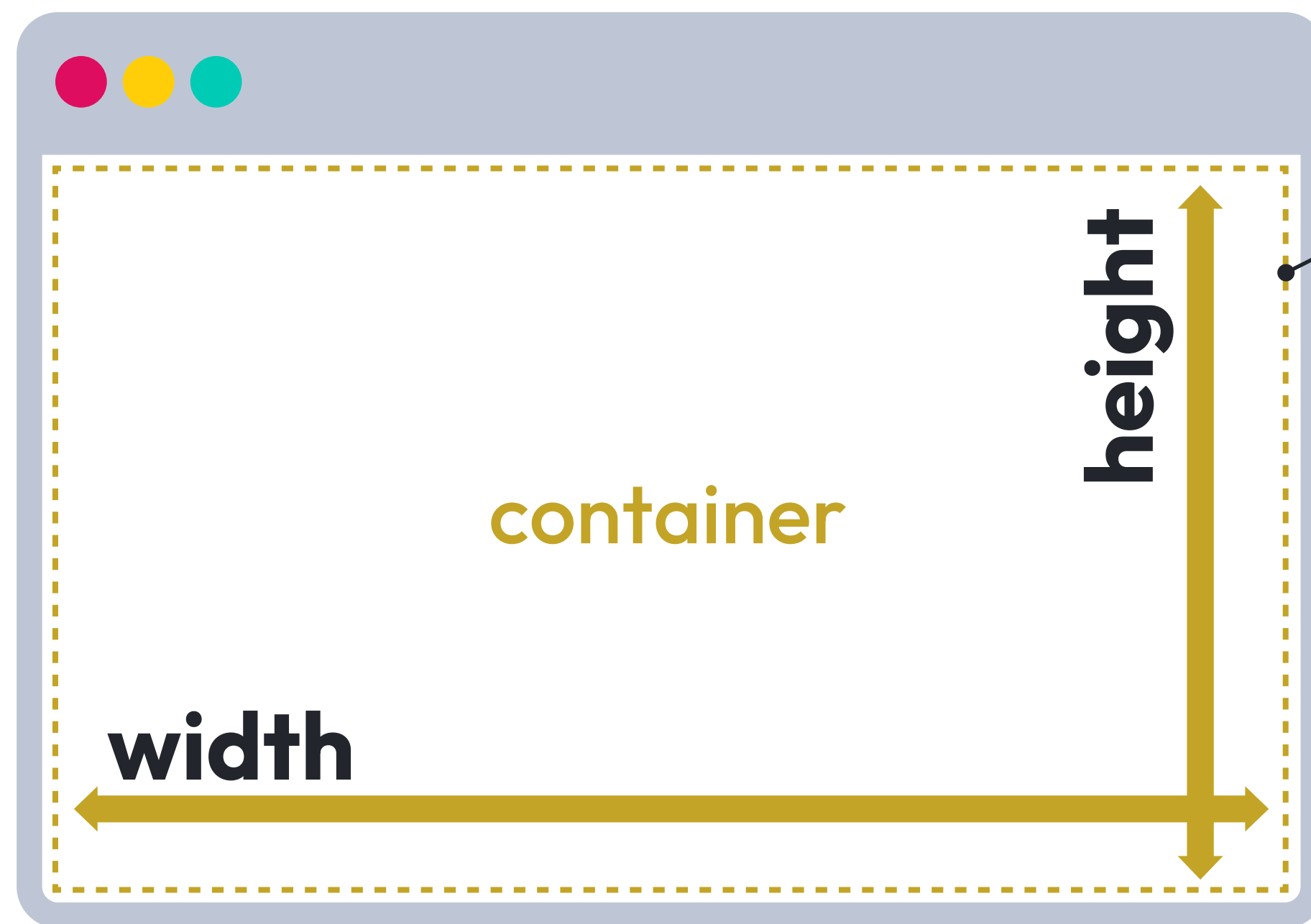


※class 名は container でなくても大丈夫です。

**<body> 直下の container を「<body> と同じ幅・高さ」に設定する。
これが重要です。**



container を <body> の代わりとして機能するようにするため、
「overflow-x : hidden ;」 も併せて指定することが必須です。



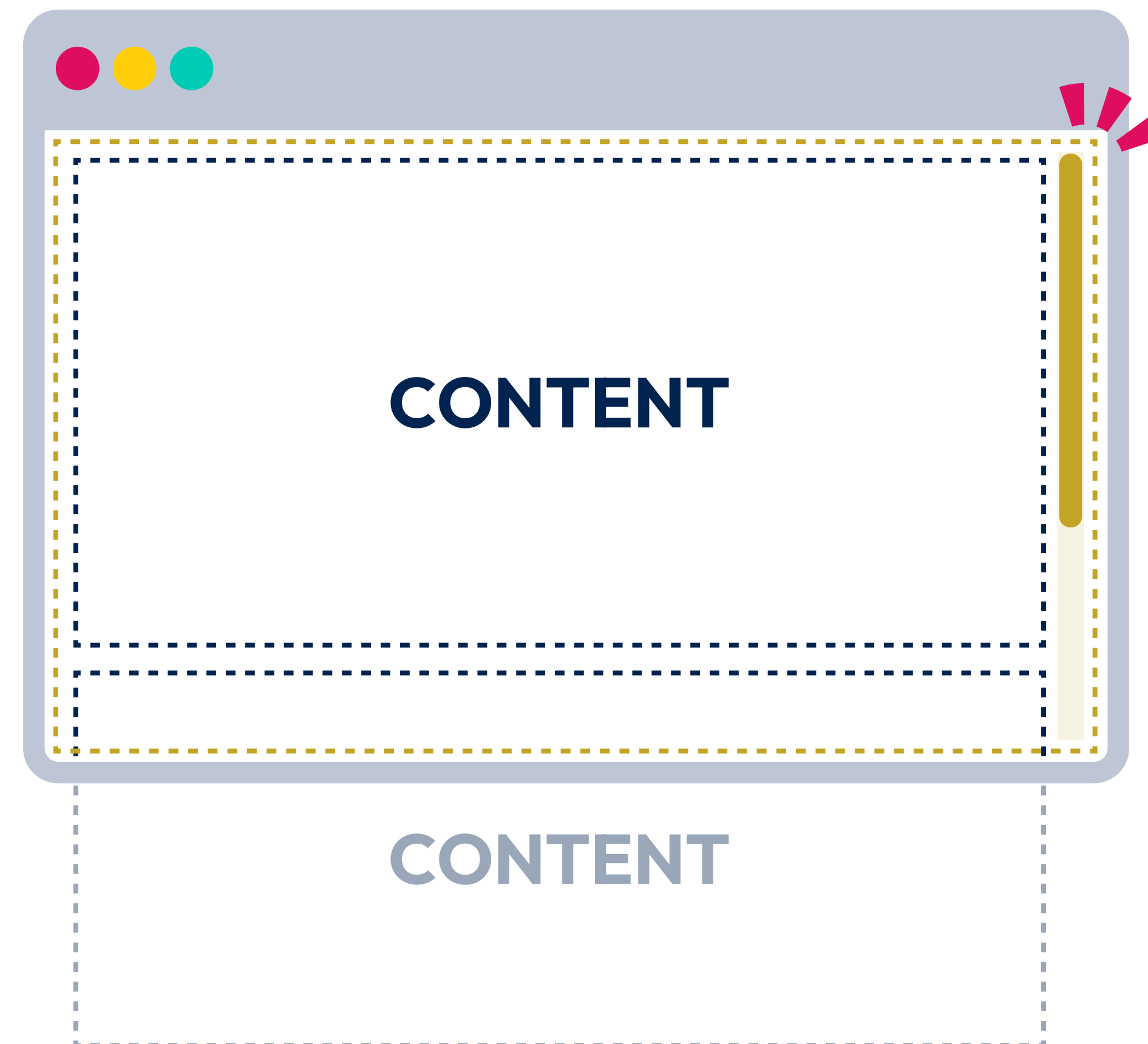
```
<div class="container">
```

```
width : 100% ;
```

```
height : 100svh ;
```

```
overflow-x : hidden ;
```

この状態で container に内包されるコンテンツの高さがブラウザの高さを超えると、
「container のスクロールバー」が出現します。



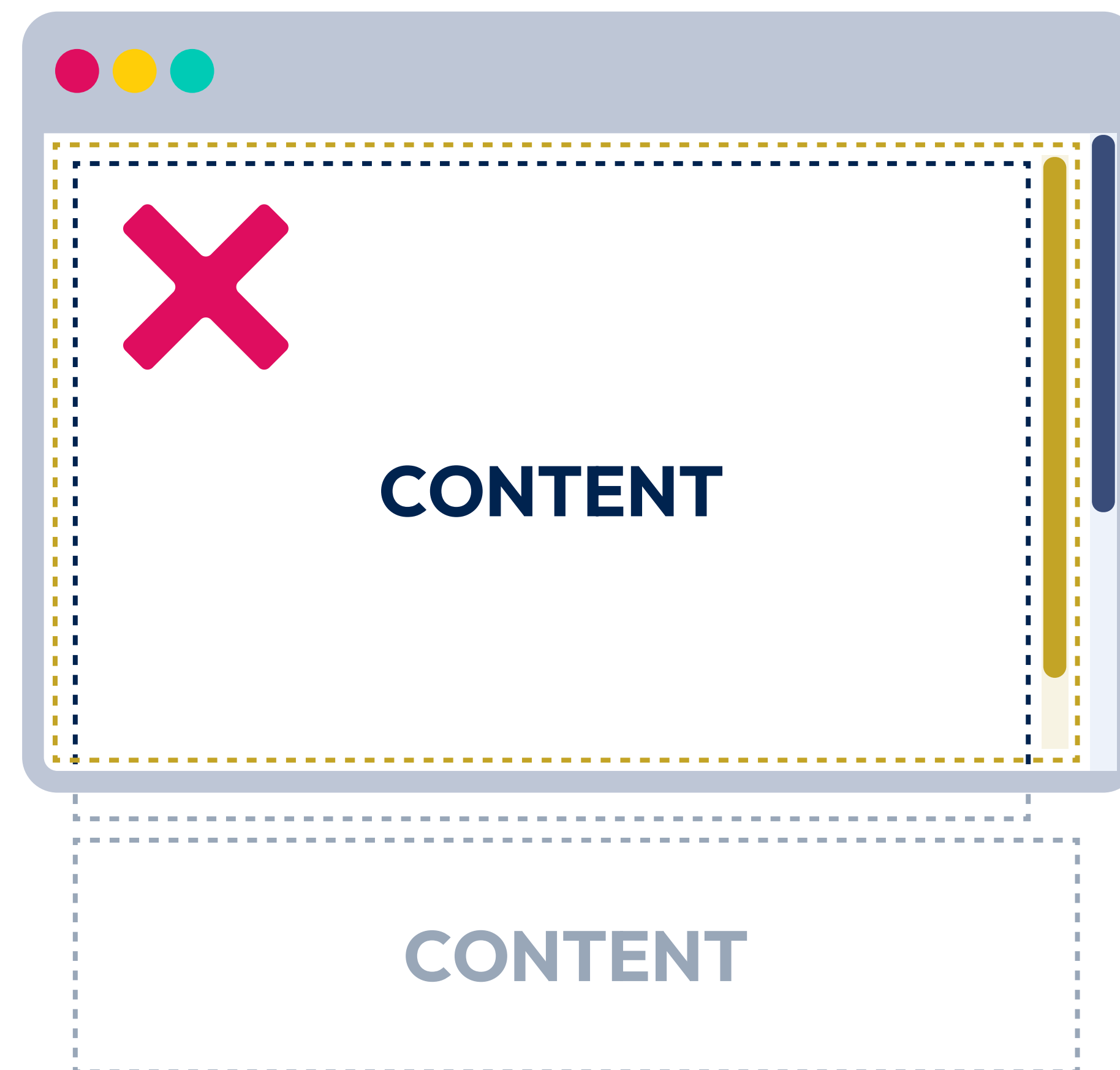
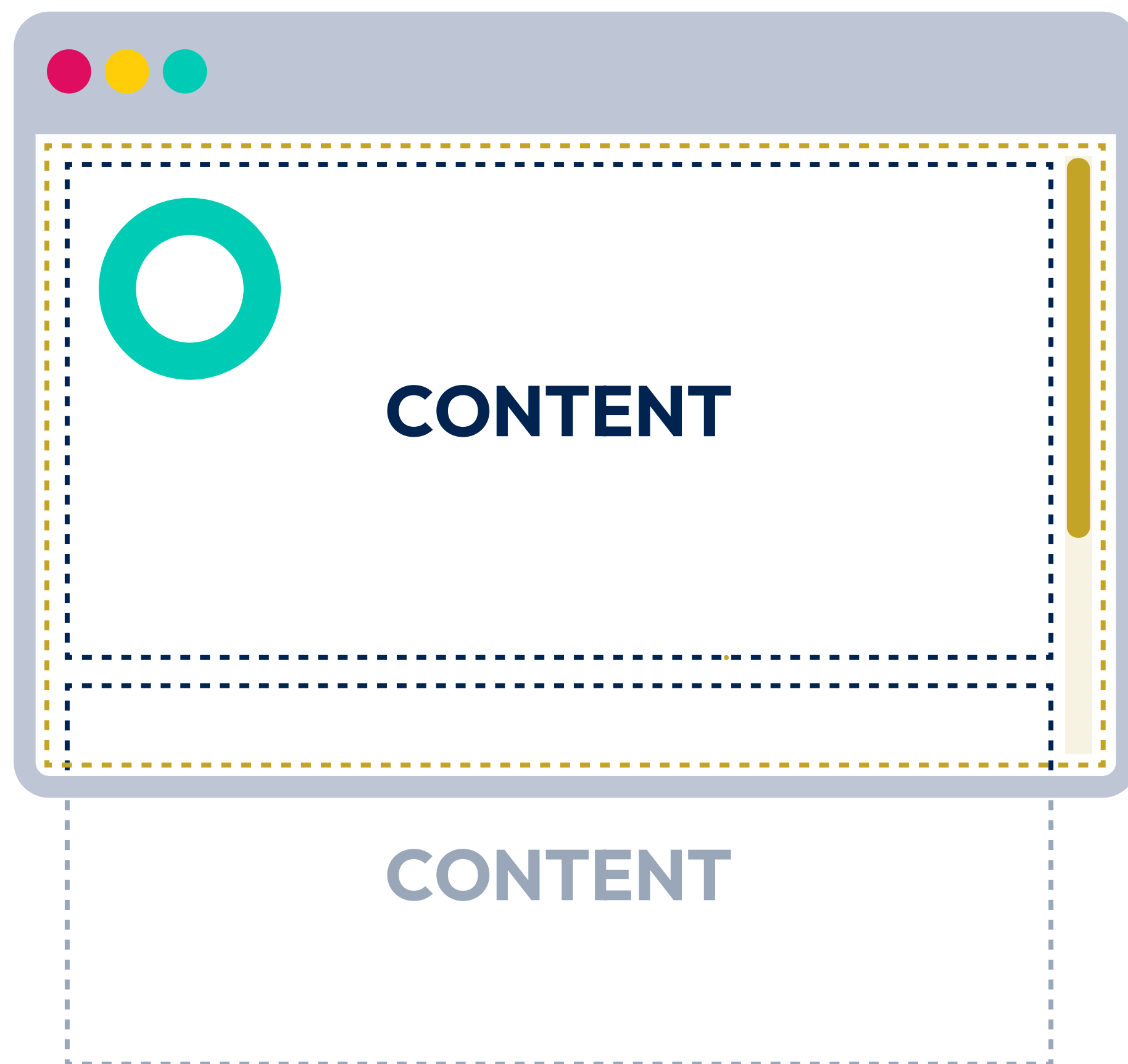
パララックスを実装するページのコンテンツは、
すべて container の中にマークアップします。

下図のように container 外にコンテンツを
マークアップすると…

```
<body>
  <div class="container">
    <div class="parallax">
      <div class="element_A">A</div>
      <div class="element_B">B</div>
      <div class="element_C">C</div>
    </div>
    <!-- ↓これは必須じゃない（下に続くセクション） -->
    <div class="otherSection"></div>
  </div>
</body>
```

```
<body>
  <div class="container">
    <div class="parallax">
      <div class="element_A">A</div>
      <div class="element_B">B</div>
      <div class="element_C">C</div>
    </div>
  </div>
  <!-- ↓これは必須じゃない（下に続くセクション） -->
  <div class="otherSection"></div>
</body>
```


右の図のように、`<body>` と `container` の両方にスクロールバーが出現する状態になってしまい、見た目の美しさと操作性が著しく低下します。



container に「perspective」を指定して、透視投影状態にします。

※perspective の値は任意です。



```
<div class="container">
```

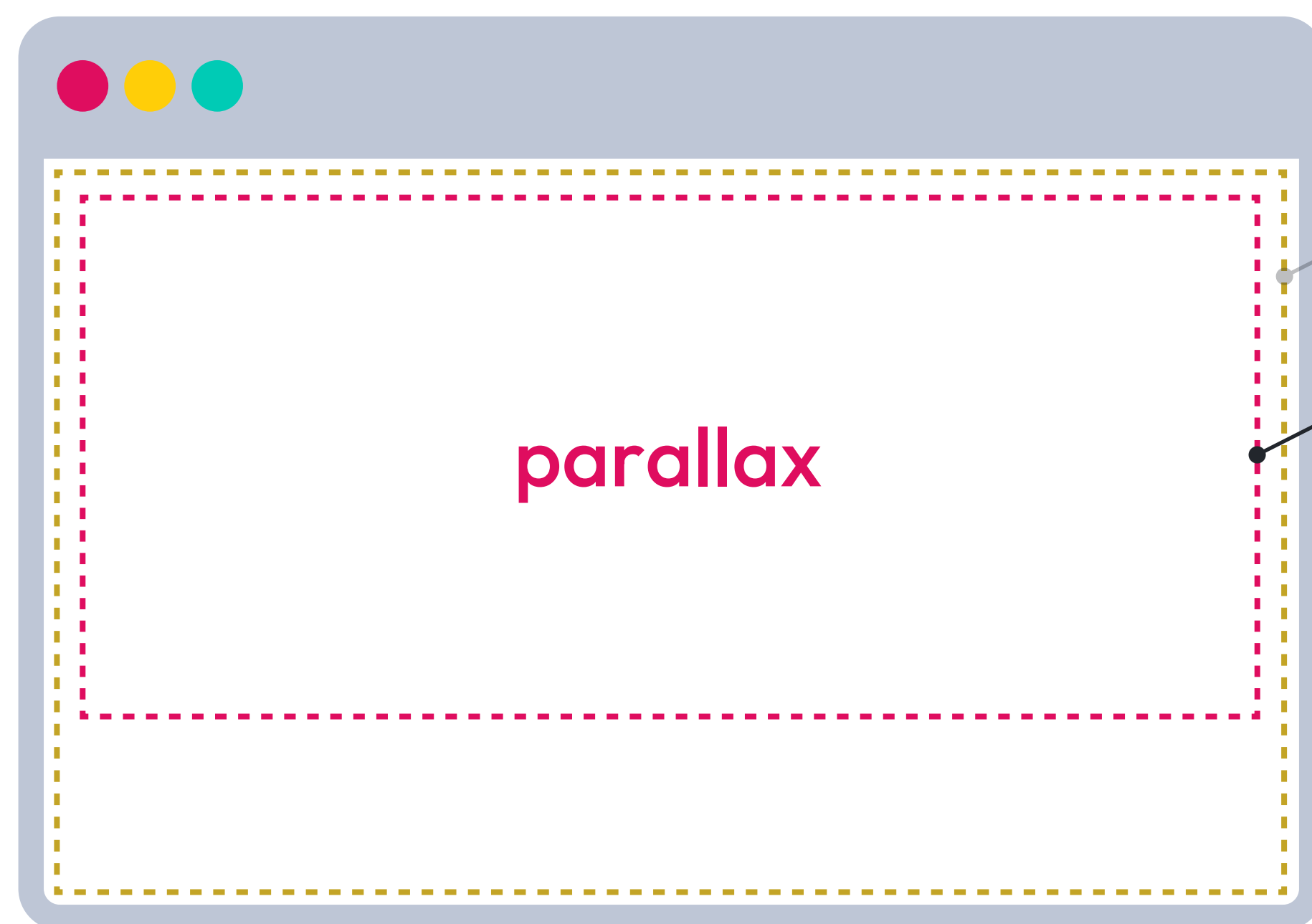
```
width : 100% ;
```

```
height : 100svh ;
```

```
overflow-x : hidden ;
```

```
perspective : 50px ;
```

container 直下の parallax に「transform-style:preserve-3d」を指定します。



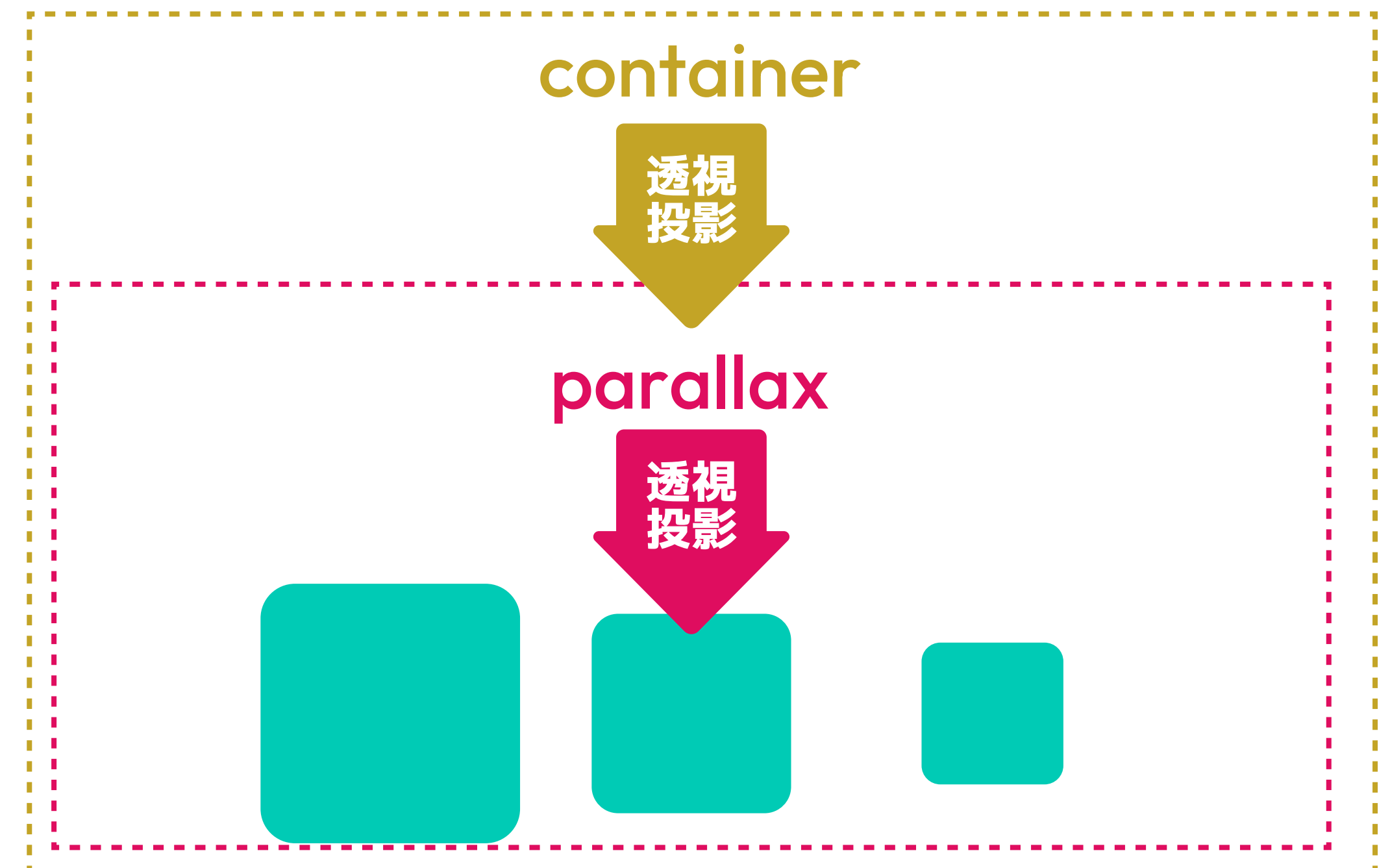
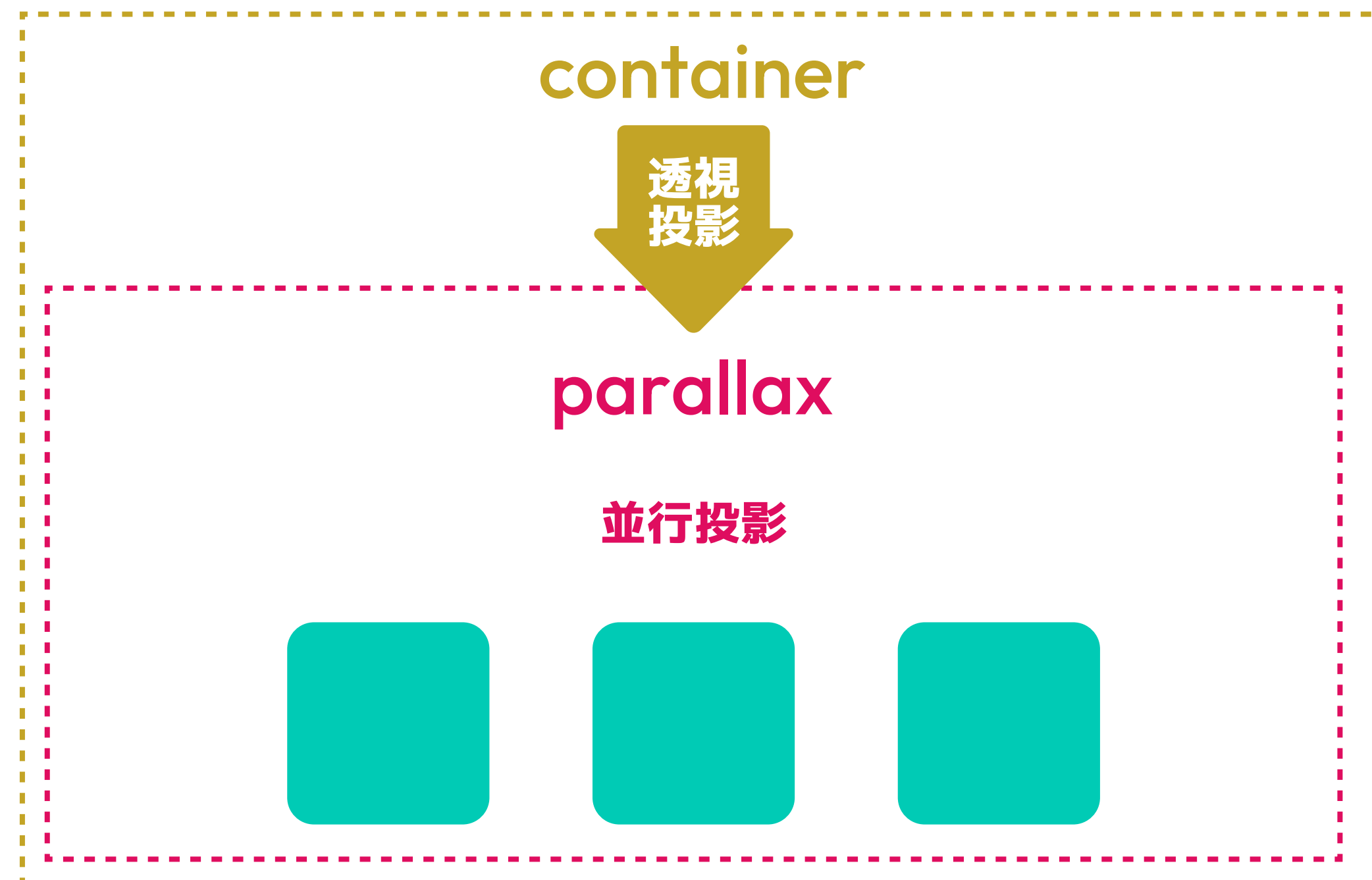
```
<div class="container">
```

```
<div class="parallax">
```

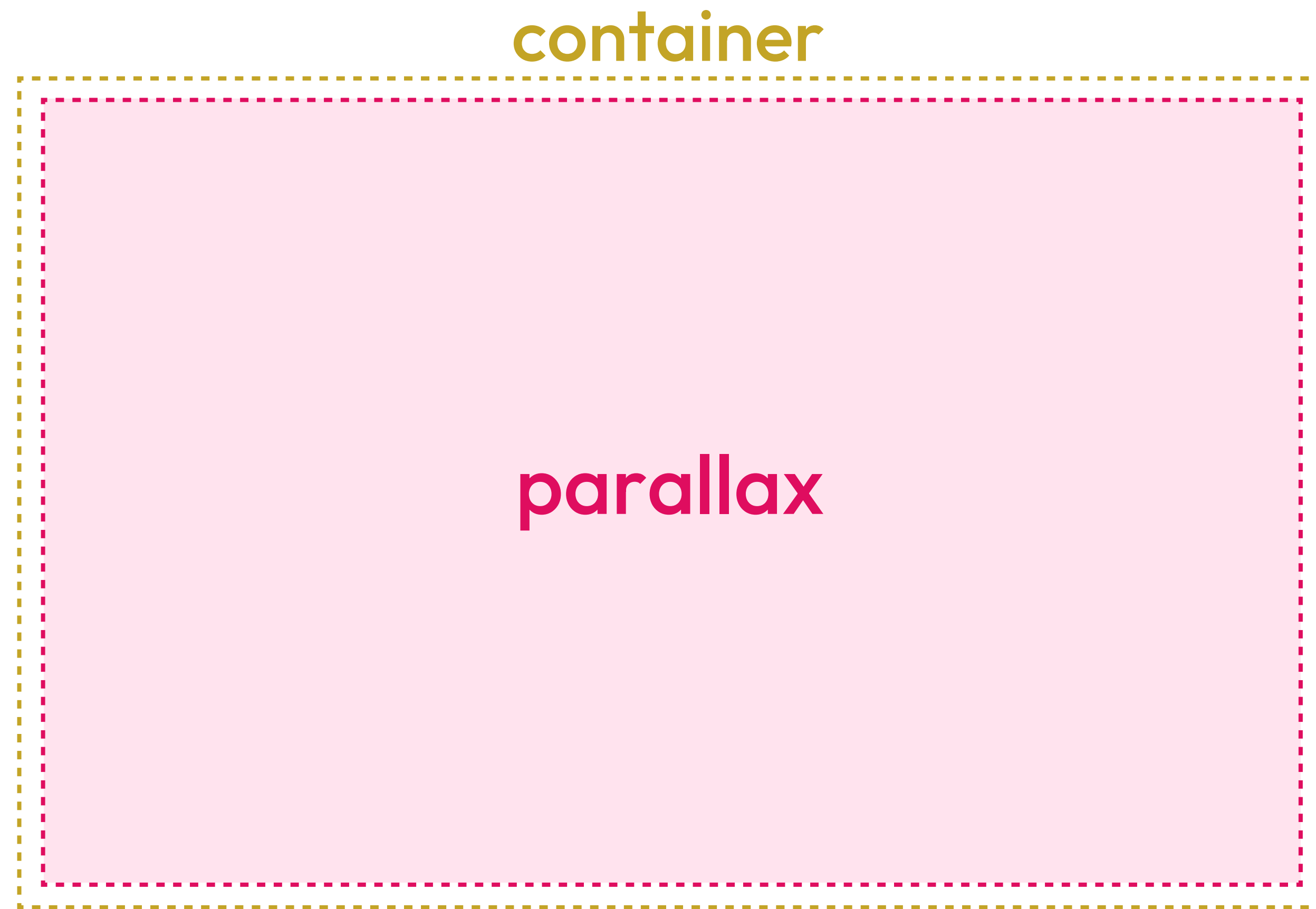
transform-style: preserve-3d;

perspective は通常直下の子要素にしかその影響を与えないので、
そのままだと parallax に内包する要素が平行投影（2D 的な見え方）のままです。

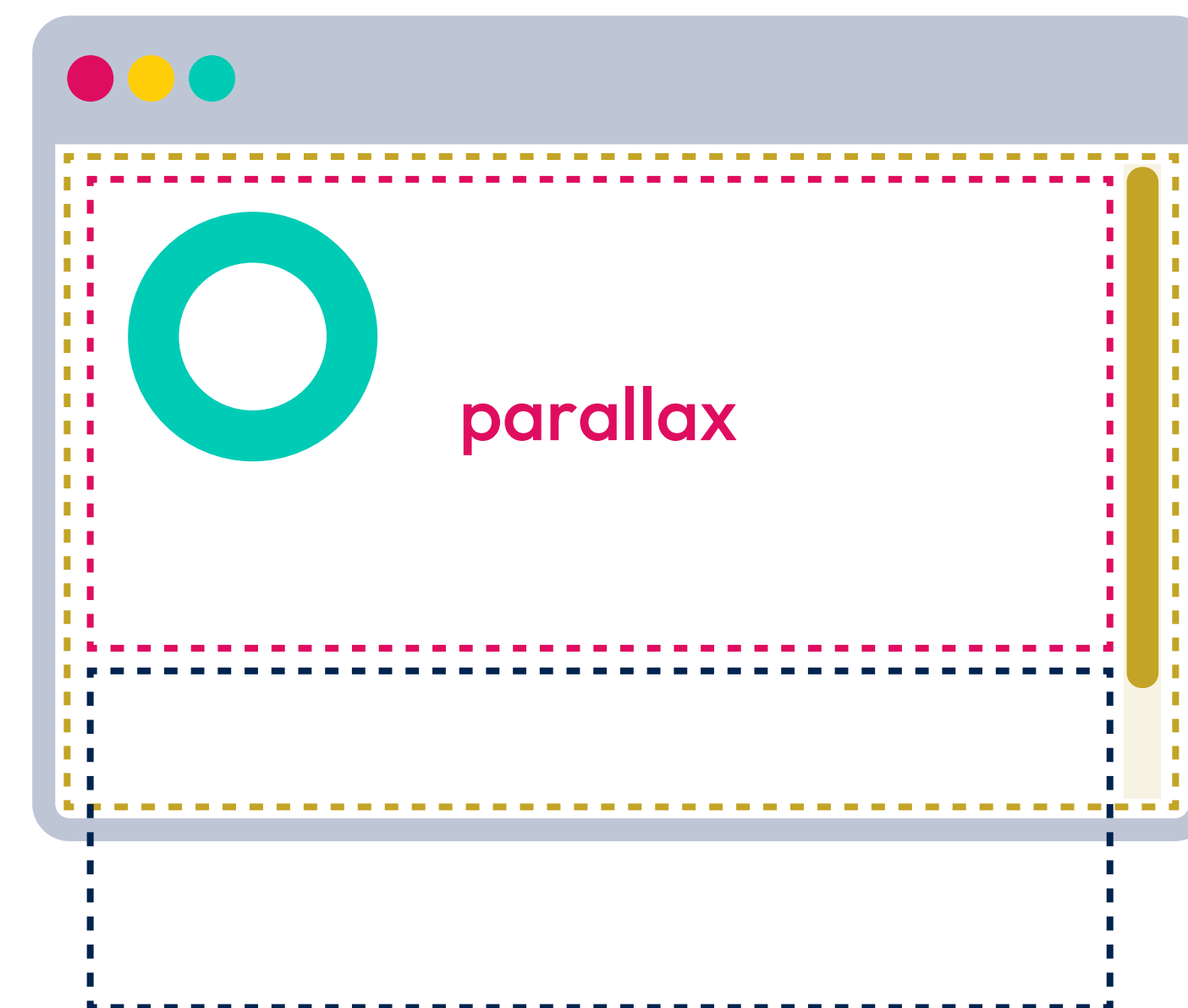
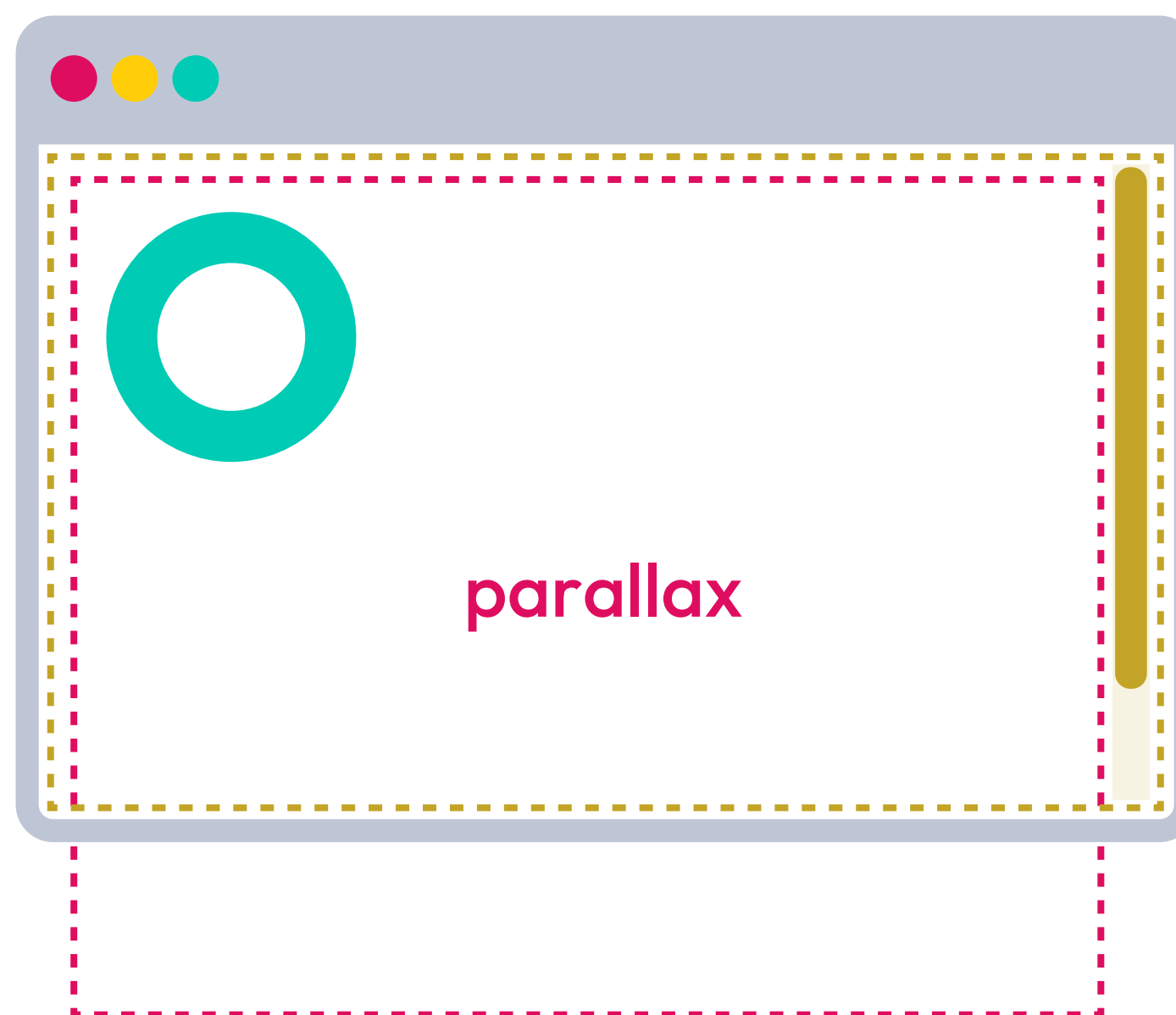
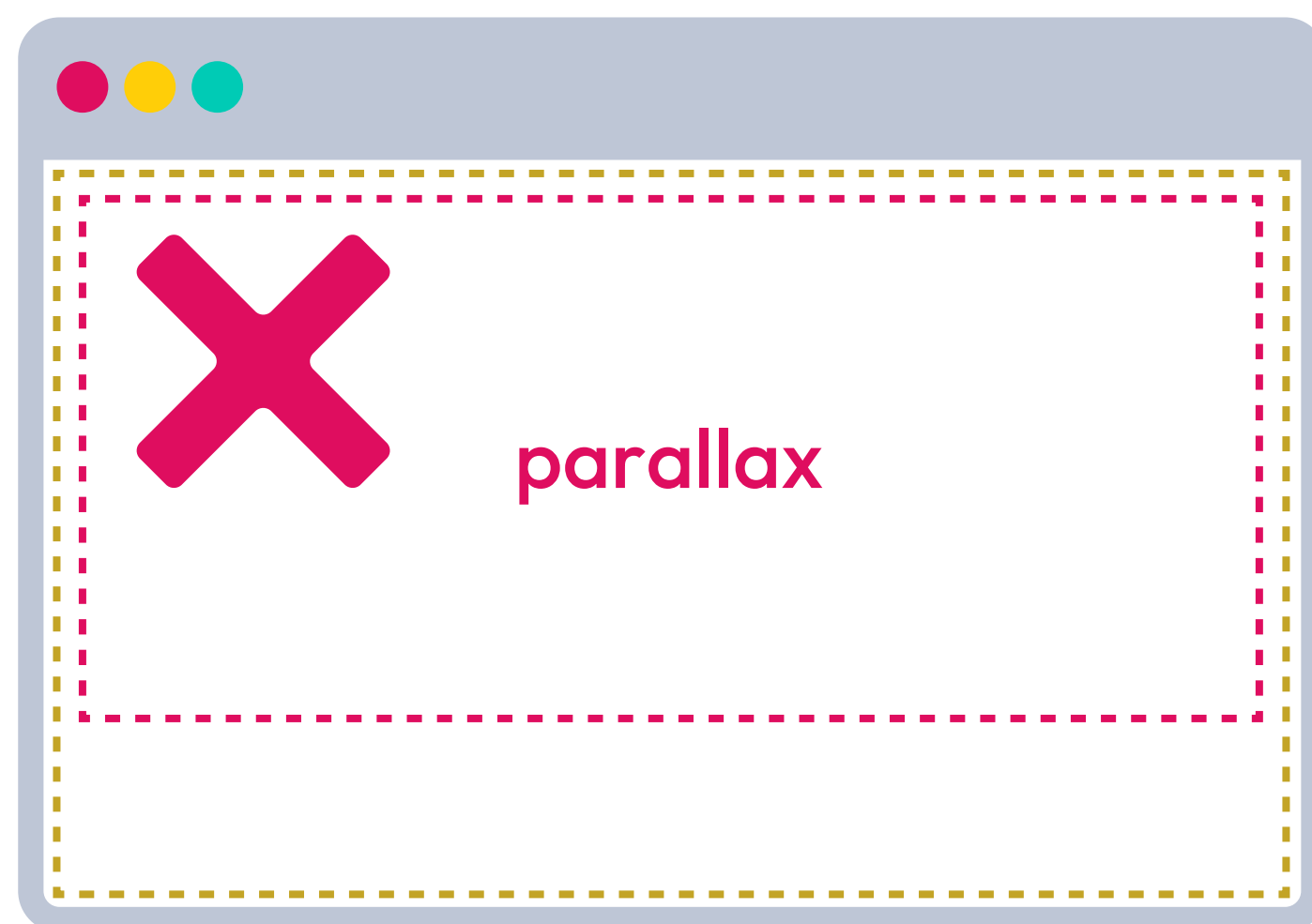
`transform-style: preserve-3d;`



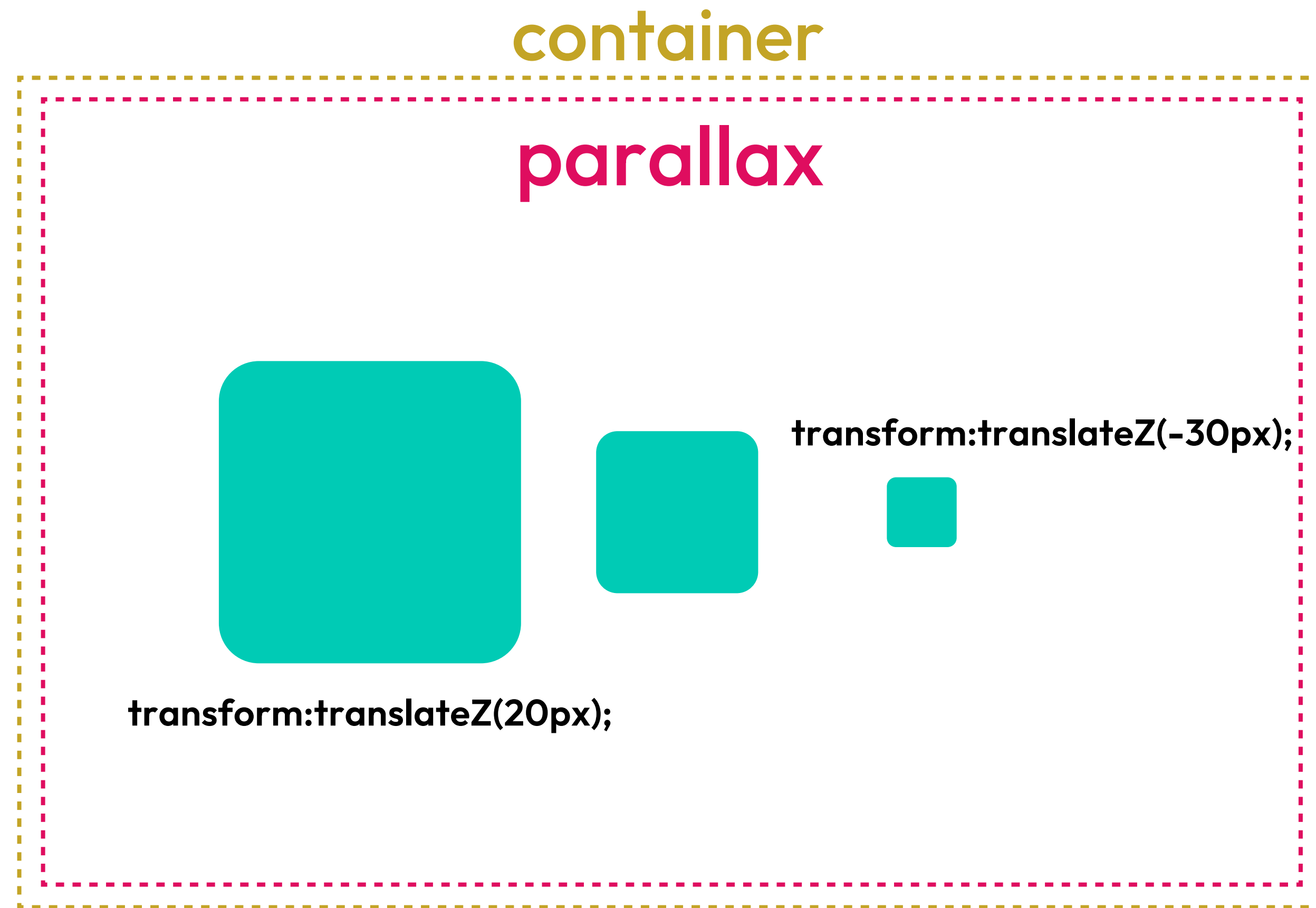
これで parallax 内の空間が透視投影状態（パララックス可能）になります。



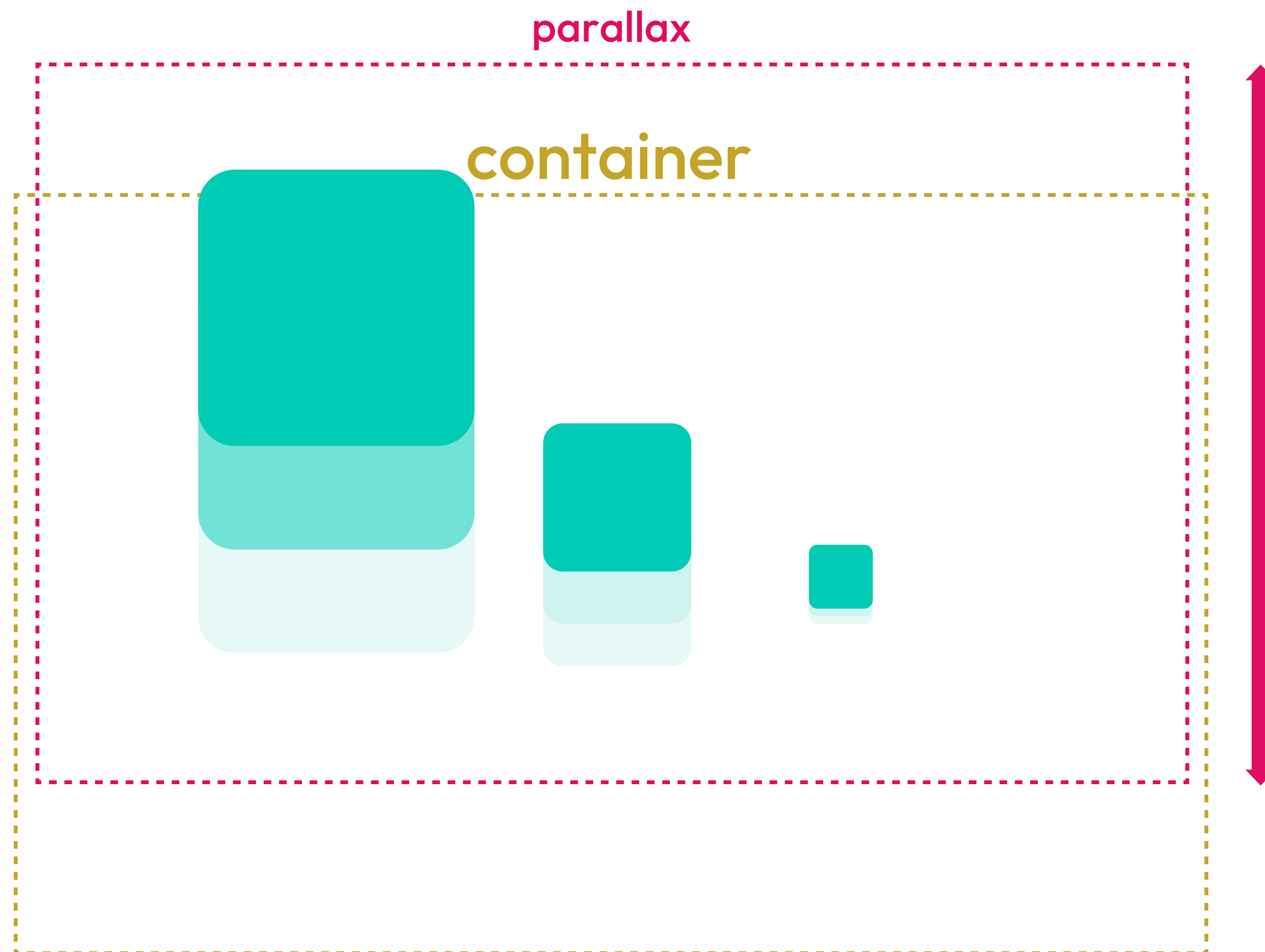
parallax が透視投影状態になっても、スクロールが発生しなければパララックスにはならないので、parallax 本体の高さを container 以上にするか、別のコンテンツを増やしてスクロールを発生させます。



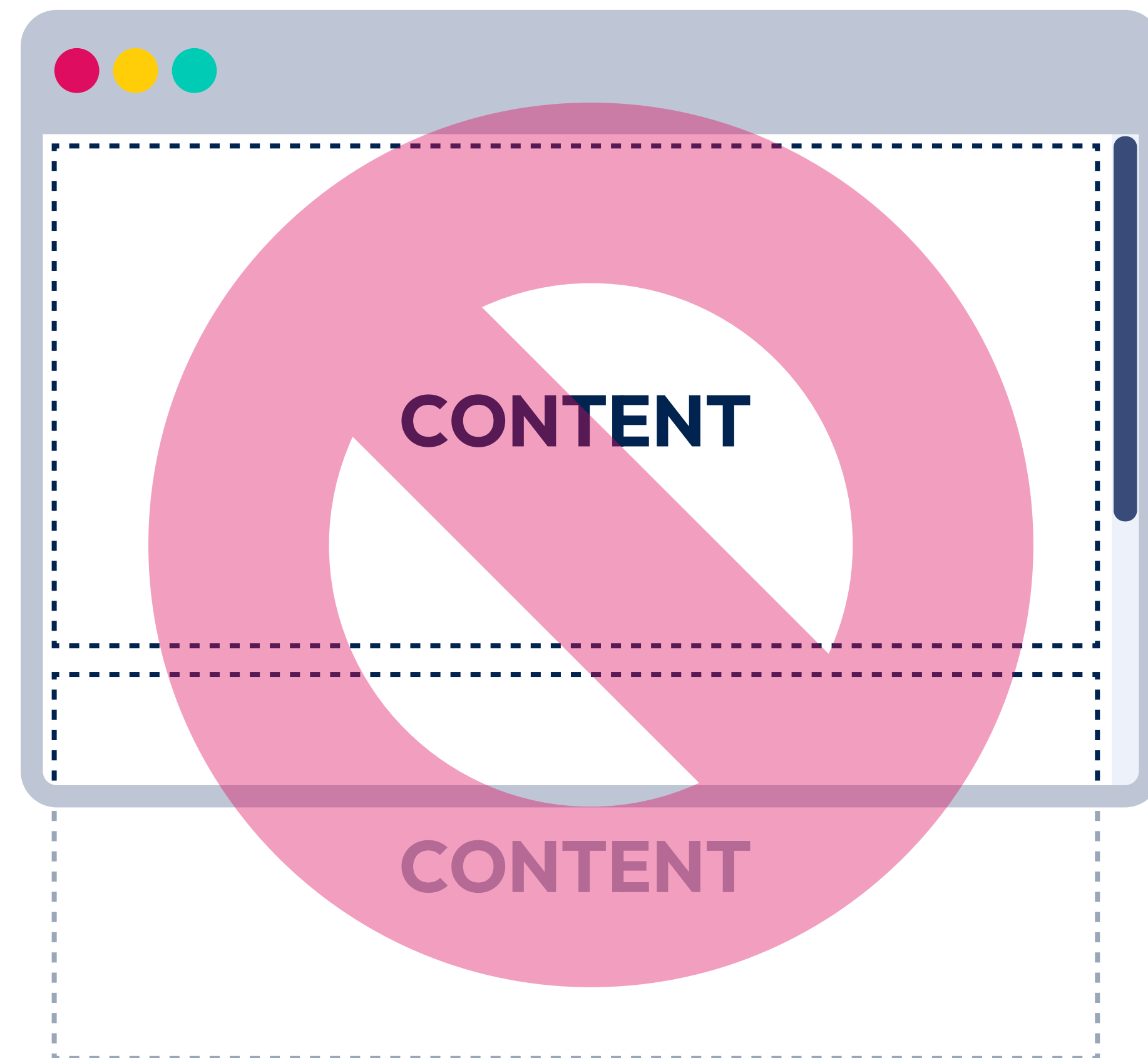
あとは、parallax の中に任意の要素をマークアップし、
それぞれに「transform:translateZ()」を指定して前後に移動させます。



これで、スクロールすると要素ごとに異なるスピードで移動する
「運動視差」を使ったパララックススクロールが実現できます。

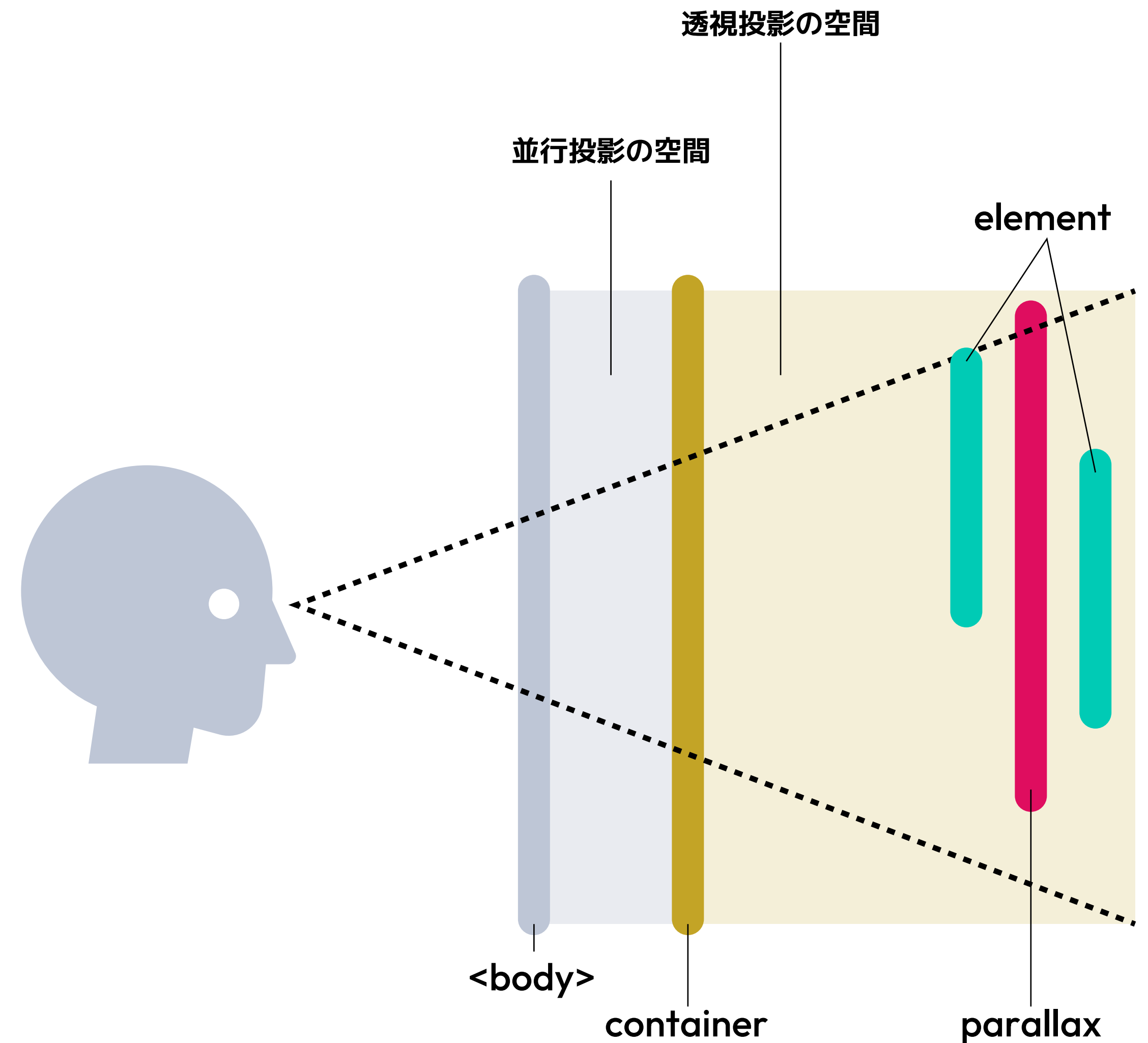


なんで `<body>` のスクロールバーだと
だめなの？

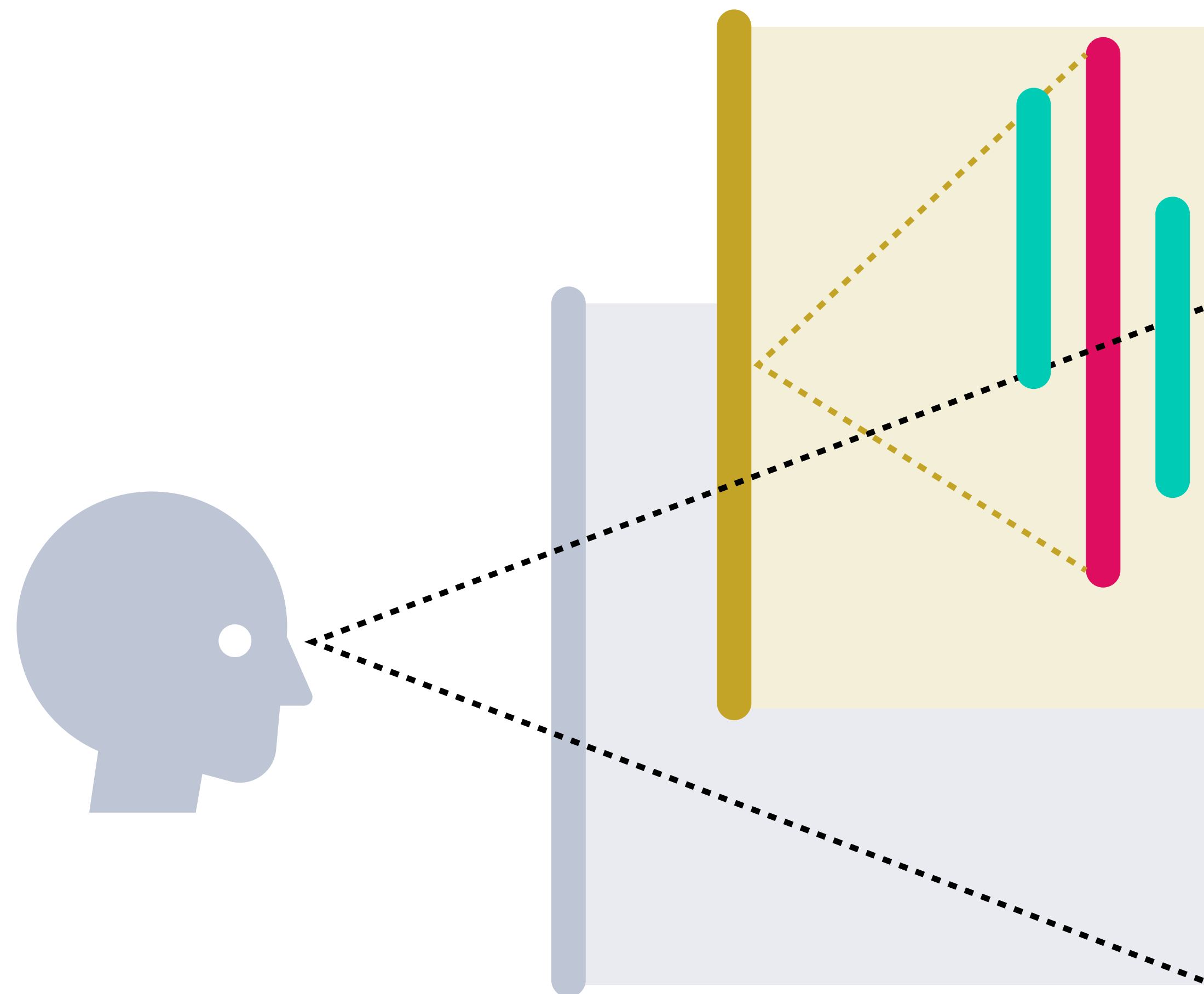


この仕様や実態を論理的にわかりやすく説明することが難しいので、あくまで「イメージ」の話にはなりますが、`<body>` と container では今回のようなケースで「スクロール時の視点の基準」が異なるようです。

左図のように、今回の状態は、`<body>` は平行投影で、container は透視投影という状態です。



<body> のスクロールの場合、透視投影の効果をもつ container がそれぞれ移動してしまうので、透視投影を使った視点移動が適切に働かないようです。



container のスクロールの場合、透視投影の効果をもつ container の中身が移動するので、透視投影を使った視点移動が適切に働くようです。

なんか、上手に説明できなくてごめんなさい・・・。

