

# STACKING CONTEXTS

# STACKING CONTEXTS

While the basics of **z-index** tend to be relatively simple, they open up a can of worms when it comes to stacking contexts.

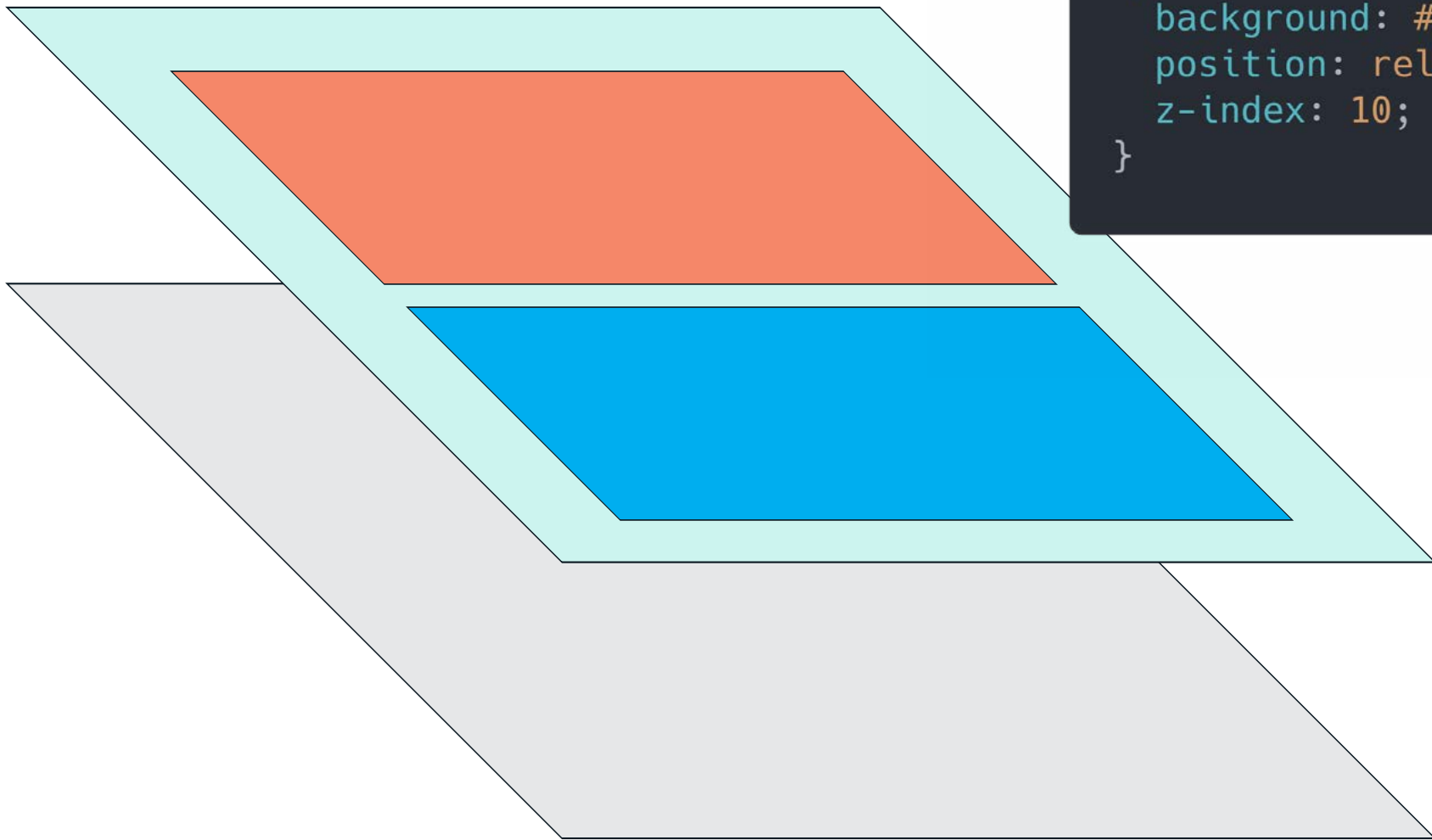
# STACKING CONTEXTS

When we create a new stacking context, it is a little bit like when we create a new block formatting context, in that we are **isolating that element from the others around it**.

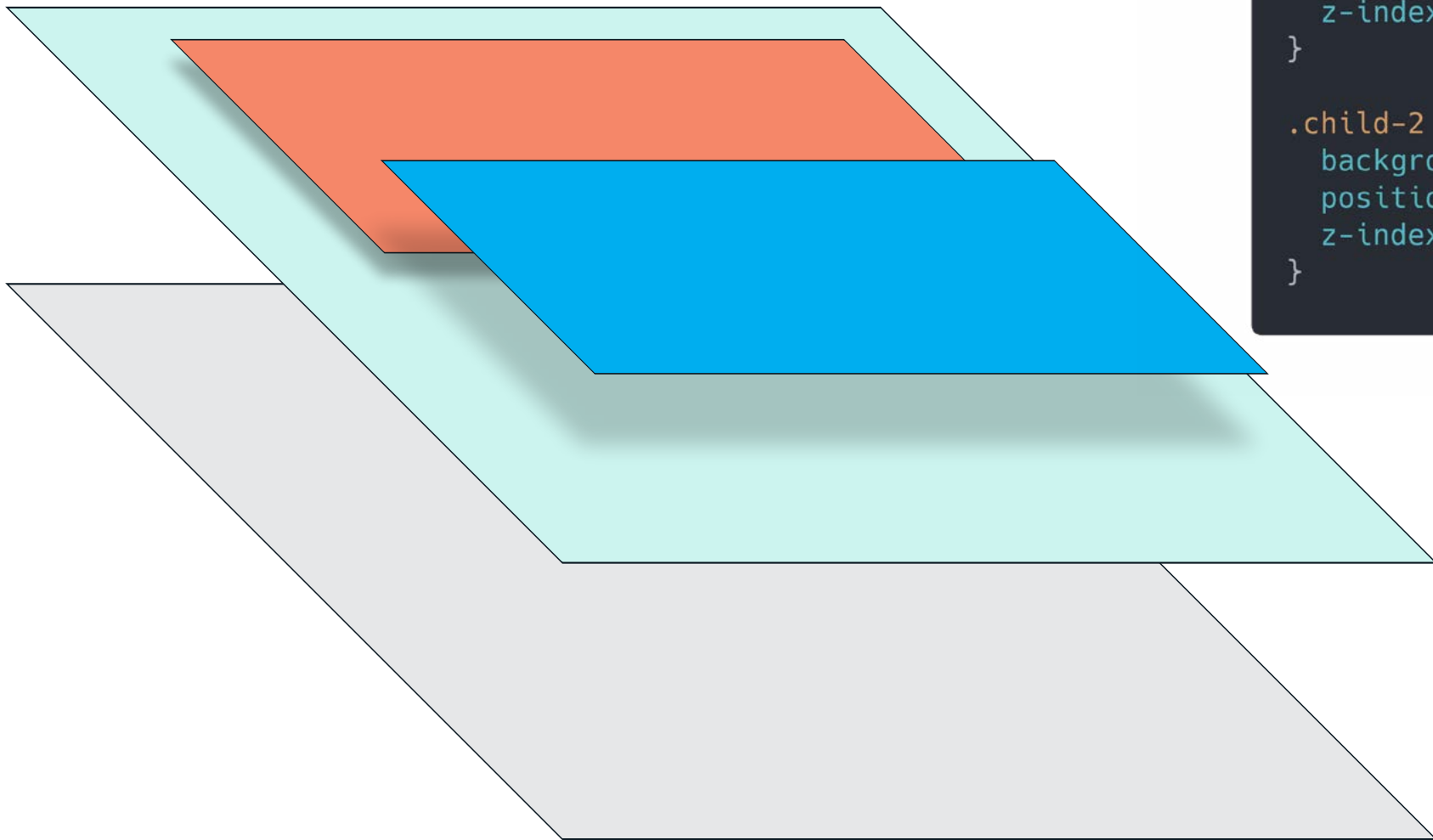
Rather than dealing with things like collapsing margins though, it is all about how z-index works.

# STACKING CONTEXTS

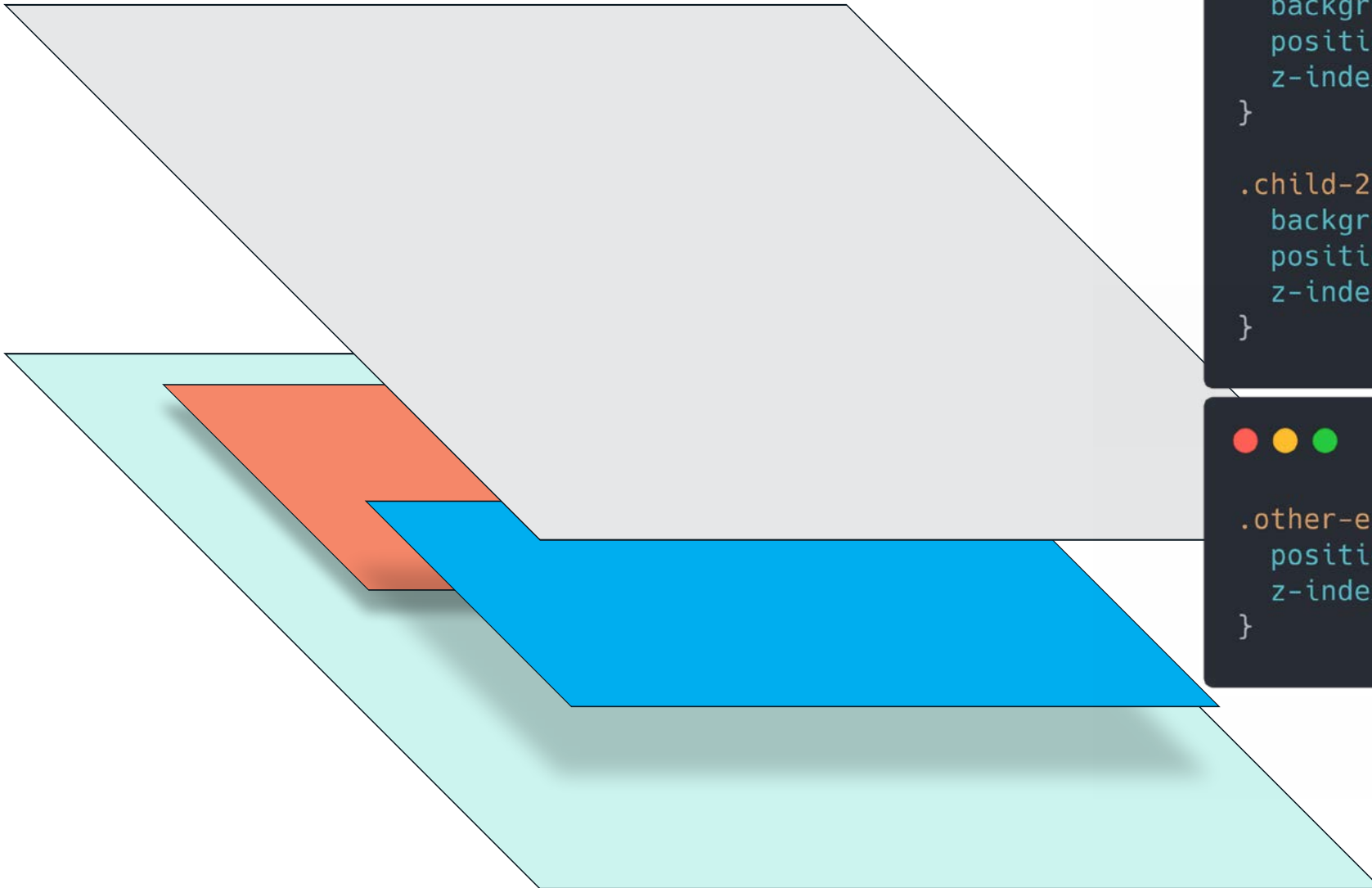
When we create a new stacking context, **z-index** is no longer relative to the root, but to that individual context.



```
.parent {  
  background: #00c7b0;  
  position: relative;  
  z-index: 10;  
}
```



```
.child-1 {  
  background: orange;  
  position: relative;  
  z-index: 10;  
}  
  
.child-2 {  
  background: blue;  
  position: relative;  
  z-index: 25;  
}
```



```
.child-1 {  
  background: orange;  
  position: relative;  
  z-index: 10;  
}
```

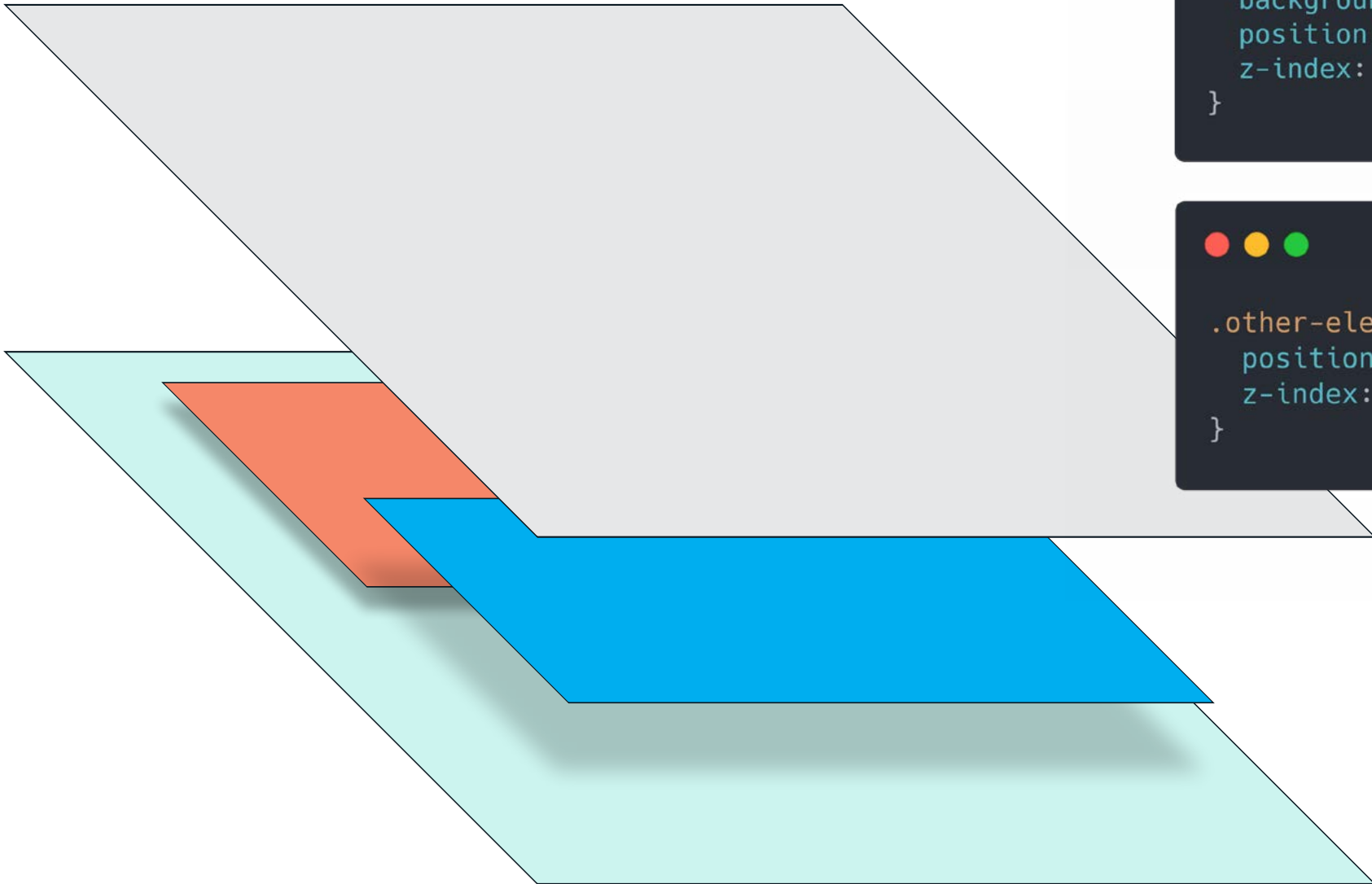
```
.child-2 {  
  background: blue;  
  position: relative;  
  z-index: 25;  
}
```



```
.other-element {  
  position: relative;  
  z-index: 11;  
}
```

???





```
.parent {  
  background: #00c7b0;  
  position: relative;  
  z-index: 10;  
}
```



```
.other-element {  
  position: relative;  
  z-index: 11;  
}
```

**THE PARENT HAS CREATED A NEW  
STACKING CONTEXT**

**IT'S CHILDREN ARE STACKED **WITHIN IT**  
AND CANNOT ESCAPE**

**THIS WOULD BE FINE**  
**EXCEPT FOR ONE THING**

# STACKING CONTEXTS

Similarly to formatting contexts, one of the frustrating things about stacking context is that there are **so many things which create a new stacking context!**

# STACKING CONTEXTS

- The root element of our page
- Both flex and grid items which have a **z-index** other than **auto**
- Elements with an **opacity** that isn't **1**
- Elements with a **transform**, **filter**, **perspective**, **clip-path**, **mask**, or **mix-blend-mode**

# STACKING CONTEXTS

- Elements with a `position: absolute` or `position: relative` which also have a `z-index`
- Elements with a `position: fixed` or a `position: sticky`

**AND THERE IS  
MORE**

**THE FULL LIST:**

[https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Positioning/Understanding\\_z\\_index/The\\_stacking\\_context](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Positioning/Understanding_z_index/The_stacking_context)

**THIS CAN BE INCREDIBLY FRUSTRATING**

**BUT KNOWING WHAT IS CAUSING  
ISSUES WITH STACKING CONTEXT  
MEANS YOU HAVE THE TOOLS AND  
KNOWLEDGE TO FIX PROBLEMS WHEN  
THEY HAPPEN**