

```
const phoneNumber1 = "(222) 555-1234"
const areaCode = /\(222\)/
console.log(phoneNumber1.match(areaCode))
```

```
OUTPUT
[ '(222)', index: 0, input: '(222) 555-1234', groups: undefined ]
```

```
const phoneNumber1 = "(222) 555-1234"
const phoneNumber2 = "(333) 555-1234"
const phoneNumber3 = "(444) 555-1234"
const areaCode = /\((222\))/
console.log(phoneNumber1.match(areaCode))
console.log(phoneNumber2.match(areaCode))
console.log(phoneNumber3.match(areaCode))
```

```
[ '(222)', index: 0, input: '(222) 555-1234', groups: undefined ]
null
null
```

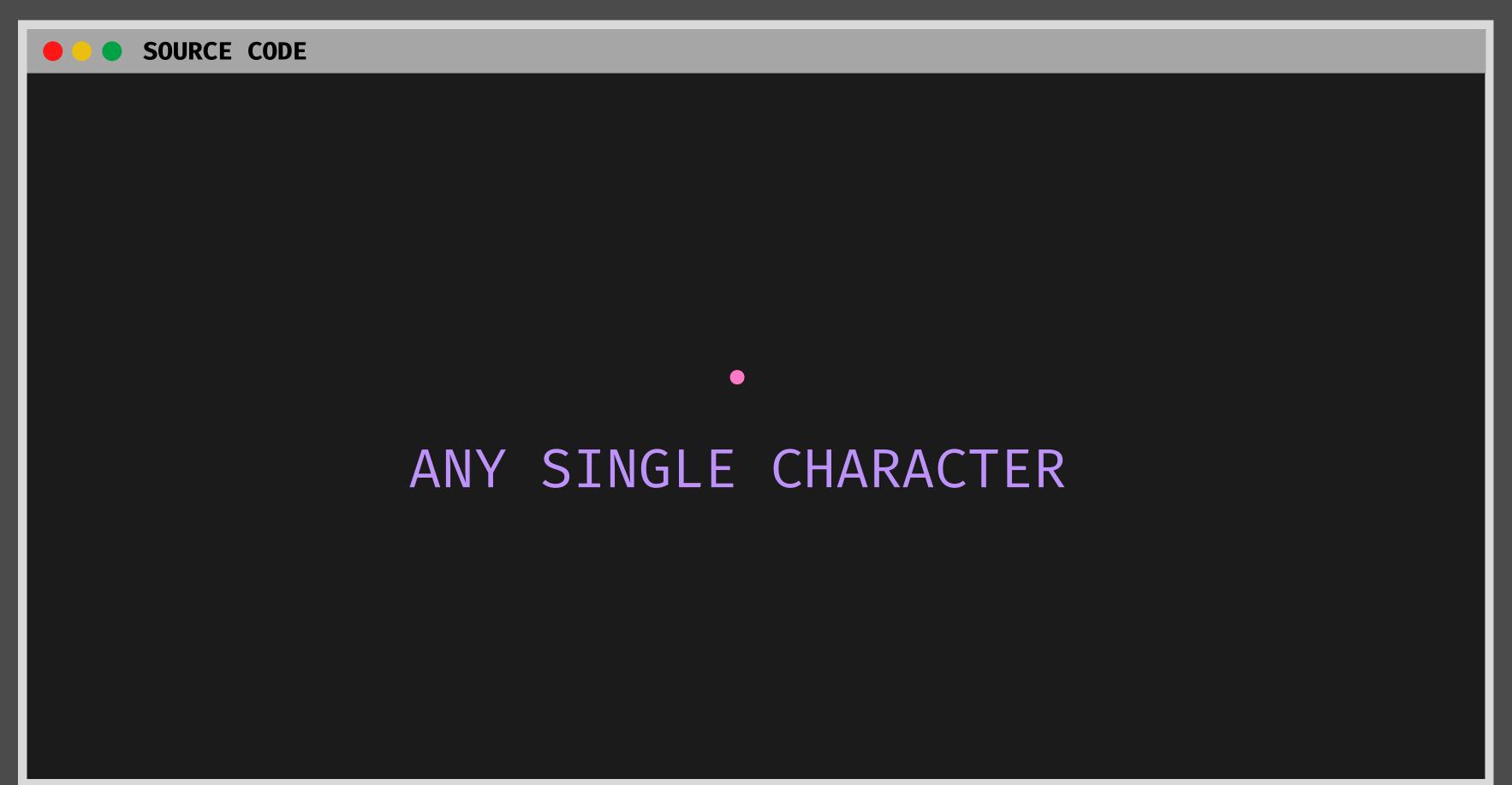
CHALLENGE

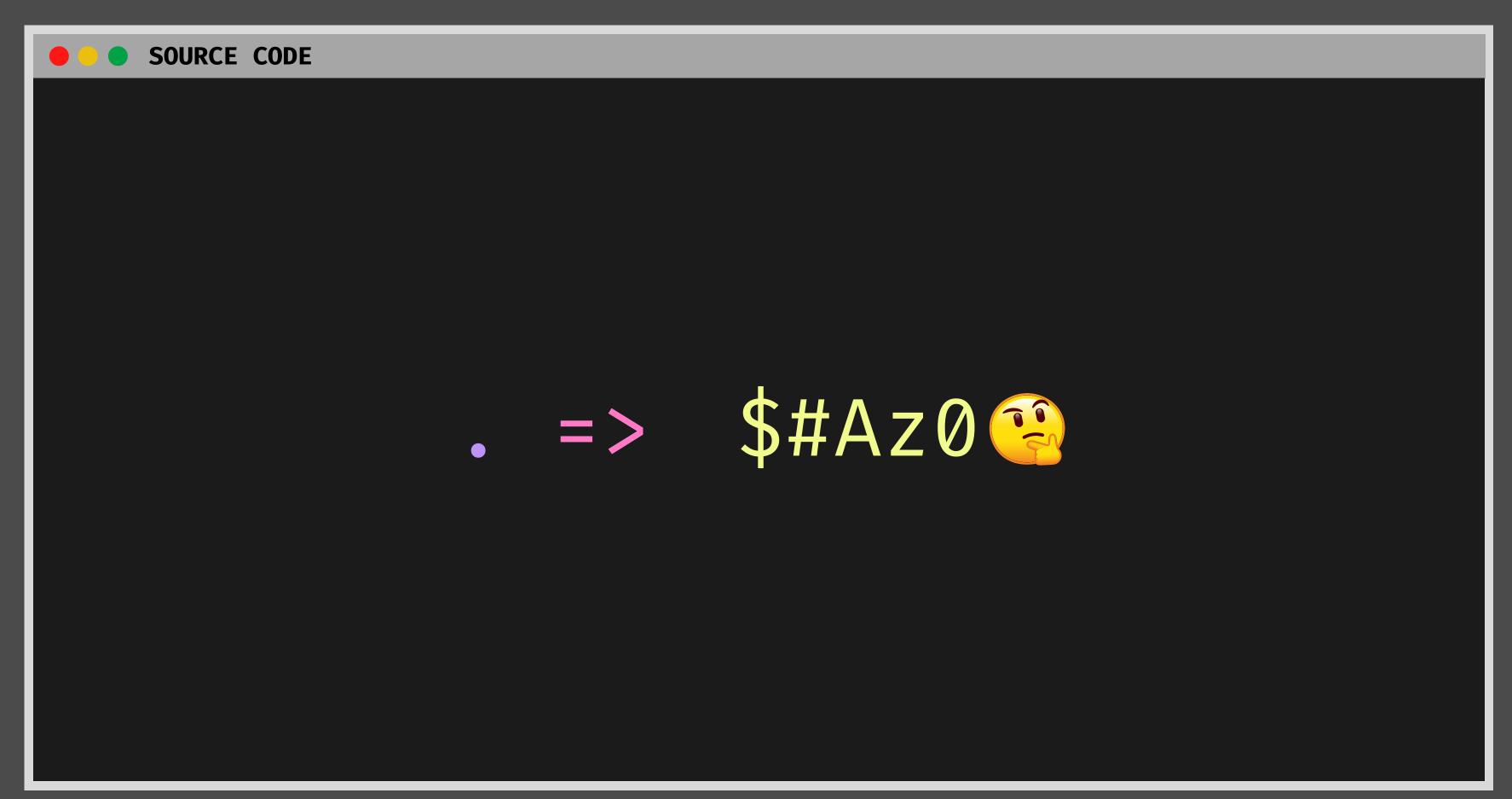
How can we match any single three digits enclosed by parentheses?

```
(222) 555-1234
```

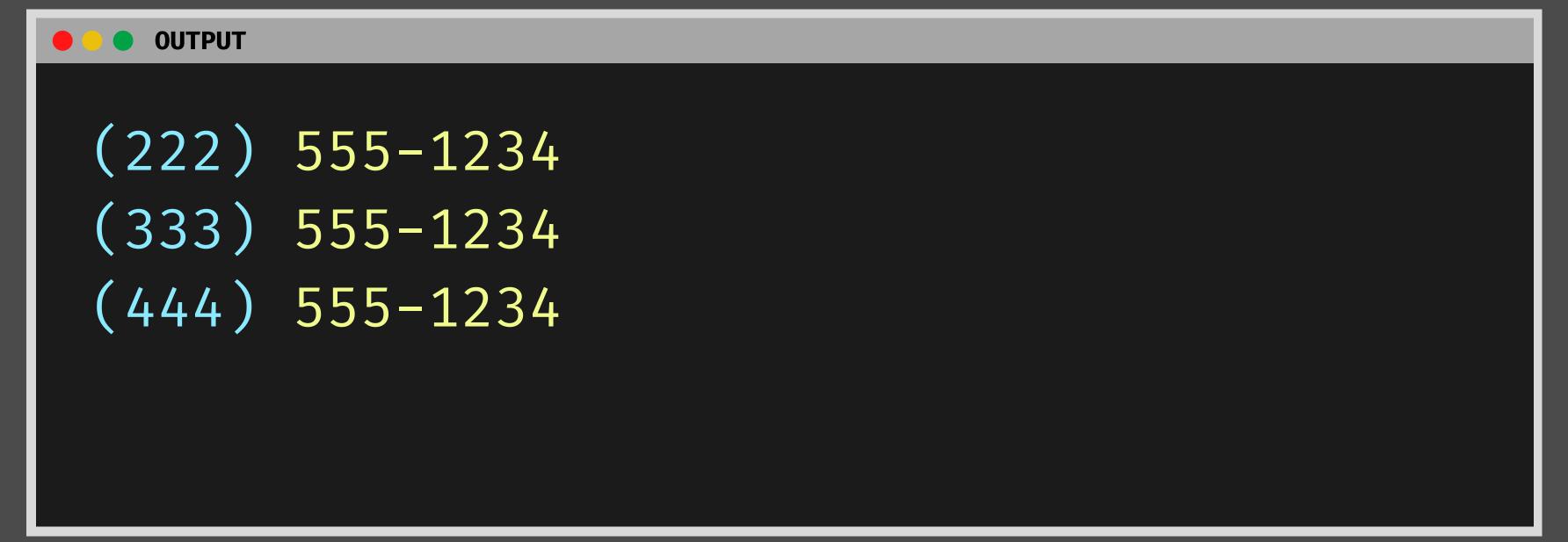
$$(333)$$
 $555-1234$

$$(444)$$
 555-1234



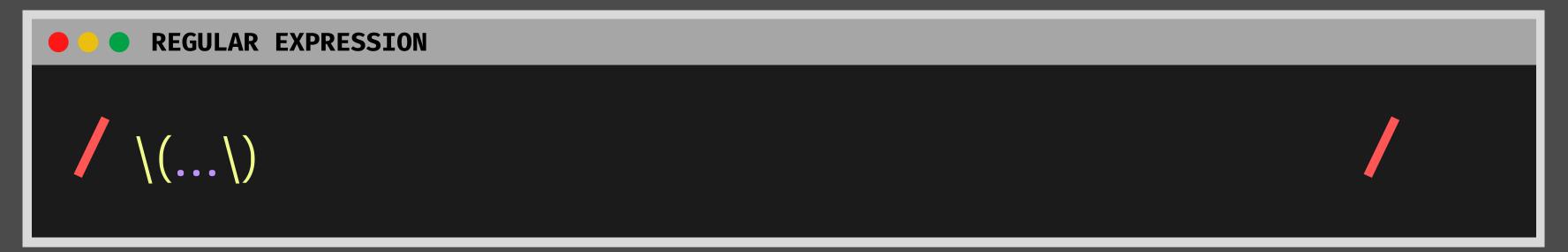






But this can be dangerous, because, for example, there is no such area code. But don't worry, in the future I will show you a metacharacter that matches digits only.

That's it for this lesson and I'll see you in the next one.



```
OUTPUT
(wat) 555-1234
(*-/) 555-1234
(%<sup>6</sup>) 555-1234
```