```
sum [] = 0
sum (x:xs) = x + xs
```
(4)

> The expression `xs` can have two conflicting types

Conflicting types

Possible type 1

```
x :: [a]
```

Inferred from the orange highlights on the left side

Possible type 2

(3)

```
x :: Int
```

Inferred from the blue highlights on the left side

Relevant type information

```
+ :: Int→Int→Int
```

Imported from Prelude