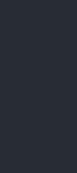


JavaScript

```
import * as R from "ramda";  
import * as Maybe from 'folktale/maybe'  
  
R.map(String)(Maybe.Just(1)); // Just('1')  
R.map(String)(Maybe.Nothing()); // Nothing
```

Haskell

```
data Maybe a = Nothing | Just a  
  
map :: (a -> b) -> Maybe a -> Maybe b  
map _ Nothing = Nothing  
map f Just a = Just f a  
  
add x y = x + y  
add1 = add 1  
map add1 (Just 1) -- Just 2  
map add1 (Nothing) -- Nothing
```



Functor

```
map(String)([1]) // ['1']  
map(String)(Observable.of(1)) // Observable.of('1')  
map(String)(Maybe.Just(1)); // Just('1')
```

