



Structure-direct

```
data S = S { a :: Int  
             , b :: String  
            }
```

```
data T = T { a :: Int  
          , b :: String  
          , c :: Bool  
          }
```

```
> view(super@S)(T 5 "abc" True)
S 5 "abc"
```

> :t (super @s)

LenS TS

```
> set(super@S) (S 6 "Matt" (T 5 "abc" True)
```

```
T 6 "Matt" True
```

# Structure-directed

```
data S = S { a :: Int
             , b :: String
             }
```

```
data T = T { a :: Int
             , b :: String
             , c :: Bool
             }
```

```
> :t (super @S)
```

```
Lens T S
```

```
> view (super @S) (T 5 "abc" True)
```

```
S 5 "abc"
```

```
> set (super @S) (S 6 "Matt") (T 5 "abc" True)
```

```
T 6 "Matt" True
```



# Implementation