

Implementation: `field`

```
class HasField (field :: Symbol) a s | s field → a where  
  field :: Lens s a
```

```
instance  
  (Generic s  
  ,ErrorUnless field s (HasTotalFieldP field (Rep s))  
  ,GHasField field (Rep s) a  
  ) ⇒ HasField field a s where  
  field = ... gfield ...
```

Idea

- Type class instances for each of the generic constructors
- Specify each simple case in turn