FREE eBook

LEARNING shapeless

Free unaffiliated eBook created from **Stack Overflow contributors.**

#shapeless

Table of Contents

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: shapeless

It is an unofficial and free shapeless ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official shapeless.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with shapeless

Remarks

This section provides an overview of what shapeless is, and why a developer might want to use it.

It should also mention any large subjects within shapeless, and link out to the related topics. Since the Documentation for shapeless is new, you may need to create initial versions of those related topics.

Examples

Installation or Setup

Detailed instructions on getting shapeless set up or installed.

Read Getting started with shapeless online: https://riptutorial.com/shapeless/topic/10138/getting-started-with-shapeless

Chapter 2: Typeclass derivation

Examples

ProductTypeClass

```
trait Show[T] {
  def show(t: T): String
}

object Show extends ProductTypeClassCompanion[Show] {

  def apply[T](implicit T: Show[T]) = T

  def from[T](f: T => String): Show[T] = new Show[T] {
    def show(t: T): String = f(t)
  }

  implicit val string = from[String](_.reverse)
  implicit val int = from[Int](Integer.toHexString)

  object typeClass extends ProductTypeClass[Show] {
    override def product[H, T <: HList](ch: Show[H], ct: Show[T]): Show[::[H, T]] = from(ht =>
      ch.show(ht.head) + " " + ct.show(ht.tail)
    )

    override def emptyProduct: Show[HNil] = from(_ => "")

    override def project[F, G](instance: => Show[G], to: (F) => G, from: (G) => F): Show[F] =
      Show.from((f: F) => instance.show(to(f)))
  }
}

case class Example(name: String, value: Int)

Show[Example].show(Example("Me", 12))  // returns "eM c"
```

Read Typeclass derivation online: https://riptutorial.com/shapeless/topic/10175/typeclass-derivation

Credits

S. No	Chapters	Contributors
1	Getting started with shapeless	Community
2	Typeclass derivation	Cyrille Corpet