

Scala の implicit を理解したい

2025-03-02


Name: ADACHI Yuya

E-mail: s2120001@jaist.ac.jp

- Scala 2 の時代に implicit が導入された
- implicit は "暗黙的" という意味
- 学習テキストとして「Scala研修テキスト」がよくまとまっている
- Scala研修テキスト:https://scala-text.github.io/scala_text/implicit.html

```
scala Copy  
  
// 暗黙の値を定義  
implicit val defaultTimeout: Int = 5000  
  
// 暗黙のパラメータを持つ関数  
def executeQuery(query: String)(implicit timeout: Int): String = {  
  // timeoutを使ったクエリ実行ロジック  
  s"Executing query: $query with timeout: $timeout"  
}  
  
// 明示的にパラメータを指定  
executeQuery("SELECT * FROM users")(10000) // "Executing query: SELECT * FROM users with timeout: 10000"  
  
// 暗黙のパラメータを使用（スコープ内のimplicit valが使われる）  
executeQuery("SELECT * FROM users") // "Executing query: SELECT * FROM users with timeout: 5000"
```


scala

 Copy

```
case class Person(name: String) {  
  def greet(): String = s"Hello, I'm $name"  
}  
  
// String から Person への暗黙の変換  
implicit def stringToPerson(name: String): Person = Person(name)  
  
// Stringに対してPersonのメソッドを呼び出せるようになる  
val greeting = "John".greet() // "Hello, I'm John"
```

implicit クラス (拡張メソッド)

scala

 Copy

```
// StringクラスにisBlankメソッドを追加
implicit class StringOps(val s: String) extends AnyVal {
  def isBlank: Boolean = s == null || s.trim.isEmpty
}

// 元々Stringクラスにはないメソッドが使える
"".isBlank      // true
" ".isBlank     // true
"hello".isBlank // false
```

- 確かに implicit は便利な機能
- implicit という用語で複数の機能に対応している
- ユーザーフレンドリーではないという批判があった
- そこで、Scala 3 では implicit を以下の 6 に整理されている
 - Given インスタンス (given) : 従来の implicit val/def に相当
 - Using パラメータ (using) : 従来の implicit parameter に相当
 - Extension メソッド (extension) : 従来の implicit class に相当
 - Type クラス導出 : implicit macro を改良したもの
 - Import Givens : import 構文の拡張 (given キーワードによる明示的な import)
 - summon 関数 : 従来の implicitly 関数に相当する新しい標準関数

scala

 Copy

```
// 型クラスの定義
trait Show[T]:
  def show(value: T): String

// givenインスタンスの定義
given Show[Int] with
  def show(value: Int): String = s"Int: $value"

given Show[String] with
  def show(value: String): String = s"String: $value"

// givenインスタンスの使用
def displayValue[T](value: T)(using shower: Show[T]): Unit =
  println(shower.show(value))

// 使用例
displayValue(42)          // "Int: 42"を出力
displayValue("hello")    // "String: hello"を出力
```

scala

 Copy

```
object StringExtensions {  
  extension (s: String) {  
    def exclaim: String = s"$s!"  
    def isPalindrome: Boolean = {  
      val normalized = s.toLowerCase.replaceAll("[^a-z0-9]", "")  
      normalized == normalized.reverse  
    }  
  }  
}  
  
object Main {  
  def main(args: Array[String]): Unit = {  
    import StringExtensions._  
  
    val greeting = "Hello World"  
    println(greeting.exclaim)           // "Hello World!"  
  
    val madam = "Madam"  
    println(madam.isPalindrome)         // true  
  
    val notPalindrome = "Hello"  
    println(notPalindrome.isPalindrome) // false  
  }  
}
```


- Scala 2 の時代に implicit が導入された
- implicit は "暗黙的" という意味
- implicit は多機能すぎて分かりにくいという批判があった
- Scala 3 では implicit は 6 種類の用語に整理しなおされた