

Day 5 (12-Nov-2025)

Q1. Smart Temperature Converter (Conditionals & Basic Functions)

Code:

```
case class SmartTemperatureConverter() {  
    // def convertTemp(value: Double, scale: String): Double = {  
    //     if (scale.equalsIgnoreCase("f")) {  
    //         (value - 32.0) * 5.0 / 9.0  
    //     } else if (scale.equalsIgnoreCase("c")) {  
    //         value * 9.0 / 5.0 + 32.0  
    //     } else {  
    //         value  
    //     }  
    // }  
  
    def convertTemp(value: Double, scale: String): Double = if (  
        scale.equalsIgnoreCase("f")  
    ) {  
        (value - 32.0) * 5.0 / 9.0  
    } else if (scale.equalsIgnoreCase("c")) {  
        value * 9.0 / 5.0 + 32.0  
    } else {  
        value  
    }  
}  
  
object SmartTemperatureConverterApp extends App {  
    val converter = SmartTemperatureConverter()  
  
    println(converter.convertTemp(0, "C")) // 32.0  
    println(converter.convertTemp(212, "F")) // 100.0  
    println(converter.convertTemp(50, "X")) // 50.0  
}
```

Output:

```

s...    /  // } else {
8...    //   value
9...    // }
10...   // }

11
12  def convertTemp(value: Double, scale: String): Double = if (
13  | scale.equalsIgnoreCase("f")
14  ) {
15  |   (value - 32.0) * 5.0 / 9.0
16  } else if (scale.equalsIgnoreCase("c")) {
17  |   value * 9.0 / 5.0 + 32.0
y.... PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER + ⌂ ...
d... rac@PTPMR707 ScalaHandsonOnQuestions % git pull --rebase origin main
h...
error: could not apply 47edf2a... Add various Scala examples demonstrating functional programming concepts
hint: Resolve all conflicts manually, mark them as resolved with
hint: "git add/rm <conflicting_files>", then run "git rebase --continue".
hint: You can instead skip this commit: run "git rebase --skip".
hint: To abort and get back to the state before "git rebase", run "git rebase --abort".
hint: Disable this message with "git config set advice.mergeConflict false"
Could not apply 47edf2a... # Add various Scala examples demonstrating functional programming concepts
● rac@PTPMR707 ScalaHandsonOnQuestions % git push
  Enumerating objects: 7, done.
  Counting objects: 100% (7/7), done.
  Delta compression using up to 8 threads
  Compressing objects: 100% (3/3), done.
  Writing objects: 100% (4/4), 1.97 KiB | 1.97 MiB/s, done.
  Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/vskumar9/PayodaScalaLanguage.git
  c28b657..9a44a3b main -> main
● rac@PTPMR707 ScalaHandsonOnQuestions % scala SmartTemperatureConverter.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-clj.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
32.0
100.0
50.0
◆ rac@PTPMR707 ScalaHandsonOnQuestions %

```

Q2: Array Mirror Puzzle (Arrays + Loops + Operators)

Code:

```

object ArrayMirrorPuzzle extends App {
  def mirrorArray(arr: Array[Int]): Array[Int] = {
    val n = arr.length
    val mirrored = for (i <- 0 until 2 * n) yield {
      if (i < n) arr(i) // first half: original elements
      else arr(2 * n - 1 - i) // second half: mirrored elements
    }
    mirrored.toArray
  }

  val input = Array(1, 2, 3)
  val mirrored = mirrorArray(input)
  println(mirrored.mkString("Array(", ", ", ", ", ")")) // Array(1, 2, 3, 3, 2, 1)
}

```

Output:

```

7     }
8     mirrored.toArray
9   }
10
11   val input = Array(1, 2, 3)
12   val mirrored = mirrorArray(input)
13   println(mirrored.mkString("Array(", " ", " ", ")")) // Array(1, 2, 3, 3, 2, 1)
14
15 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER + × ⋮ | []
rac@PTPMRT07 ScalaHandsonOnQuestions % git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 1.97 KiB | 1.97 MiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/vskumar9/PayodaScalaLanguage.git
  c28b657..9a4a43b main -> main
rac@PTPMRT07 ScalaHandsonOnQuestions % scala SmartTemperatureConverter.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cli.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
32.0
100.0
50.0
rac@PTPMRT07 ScalaHandsonOnQuestions % scala ArrayMirrorPuzzle.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cli.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Array(1, 2, 3, 3, 2, 1)
rac@PTPMRT07 ScalaHandsonOnQuestions %

```

Q3. Digit Sum using Simple Recursion

Code:

```

object ArrowPuzzle extends App {
  def digitSum(n: Int): Int = {
    if (n == 0) 0 // Base case: no more digits
    else (n % 10) + digitSum(n / 10) // Last digit + sum of remaining digits
  }

  // Example usage:
  println(digitSum(1345)) // Output: 13
}

```

Output:

```

rac@PTPMRT07 ScalaHandsonOnQuestions % scala ArrayMirrorPuzzle.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
.... libexec/scala-clj.jar)
u... WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
c... WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
c... Compiling project (Scala 3.7.3, JVM (25))
s... Compiled project (Scala 3.7.3, JVM (25))
s... Array(1, 2, 3, 3, 2, 1)
s... 
● rac@PTPMRT07 ScalaHandsonOnQuestions % scala ArrowPuzzle.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
.... libexec/scala-clj.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
13
◆ rac@PTPMRT07 ScalaHandsonOnQuestions % fi

```

Q4. Tail Recursive Factorial with Debug Prints

Code:

```

case class TailRecursiveFactorialWithDebugPrints() {
    def factorial(n: Int): Int = {
        @annotation.tailrec
        def factorial(acc: Int, n: Int): Int = {
            println(s"[acc=$acc, n=$n]")
            if (n <= 1) acc
            else factorial(acc * n, n - 1)
        }
        factorial(1, n)
    }
}

object TailRecursiveFactorialApp extends App {
    val number = 5
    val result = TailRecursiveFactorialWithDebugPrints().factorial(number)
}

```

Output:

```

WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
c... Compiling project (Scala 3.7.3, JVM (25))
c... Compiled project (Scala 3.7.3, JVM (25))
13
● rac@PTPMRT07 ScalaHandsonOnQuestions % scala TailRecursiveFactorialWithDebugPrints.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/
libexec/scala-clj.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
[acc=1, n=5]
[acc=5, n=4]
[acc=20, n=3]
[acc=60, n=2]
[acc=120, n=1]
◆ rac@PTPMRT07 ScalaHandsonOnQuestions %

```

Q5. Find Max Element in an Array using Tail Recursion (No Loops)

Code:

```
case class FindMaxElementInAnArrayUsingTailRecursion() {  
    def maxInArray(arr: Array[Int]): Int = {  
        @annotation.tailrec  
        def helper(index: Int, currentMax: Int): Int = {  
            if (index == arr.length) currentMax  
            else {  
                val newMax = if (arr(index) > currentMax) arr(index) else currentMax  
                helper(index + 1, newMax)  
            }  
        }  
        if (arr.isEmpty) throw new IllegalArgumentException("Array cannot be empty")  
        helper(1, arr(0))  
    }  
  
    object FindMaxElementInAnArrayUsingTailRecursion {  
        def main(args: Array[String]): Unit = {  
            val finder = FindMaxElementInAnArrayUsingTailRecursion()  
            val nums = Array(5, 9, 3, 7, 2)  
            val maxElement = finder.maxInArray(nums)  
            println(s"The maximum element in the array is: $maxElement")  
        }  
    }  
}
```

Output:

```
n... WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Compiled project (Scala 3.7.3, JVM (25))  
[acc=1, n=5]  
[acc=5, n=4]  
[acc=20, n=3]  
[acc=60, n=2]  
[acc=120, n=1]  
● racd@PTPMRT07 ScalaHansonOnQuestions % scala FindMaxElementInAnArrayUsingTailRecursion.scala  
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called  
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-clients.jar)  
c... WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Compiled project (Scala 3.7.3, JVM (25))  
The maximum element in the array is: 9  
◆ racd@PTPMRT07 ScalaHansonOnQuestions % █
```

Scala For - Comprehension & Collections - Set 1 (Innovative + Educational)

Q1. Building a 2D Multiplication Table (Nested For - Comprehension)

Code:

```
import scala.io.StdIn._

case class Building2DMultiplicationTable() {
    def multiplicationTable(n: Int): List[String] = {
        //     for {
        //         i <- 1 to n
        //         j <- 1 to n
        //     } yield s"$i x $j = ${i * j}"

        (1 to n).flatMap(i => (1 to n).map(j => s"$i x $j = ${i * j}"))
    }.toList
}

object Building2DMultiplicationTable {
    def main(args: Array[String]): Unit = {
        val tableBuilder = Building2DMultiplicationTable()
        val n = readLine().toInt // Read size of multiplication table
        val table = tableBuilder.multiplicationTable(n)
        table.foreach(println)
    }
}
```

Output:

```
... ● rac@PTPMRT07 10-11-2025 % scala Building2DMultiplicationTable.scala
U... WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
C... WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0
. jar)
S... WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
S...
3
C... Enter a number:
or...
1 x 1 = 1
1 x 2 = 2
1 x 3 = 3
2 x 1 = 2
2 x 2 = 4
2 x 3 = 6
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
✉ rac@PTPMRT07 10-11-2025 %
```

Q2. Combining Two Lists Functionally (Cartesian Product + Filter)

Code:

```
case class CombiningTwoListsFunctionally() {  
  
    def CartesianProduct1(  
        students: List[String],  
        subjects: List[String]  
    ): List[(String, String)] = {  
        for {  
            s <- students  
            sub <- subjects  
            if s.length >= sub.length  
        } yield (s, sub)  
    }  
  
    def CartesianProduct2(  
        students: List[String],  
        subjects: List[String]  
    ): List[(String, String)] = {  
        students.flatMap(s =>  
            subjects  
                .withFilter(sub => s.length >= sub.length)  
                .map(sub => (s, sub))  
        )  
    }  
}  
  
object CombiningTwoListsFunctionally {  
    def main(args: Array[String]): Unit = {  
        val combiner = CombiningTwoListsFunctionally()  
        val students = List("Asha", "Bala", "Chitra")  
        val subjects = List("Math", "Physics")  
        val result = combiner.CartesianProduct1(students, subjects)  
        println("Cartesian Product of Students and Subjects:")  
        result.foreach(println)  
  
        val functionalResult = combiner.CartesianProduct2(students, subjects)  
        println("Cartesian Product: 2 Approach:")  
        functionalResult.foreach(println)  
    }  
}
```

Output:

```
rac@PTPMR707 10-11-2025 % scala Building2DMultiplicationTable.scala
● rac@PTPMR707 10-11-2025 % scala CombiningTwoListsFunctionally.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.1.0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
d... Cartesian Product of Students and Subjects:
(Asha,Math)
(Bala,Math)
(Chitra,Math)
Cartesian Product: 2 Approach:
(Asha,Math)
(Bala,Math)
(Chitra,Math)
◆ rac@PTPMR707 10-11-2025 %
```

Q3. Extracting and Transforming Nested Data (flatMap Insight)

Code:

```
object ExtractingAndTransformingNestedData extends App {

    val departments = List(
        ("IT", List("Ravi", "Meena")),
        ("HR", List("Anita")),
        ("Finance", List("Vijay", "Kiran"))
    )

    // Method 1: For-comprehension
    val result1 = for {
        (dept, employees) <- departments
        emp <- employees
    } yield s"$dept: $emp"

    println("Result using for-comprehension:")
    result1.foreach(println)

    // Method 2: flatMap + map
    val result2 = departments.flatMap { case (dept, employees) =>
        employees.map(emp => s"$dept: $emp")
    }

    println("\nResult using flatMap + map:")
    result2.foreach(println)
}
```

Output:

```

rac@PTPMRT07 10-11-2025 % scala ExtractingAndTransformingNestedData.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Result using for-comprehension:
IT: Ravi
IT: Meena
HR: Anita
Finance: Vijay
Finance: Kiran

Result using flatMap + map:
IT: Ravi
IT: Meena
HR: Anita
Finance: Vijay
Finance: Kiran
◆ rac@PTPMRT07 10-11-2025 %

```

Q4. Functional Filtering and Transformation (Realistic Scenario)

Code:

```

case class Order(id: Int, amount: Double, status: String)

object FunctionalFilteringAndTransformation {
  def main(args: Array[String]): Unit = {
    val orders = List(
      Order(1, 1200.0, "Delivered"),
      Order(2, 250.5, "Pending"),
      Order(3, 980.0, "Delivered"),
      Order(4, 75.0, "Cancelled")
    )

    // Filter completed orders and transform to get their amounts
    val result1 = for {
      Order(id, amount, status) <- orders
      if status == "Delivered"
      if amount > 500
    } yield s"Order #$id -> ₹$amount"

    // Using functional methods
    val result2 = orders
      .withFilter { case Order(_, _, status) => status == "Delivered" }
      .withFilter { case Order(_, amount, _) => amount > 500 }
      .map { case Order(id, amount, _) => s"Order #$id -> ₹$amount" }

    println("Amounts of Completed Orders:")
    result1.foreach(println)
    println("\nAmounts of Completed Orders (Functional):")
    result2.foreach(println)
  }
}

```

```
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala ExtractingAndTransformingNestedData.scala
Finance: Kiran
n... ● rac@PTPMRT07 10-11-2025 % scala FunctionalFilteringAndTransformation.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Amounts of Completed Orders:
Order #1 -> ₹1200.0
Order #3 -> ₹980.0

Amounts of Completed Orders (Functional):
Order #1 -> ₹1200.0
Order #3 -> ₹980.0
◆ rac@PTPMRT07 10-11-2025 %
```

Q5. Word Frequency Counter using Collections + For Comprehension

Code:

```
object WordFrequencyCounte extends App {
  val lines = List(
    "Scala is powerful",
    "Scala is concise",
    "Functional programming is powerful"
  )

  // 1) Flatten words via for-comprehension
  val words = for {
    line <- lines
    word <- line.split("\s+")
  } yield word

  // 2) Group and count
  val freq1: Map[String, Int] =
    words.groupBy(identity).view.mapValues(_.size).toMap

  // Alternative approach using functional methods
  val freq2 = lines
    .flatMap(_.split("\s+"))
    .groupBy(identity)
    .view
    .mapValues(_.size)
    .toMap
```

```

// Print results
println(freq1)
println(freq2)
// Map(Scala -> 2, is -> 3, powerful -> 2, concise -> 1, Functional -> 1, programming
-> 1)

}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala FunctionalFilteringAndTransformation.scala
Order #1 -> ₹1200.0
Order #3 -> ₹980.0

Amounts of Completed Orders (Functional):
Order #1 -> ₹1200.0
Order #3 -> ₹980.0
● rac@PTPMRT07 10-11-2025 % scala WordFrequencyCounte.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
HashMap(is -> 3, Scala -> 2, programming -> 1, powerful -> 2, concise -> 1, Functional -> 1)
HashMap(programming -> 1, powerful -> 2, concise -> 1, Functional -> 1)
◆ rac@PTP

```

Set: Scala Operators — Hands - On Learning

Q1. The Mystery of `:+` and `+`:

Code:

```

object `TheMysteryOf:+And+:` extends App {
  val nums = List(2, 4, 6)
  val list1 = List(1, 2, 3)

  // Using `:+` to append an element to the end of a list
  val appendedList = nums :+ 8
  println(
    s"Appended List: $appendedList"
  ) // Output: Appended List: List(1, 2, 3, 8)

  // Using `+:` to prepend an element to the beginning of a list
  val prependedList = 0 +: nums
  println(
    s"Prepended List: $prependedList"
  ) // Output: Prepended List: List(0, 1, 2, 3)
}

```

```

val appendAndprepend = 0 +: nums :+ 8
println(
  s"Append and Prepend: $appendAndprepend"
) // Output: Append and Prepend: List(0, 2, 4, 6, 8)

// Using `++` to concatenate two lists
val concatenatedList = list1 ++ nums
println(
  s"Concatenated List: $concatenatedList"
) // Output: Concatenated List: List(1, 2, 3, 4, 5, 6)

}

```

Code:

```

rac@PTPMRT07 10-11-2025 % scala WordFrequencyCounte.scala
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
HashMap(is -> 3, Scala -> 2, programming -> 1, powerful -> 2, concise -> 1, Functional -> 1)
HashMap(is -> 3, Scala -> 2, programming -> 1, powerful -> 2, concise -> 1, Functional -> 1)
● rac@PTPMRT07 10-11-2025 % scala TheMysteryOf:+And+:scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Appended List: List(2, 4, 6, 8)
Prepended List: List(0, 2, 4, 6)
Append and Prepend: List(0, 2, 4, 6, 8)
Concatenated List: List(1, 2, 3, 2, 4, 6)
✉ rac@PTPMRT07 10-11-2025 %

```

Q2. Operator Overload - Reverse Thinking

Code:

```

case class Counter(value: Int) {

  // Add two Counter objects
  def +(that: Counter): Int = this.value + that.value

  // Add Counter with Int
  def +(that: Int): Int = this.value + that
}

object OperatorOverload extends App {

  val a = Counter(5)
  val b = Counter(7)
}

```

```

    println(a + b) // 12
    println(a + 10) // 15
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala TheMysteryOf:+And++.scala
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Appended List: List(2, 4, 6, 8)
Prepended List: List(0, 2, 4, 6)
Append and Prepend: List(0, 2, 4, 6, 8)
Concatenated List: List(1, 2, 3, 2, 4, 6)
● rac@PTPMRT07 10-11-2025 % scala OperatorOverload.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
12
15
✉ rac@PTPMRT07 10-11-2025 %

```

Q3. The Curious Case of :: and List Construction

Code:

```

object `TheCuriousCaseOf::AndListConstruction` extends App {
    // Building the list [1, 2, 3, 4] using only :: and Nil
    val myList = 1 :: 2 :: 3 :: 4 :: Nil

    // Print the constructed list
    println("Constructed List: " + myList)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala OperatorOverload.scala
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
12
15
● rac@PTPMRT07 10-11-2025 % scala TheCuriousCaseOf::AndListConstruction.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Constructed List: List(1, 2, 3, 4)
✉ rac@PTPMRT07 10-11-2025 %

```

Q4. ++ vs :: — The Merge Game

Code:

```

object `++vs:::—TheMergeGame` extends App {

```

```

// Method to merge two lists using ++ operator
def mergeUsingPlusPlus[A](list1: List[A], list2: List[A]): List[A] = {
    list1 ++ list2
}

// Method to merge two lists using :::: operator
def mergeUsingColonColonColon[A](list1: List[A], list2: List[A]): List[A] = {
    list1 :::: list2
}

val listA = List(1, 2)
val listB = List(3, 4)

// Merging using ++
val mergedPlusPlus = mergeUsingPlusPlus(listA, listB)
println(s"Merged using ++ : $mergedPlusPlus")

// Merging using :::
val mergedColonColonColon = mergeUsingColonColonColon(listA, listB)
println(s"Merged using :::: : $mergedColonColonColon")
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala TheCuriousCaseOf::AndListConstruction.scala
...
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
U... Constructed List: List(1, 2, 3, 4)
o rac@PTPMRT07 10-11-2025 % scala ++vs::::-TheMergeGame.scala ` 
c... bquote> ;
b... bquote>
s... ● rac@PTPMRT07 10-11-2025 % scala ++vs::::-TheMergeGame.scala
s... WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
c... WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0
. jar)
or... WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Merged using ++ : List(1, 2, 3, 4)
Merged using :::: : List(1, 2, 3, 4)
◆ rac@PTPMRT07 10-11-2025 %

```

Q5. The -> Arrow Puzzle

Code:

```

object `The->ArrowPuzzle` extends App {
    val animals = Map(
        "dog" -> "bark",
        "cat" -> "meow",
    )
}

```

```

    "cow" -> "moo"
  )

// 1. Add lion
val updatedAnimals = animals + ("lion" -> "roar")

// 2. Retrieve cow sound
println("Cow sound: " + updatedAnimals("cow"))

// 3. Safe lookup for tiger
println("Tiger sound: " + updatedAnimals.getOrElse("tiger", "Unknown animal"))

}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala ++vs:::-TheMergeGame.scala
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Merged using ++ : List(1, 2, 3, 4)
Merged using :: : List(1, 2, 3, 4)
● rac@PTPMRT07 10-11-2025 % scala TheArrowPuzzle.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Cow sound: moo
Tiger sound: Unknown animal
◆ rac@PTPMRT07 10-11-2025 %

```

scala Functional Concepts

Q1. Mood Transformer (Higher - Order Function + Closure)

Code:

```

object MoodTransformer extends App {
  def moodChanger(prefix: String): String => String = {
    word => s"$prefix-$word-$prefix"
  }

  val happyMood = moodChanger("happy")
  val angryMood = moodChanger("angry")
  println(happyMood("day")) // happy-day-happy
  println(angryMood("crowd")) // angry-crowd-angry
  println(moodChanger("joyful")("sunshine")) // joyful-sunshine-joyful
}

```

```
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala TheArrowPuzzle.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Cow sound: moo
Tiger sound: Unknown animal
● rac@PTPMRT07 10-11-2025 % scala MoodTransformer.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
happy-day-happy
angry-crowd-angry
joyful-sunshine-joyful
◆ rac@PTPMRT07 10-11-2025 %
```

Q2. Sentence Pipeline (Function Composition with Transformation Chain)

Code:

```
object SentencePipeline extends App {
    val trimSpaces: String => String = _.trim
    val toLower: String => String = _.toLowerCase
    val capitalizeFirst: String => String = s => s.head.toUpperCase + s.tail

    val processSentence: String => String =
        trimSpaces andThen toLower andThen capitalizeFirst

    val messy = "    HeLlo WOrld    "
    println(processSentence(messy)) // Hello world
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala MoodTransformer.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
happy-day-happy
angry-crowd-angry
joyful-sunshine-joyful
● rac@PTPMRT07 10-11-2025 % scala SentencePipeline.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Warning: there was 1 deprecation warning; re-run with -deprecation for details
Compiled project (Scala 3.7.3, JVM (25))
Hello world
◆ rac@PTPMRT07 10-11-2025 %
```

Q3. Delayed Greeter (Partial Application + Time Control)

Code:

```
object DelayedGreeter extends App {  
    def delayedMessage(delayMs: Int)(message: String): Unit = {  
        Thread.sleep(delayMs)  
        println(message)  
    }  
  
    val oneSecondSay = delayedMessage(1000)  
  
    oneSecondSay("Hello...")  
    oneSecondSay("How are you?")  
    oneSecondSay("This message is delayed each time!")  
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala SentencePipeline.scala  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
o... Compiling project (Scala 3.7.3, JVM (25))  
s... Warning: there was 1 deprecation warning; re-run with -deprecation for details  
s... Compiled project (Scala 3.7.3, JVM (25))  
Hello world  
● rac@PTPMRT07 10-11-2025 % scala DelayedGreeter.scala  
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called  
n... WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-c  
.jar)  
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Compiled project (Scala 3.7.3, JVM (25))  
Hello...  
How are you?  
This message is delayed each time!  
◆ rac@PTPMRT07 10-11-2025 %
```

Q4. Adaptive Discount (Higher - Order Function Returning Strategy)

Code:

```
object AdaptiveDiscount extends App {  
    def discountStrategy(memberType: String): Double => Double = {  
        memberType.toLowerCase match {  
            case "gold"    => amount => amount * 0.8  
            case "silver"  => amount => amount * 0.9  
            case _          => amount => amount  
        }  
    }  
}
```

```

val goldDiscount = discountStrategy("gold")
println(goldDiscount(1000)) // 800.0

val silverDiscount = discountStrategy("silver")
println(silverDiscount(1000)) // 900.0

val normalDiscount = discountStrategy("regular")
println(normalDiscount(1000)) // 1000.0

}

```

Output:

```

$... rac@PTPMRT07 10-11-2025 % scala DelayedGreeter.scala
s... Compiling project (Scala 3.7.3, JVM (25))
c... Compiled project (Scala 3.7.3, JVM (25))
Hello...
or... How are you?
This message is delayed each time!
● rac@PTPMRT07 10-11-2025 % scala AdaptiveDiscount.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
800.0
900.0
1000.0
◆ rac@PTPMRT07 10-11-2025 %

```

Q5. Intentional Crasher (Partial Function with Safety Net)

Code:

```

object IntentionalCrasher extends App {
  val safeDivide: PartialFunction[Int, String] = {
    case x if x != 0 => s"Result: ${100 / x}"
  }

  val safe = safeDivide.lift

  println(safe(10)) // Some("Result: 10")
  println(safe(0)) // None
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala AdaptiveDiscount.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
800.0
900.0
1000.0
● rac@PTPMRT07 10-11-2025 % scala IntentionalCrasher.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Some(Result: 10)
None
◆ rac@PTPMRT07 10-11-2025 %

```

Set: Currying · Pattern Matching · Option · Either — Functional Mindset

Q1. The Personalized Calculator (Currying + Partial Application)

Code:

```

object ThePersonalizedCalculator extends App {
  def calculate(op: String)(x: Int, y: Int): Int = op match {
    case "add" => x + y
    case "sub" => x - y
    case "mul" => x * y
    case "div" => x / y
    case _        => throw new IllegalArgumentException("Unknown operation")
  }

  val add = calculate("add")
  val multiply = calculate("mul")

  println(add(10, 5))      // 15
  println(multiply(3, 4))   // 12
  println(calculate("sub")(20, 8)) // 12
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala IntentionalCrasher.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Some(Result: 10)
None
● rac@PTPMRT07 10-11-2025 % scala ThePersonalizedCalculator.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
15
12
12
◆ rac@PTPMRT07 10-11-2025 %

```

Q2. The Shape Analyzer (Pattern Matching on Case Classes)

Code:

```

object TheShapeAnalyzer extends App {
    case class Circle(r: Double)
    case class Rectangle(w: Double, h: Double)
    case class Triangle(a: Double, b: Double, c: Double)
    case class Square(s: Double)
    case class Ellipse(a: Double, b: Double)

    def area(shape: Any): Double = shape match {
        case Circle(r)          => Math.PI * r * r
        case Rectangle(w, h)    => w * h
        case Triangle(a, b, c)  =>
            val s = (a + b + c) / 2
            Math.sqrt(s * (s - a) * (s - b) * (s - c))
        case Square(s)          => s * s
        case Ellipse(a, b)      => Math.PI * a * b
        case _                  => -1.0
    }

    println(area(Circle(3))) // 28.274333882308138
    println(area(Rectangle(4, 5))) // 20.0
    println(area(Triangle(3, 4, 5))) // 6.0
    println(area(Square(4))) // 16.0
    println(area(Ellipse(3, 4))) // 37.69911184307752
    println(area("unknown")) // -1.0
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala ThePersonalizedCalculator.scala
12
12
● rac@PTPMRT07 10-11-2025 % scala TheShapeAnalyzer.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
28.274333882308138
20.0
6.0
16.0
37.69911184307752
-1.0
◆ rac@PTPMRT07 10-11-2025 %

```

Q3. The Safe Divider (Option)

Code:

```

object TheSafeDivide extends App {
  def safeDivide(x: Int, y: Int): Option[Int] = {
    if (y == 0) None
    else Some(x / y)
  }

  val result = safeDivide(10, 0).getOrElse(-1)
  println(result) // -1
  val result2 = safeDivide(10, 2).getOrElse(-1)
  println(result2) // 5
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala TheShapeAnalyzer.scala
28.274333882308138
20.0
6.0
16.0
37.69911184307752
-1.0
● rac@PTPMRT07 10-11-2025 % scala TheSafeDivide.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
-1
5
◆ rac@PTPMRT07 10-11-2025 %

```

Q4. The Login Validator (Either for Error Handling)

Code:

```

object TheLoginValidator extends App {
  def validateLogin(
    username: String,
    password: String
  ): Either[String, String] = {
    if (username.isEmpty) Left("Username missing")
    else if (password.isEmpty) Left("Password missing")
    else Right("Login successful")
  }

  println(validateLogin("", "123"))
  println(validateLogin("user", ""))
  println(validateLogin("user", "123"))

}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala TheSafeDivide.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
-1
5
● rac@PTPMRT07 10-11-2025 % scala TheLoginValidator.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Left.Username missing)
Left>Password missing)
Right(Login successful)
◆ rac@PTPMRT07 10-11-2025 %

```

Q5. The Smart Parser (Pattern Matching on Option & Either)

Code:

```

object TheSmartParser extends App {
  def safeDivide(x: Int, y: Int): Option[Int] =
    if (y == 0) None else Some(x / y)

  def parseAndDivide(input: String): Either[String, Int] = {
    input.toIntOption match {
      case None =>
        Left("Invalid number")

      case Some(number) =>

```

```

        safeDivide(100, number) match {
            case None          => Left("Division by zero")
            case Some(result) => Right(result)
        }
    }

}

println(parseAndDivide("25")) // Right(4)
println(parseAndDivide("0")) // Left(Division by zero)
println(parseAndDivide("abc")) // Left(Invalid number)
println(parseAndDivide("")) // Left(Invalid number)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala TheLoginValidator.scala
Compiled project (Scala 3.7.3, JVM (25))
LeftUsername missing)
Left>Password missing)
Right(Login successful)
● rac@PTPMRT07 10-11-2025 % scala TheSmartParser.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scalajs.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Right(4)
Left(Division by zero)
Left(Invalid number)
Left(Invalid number)
◆ rac@PTPMRT07 10-11-2025 %

```

Set: Advanced Scala — Apply/Unapply · Code Blocks · Lazy Evaluation

Q1. The apply Evaluator — Code Block as Input

Code:

```

object TheApplyEvaluator extends App {
    object Evaluator {
        def apply(block: => Any): Unit = {
            println("Evaluating block...")
            val result = block
            println(s"Result = $result")
        }
    }
}

```

```
Evaluator {  
    val x = 5  
    val y = 3  
    x * y + 2  
}  
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala TheA
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
[error] TheA is not a scala sub-command and it is not a valid path to an input file or directory.
Try viewing the relevant help to see the list of available sub-commands and options.
scala --help
● rac@PTPMRT07 10-11-2025 % scala TheApplyEvaluator.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Evaluating block...
Result = 17
rac@PTPMRT07 10-11-2025 %
```

Q2. The Dual Nature Object — Apply & Unapply Mirror

Code:

```

}

val invalidEmail = Email("", "")
invalidEmail match {
  case Email(user, domain) => println(s"User: $user, Domain: $domain")
  case _                      => println("Invalid email")
}

}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala TheApplyEvaluator.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
c... Evaluating block...
Result = 17
● rac@PTPMRT07 10-11-2025 % scala TheDualNatureObject.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
alice@mail.com
User: alice, Domain: mail.com
Invalid email
◆ rac@PTPMRT07 10-11-2025 %

```

Q3. Lazy Evaluation with Triggers

Code:

```

class LazyCounter {
  private var computeCount = 0

  lazy val value: Int = {
    computeCount += 1
    println("Computing value...")
    42
  }

  def getCount: Int = computeCount
}

object LazyEvaluationWithTriggers extends App {
  val c = new LazyCounter
}

```

```

    println("Before first access")
    println(c.value)

    println("Access again")
    println(c.value)

    println("Compute count: " + c.getCount)
}

//Before first access
// Computing value...
// 42
// Access again
// 42
// Compute count: 1
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala LazyEvaluationWithTriggers.scala
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/maven2/org/scala-lang/scala3-library_3/3.7.3/scala3-library_3-3.7.3.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Before first access
Computing value...
42
Access again
42
Compute count: 1
◆ rac@PTPMRT07 10-11-2025 %

```

Q4. Unapply Chain — Nested Pattern Matching

Code:

```

object UnapplyChainNestedPatternMatching extends App {
  case class Address(city: String, pincode: Int)
  case class Person(name: String, address: Address)

  val p = Person("Ravi", Address("Chennai", 600001))

  p match {
    case Person(_, Address(city, pin)) if city.startsWith("C") =>
      println(s"$city - $pin")
    case _ =>
      println("No match")
  }
}

```

```
}
```

```
}
```

Output:

```
rac@PTPMRT07: ~ % scala LazyEvaluationWithTriggers.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Before first access
Computing value...
42
Access again
42
Compute count: 1
● rac@PTPMRT07: ~ % scala UnapplyChainNestedPatternMatching.scala
'WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Chennai - 600001
◆ rac@PTPMRT07: ~ %
```

Q5. Lazy Pipeline Builder (Lazy Evaluation + Apply Block + Closure Retention)

Code:

```
object LazyPipelineBuilder extends App {
  object Pipeline {
    def apply[T](block: => T): LazyPipeline[T] = new LazyPipeline(block)
  }

  class LazyPipeline[T](block: => T) {
    lazy val result: T = block
    def map[R](f: T => R): LazyPipeline[R] =
      Pipeline { f(result) } // defer execution until .result is accessed
  }
}

// Usage
val p = Pipeline {
  println("Step 1: Preparing data")
  List(1, 2, 3)
}.map { xs =>
  println("Step 2: Transforming data")
  xs.map(_ * 2)
}

println("Before accessing pipeline...")
println("Result: " + p.result)
}
```

Output:

```
c... ● rac@PTPMRT07 10-11-2025 % scala LazyPipelineBuilder.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/maven2/org/scalatest/lang/scala3-library_3/3.7.3/scala3-library_3-3.7.3.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Before accessing pipeline...
Step 1: Preparing data
Step 2: Transforming data
Result: List(2, 4, 6)
◆ rac@PTPMRT07 10-11-2025 %
```

Set: Operator Overloading & Implicits — Innovative Hands - On Questions

Q1. Create a Vector Algebra DSL using Operator Overloading

Code:

```
object CreateVectorAlgebraDSL_UsingOperatorOverloading extends App {
  case class Vec2D(x: Int, y: Int) {
    def +(that: Vec2D): Vec2D = Vec2D(this.x + that.x, this.y + that.y)
    def -(that: Vec2D): Vec2D = Vec2D(this.x - that.x, this.y - that.y)
    def *(scalar: Int): Vec2D = Vec2D(this.x * scalar, this.y * scalar)
    override def toString: String = s"Vec2D($x,$y)"
  }

  implicit class IntOps(n: Int) {
    def *(v: Vec2D): Vec2D = v * n
  }

  val v1 = Vec2D(2, 3)
  val v2 = Vec2D(4, 1)

  println(v1 + v2) // Vec2D(6,4)
  println(v1 - v2) // Vec2D(-2,2)
  println(v1 * 3) // Vec2D(6,9)
  println(3 * v1) // Vec2D(6,9)

  val result = 2 * v1 + v2 * 3 - v1
  println(result) // Vec2D(14,6)
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala LazyPipelineBuilder.scala
Step 1: Preparing data
Step 2: Transforming data
Result: List(2, 4, 6)
● rac@PTPMRT07 10-11-2025 % scala CreateVectorAlgebraDSL_UsingOperatorOverloading.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Vec2D(6,4)
Vec2D(-2,2)
Vec2D(6,9)
Vec2D(6,9)
Vec2D(14,6)
◆ rac@PTPMRT07 10-11-2025 %
```

Q2. Implicit Ordering with Custom Operator

Code:

```
object ImplicitOrderingWithCustomOperator extends App {
  case class Person(name: String, age: Int)

  implicit class PersonOrdering(p: Person) {
    def <(other: Person): Boolean = p.age < other.age
    def >(other: Person): Boolean = p.age > other.age
    def <=(other: Person): Boolean = p.age <= other.age
    def >=(other: Person): Boolean = p.age >= other.age
  }

  val p1 = Person("Ravi", 25)
  val p2 = Person("Meena", 30)

  println(p1 < p2) // true
  println(p1 >= p2) // false

  // shows normal if-condition usage
  if (p1 < p2) {
    println("Meena is older")
  }
}
```

Output:

```

rac@PTPMRT07 10-11-2025 % scala CreateVectorAlgebraDSL_UsingOperatorOverloading.scala
Vec2D(6,4)
Vec2D(-2,2)
Vec2D(6,9)
Vec2D(6,9)
Vec2D(14,6)
● rac@PTPMRT07 10-11-2025 % scala ImplicitOrderingWithCustomOperator.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
true
false
Meena is older
◆ rac@PTPMRT07 10-11-2025 %

```

Q3. String Enrichment via Implicit Class + Operator

Code:

```

object StringEnrichmentViaImplicitClassPlusOperator extends App {

    implicit class RichString(private val s: String) {
        def *(n: Int): String =
            if (n <= 0) "" else List.fill(n)(s).mkString
        // new StringOps(s).*(n)

        def ~(other: String): String = s + " " + other
    }

    println("Hi" * 3)           // HiHiHi
    println("Hello" ~ "World") // Hello World
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala ImplicitOrderingWithCustomOperator.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
true
false
Meena is older
● rac@PTPMRT07 10-11-2025 % scala StringEnrichmentViaImplicitClassPlusOperator.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-0
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
HihiHi
Hello World
◆ rac@PTPMRT07 10-11-2025 %

```

Q4. Implicit Parameter for Precision in Operator

Code:

```

object ImplicitParameterForPrecisionInOperator extends App {
    implicit val roundingPrecision: Double = 0.05

    case class Money(amount: Double) {

        private def roundToPrecision(
            value: Double
        )(implicit precision: Double): Double = {
            val rounded = Math.round(value / precision) * precision
            BigDecimal(rounded).setScale(2, BigDecimal.RoundingMode.HALF_UP).toDouble
        }

        def +(other: Money)(implicit precision: Double): Money =
            Money(roundToPrecision(this.amount + other.amount))

        def -(other: Money)(implicit precision: Double): Money =
            Money(roundToPrecision(this.amount - other.amount))

        override def toString: String = f"Money($amount%.2f)"
    }

    val m1 = Money(10.23)
    val m2 = Money(5.19)

    println(m1 + m2) // Money(15.40)
    println(m1 - m2) // Money(5.05)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala StringEnrichmentViaImplicitClassPlusOperator.scala
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
HiHiHi
Hello World
c... ● rac@PTPMRT07 10-11-2025 % scala ImplicitParameterForPrecisionInOperator.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Money(15.40)
Money(5.05)
◆ rac@PTPMRT07 10-11-2025 %

```

Q5. Chained Implicit Conversions — Building an Int to Rational Bridge

Code:

```
object ChainedImplicitConversions extends App {  
    case class Rational(n: Int, d: Int) {  
        require(d != 0, "Denominator cannot be zero")  
  
        def /(that: Rational): Rational =  
            Rational(this.n * that.d, this.d * that.n)  
  
        override def toString: String = s"Rational($n,$d)"  
    }  
  
    implicit def intToRational(x: Int): Rational = Rational(x, 1)  
  
    val r = 1 / Rational(2, 3)  
    println(r) // Rational(3,2)  
  
    // Normal Int / Int still works - it's not ambiguous  
    val normal = 6 / 2  
    println(normal) // 3  
  
}
```

Output:

```
U... rac@PTPMRT07 10-11-2025 % scala ImplicitParameterForPrecisionInOperator.scala  
C... WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
S... Compiling project (Scala 3.7.3, JVM (25))  
S... Compiled project (Scala 3.7.3, JVM (25))  
S... Money(15,40)  
S... Money(5,05)  
C... ● rac@PTPMRT07 10-11-2025 % scala ChainedImplicitConversions.scala  
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called  
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)  
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Warning: there was 1 feature warning; re-run with -feature for details  
Compiled project (Scala 3.7.3, JVM (25))  
Rational(3,2)  
3  
◆ rac@PTPMRT07 10-11-2025 %
```

Set: Exception Handling + Function Composition + Collections Operations

Q1. Safe Division — Functional Exception Handling

Code:

```
object SafeDivisionFunctionalExceptionHandling extends App {  
  def safeDivide(a: Int, b: Int): Either[String, Double] =  
    if (b == 0) Left("Division by zero")  
    else Right(a.toDouble / b)  
  
  val pairs = List((10, 2), (5, 0), (8, 4))  
  
  val results = pairs.map { case (a, b) => safeDivide(a, b) }  
  
  val valid = results.collect { case Right(value) => value }  
  val errors = results.collect { case Left(err) => err }  
  
  println(s"Valid results: $valid")  
  println(s"Errors: $errors")  
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala ChainedImplicitConversions.scala  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Warning: there was 1 feature warning; re-run with -feature for details  
Compiled project (Scala 3.7.3, JVM (25))  
Rational(3,2)  
3  
● rac@PTPMRT07 10-11-2025 % scala SafeDivisionFunctionalExceptionHandling.scala  
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called  
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-compiler.jar)  
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Compiled project (Scala 3.7.3, JVM (25))  
Valid results: List(5.0, 2.0)  
Errors: List(Division by zero)  
△ rac@PTPMRT07 10-11-2025 %
```

Q2. Function Pipelines — Compose vs andThen

Code:

```
object FunctionPipelinesComposeVSandThen extends App {  
  val trim: String => String = _.trim  
  val toInt: String => Int = _.toInt  
  val doubleIt: Int => Int = _ * 2  
  
  val pipeline1 = trim andThen toInt andThen doubleIt  
  val pipeline2 = doubleIt compose toInt compose trim
```

```

    println(pipeline1(" 21 ")) // 42
    println(pipeline2(" 40 ")) // 80
}

```

Output:

```

d... rac@PTPMRT07 10-11-2025 % scala SafeDivisionFunctionalExceptionHandling.scala
n... WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
n... WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Valid results: List(5.0, 2.0)
Errors: List(Division by zero)
o... ● rac@PTPMRT07 10-11-2025 % scala FunctionPipelinesComposeVSandThen.scala
s... WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cl
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
n... Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
42
80
◆ rac@PTPMRT07 10-11-2025 %

```

Q3. Collections + Partial Functions — Selective Transformations

Code:

```

object CollectionsPartialFunctionsSelectiveTransformation extends App {
  val items = List(1, "apple", 3.5, "banana", 42)

  val doubledInts = items.collect { case i: Int =>
    i * 2
  }

  println(doubledInts)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala FunctionPipelinesComposeVSandThen.scala
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
42
80
● rac@PTPMRT07 10-11-2025 % scala CollectionsPartialFunctionsSelectiveTransformation.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cl
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
List(2, 84)
◆ rac@PTPMRT07 10-11-2025 %

```

Q4. Retry Logic — Functional Exception Recovery

Code:

```
object RetryLogicFunctionalExceptionRecovery extends App {  
  def fetchData(): Int = {  
    val n = scala.util.Random.nextInt(3)  
    if (n == 0) throw new RuntimeException("Network fail")  
    n  
  }  
  
  def retry[T](times: Int)(op: => T): Option[T] = {  
    try {  
      Some(op)  
    } catch {  
      case _: Throwable =>  
        if (times > 1) retry(times - 1)(op)  
        else None  
    }  
  }  
  
  val data = retry(3)(fetchData())  
  println(data)  
}
```

Output:

```
rac@PTPMRT07 10-11-2025 % scala CollectionsPartialFunctionsSelectiveTransformation.scala  
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-client.jar)  
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Compiled project (Scala 3.7.3, JVM (25))  
List(2, 84)  
● rac@PTPMRT07 10-11-2025 % scala RetryLogicFunctionalExceptionRecovery.scala  
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called  
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-client.jar)  
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$  
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release  
Compiling project (Scala 3.7.3, JVM (25))  
Compiled project (Scala 3.7.3, JVM (25))  
Some(1)  
△ rac@PTPMRT07 10-11-2025 %
```

Q5. Map – FlatMap Composition — Clean Data Pipeline

Code:

```
object MapFlatMapCompositionCleanDataPipeline extends App {
```

```

val data = List("10", "20", "x", "30")

val result = data
  .map(x => scala.util.Try(x.toInt))
  .flatMap(_.toOption)
  .map(n => n * n)

println(result) // List(100, 400, 900)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala RetryLogicFunctionalExceptionRecovery.scala
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cl
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Some(1)
● rac@PTPMRT07 10-11-2025 % scala MapFlatMapCompositionCleanDataPipeline.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cl
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
List(100, 400, 900)
◆ rac@PTPMRT07 10-11-2025 %

```

Set: Futures & Asynchronous Programming — Hands - On Questions

Q1. Future Sequencing — Parallel vs Sequential

Code:

```

import scala.concurrent._
import scala.concurrent.ExecutionContext.Implicits.global
import scala.concurrent.duration._

object FutureSequencingParallelVSSequential {

  def task(name: String, delay: Int): Future[String] = Future {
    Thread.sleep(delay)
    s"$name done"
  }

  def time[A](label: String)(block: => Future[A]): Unit = {
    val start = System.currentTimeMillis()
    val result = Await.result(block, 10.seconds)
  }
}

```

```

val end = System.currentTimeMillis()
println(s"[$label] Result: $result")
println(s"[$label] Time taken: ${end - start} ms\n")
}

@main def run(): Unit = {

  println("Starting sequential analysis...")

  // 1. Sequential execution
  time("Sequential") {
    task("A", 1000).flatMap { a =>
      task("B", 1500).flatMap { b =>
        task("C", 1200).map { c =>
          List(a, b, c)
        }
      }
    }
  }
}

  println("Starting parallel analysis...")

  // 2. Parallel execution
  time("Parallel") {
    Future.sequence(List(
      task("A2", 1000),
      task("B2", 1500),
      task("C2", 1200)
    ))
  }
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala MapFlatMapCompositionCleanDataPipeline.scala
● rac@PTPMRT07 10-11-2025 % scala FutureSequencingParallelVSSequential.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cl
.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Starting sequential analysis...
[Sequential] Result: List(A done, B done, C done)
[Sequential] Time taken: 3746 ms

Starting parallel analysis...
[Parallel] Result: List(A2 done, B2 done, C2 done)
[Parallel] Time taken: 1507 ms
◆ rac@PTPMRT07 10-11-2025 %

```

Q2. Safe Composition — Future with Recover and RecoverWith

Code:

```

import scala.concurrent._
import scala.concurrent.ExecutionContext.Implicits.global
import scala.concurrent.duration._

def riskyOperation(): Future[Int] = Future {
  val n = scala.util.Random.nextInt(3)
  if (n == 0) throw new RuntimeException("Failed!")
  n
}

@main def SafeCompositionRecoverDemo(): Unit = {
  // 1) recover: fallback VALUE -1
  val f1: Future[Int] =
    riskyOperation().recover { case _: Throwable => -1 }

  val r1 = Await.result(f1, 10.seconds)
  println(s"recover result = $r1")

  // 2) recoverWith: fallback FUTURE (retry once), then final fallback -1
  val f2: Future[Int] =
    riskyOperation()
      .recoverWith { case _: Throwable => riskyOperation() } // retry once
      .recover { case _: Throwable => -1 }                      // final fallback

  val r2 = Await.result(f2, 20.seconds)
  println(s"recoverWith (retry once) result = $r2")
}

```

Output:

```
rac@PTPMRT07 10-11-2025 % scala FutureSequencingParallelVSSequential.scala
[Sequential] Time taken: 3746 ms

Starting parallel analysis...
[Parallel] Result: List(A2 done, B2 done, C2 done)
[Parallel] Time taken: 1507 ms

● rac@PTPMRT07 10-11-2025 % scala SafeCompositionFutureWithRecoverAndRecoverWith.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-client.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
recover result = 2
recoverWith (retry once) result = 2
◆ rac@PTPMRT07 10-11-2025 %
```

Q3. Chaining Dependent Futures — Function Composition

Code:

```
import scala.concurrent._
import scala.concurrent.ExecutionContext.Implicits.global
import scala.concurrent.duration._
import scala.util.{Success, Failure}

def getUser(id: Int): Future[String] =
  Future { s"User$id" }

def getOrders(user: String): Future[List[String]] =
  Future { List(s"$user-order1", s"$user-order2") }

def getOrderTotal(order: String): Future[Double] =
  Future { scala.util.Random.between(10.0, 100.0) }

@main def ChainingDependentFuturesDemo(): Unit = {

  // Blocking
  // for-comprehension version
  val totalF: Future[Double] = for {
    user <- getUser(42)
    orders <- getOrders(user)
    totals <- Future.traverse(orders)(getOrderTotal)
  } yield totals.sum

  val total1 = Await.result(totalF, 5.seconds)
  println(s"Total amount for user 42 (for-comp) = $total1")
```

```

// flatMap version

val totalF2: Future[Double] =
  getUser(42).flatMap { user =>
    getOrders(user).flatMap { orders =>
      Future.traverse(orders)(getOrderTotal).map(_.sum)
    }
  }

val total2 = Await.result(totalF2, 5.seconds)
println(s"Total amount for user 42 (flatMap) = $total2")

// Non-blocking (callbacks)

totalF2.onComplete {
  case Success(total) => println(s"Total amount for user 42 = $total")
  case Failure(e)      => println(s"Failed to compute total: ${e.getMessage}")
}

// Keep the app alive long enough for the callback (demo-only)
Await.ready(totalF2, 5.seconds)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala SafeCompositionFutureWithRecoverAndRecoverWith.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
recover result = 2
recoverWith (retry once) result = 2
● rac@PTPMRT07 10-11-2025 % scala ChainingDependentFuturesFunctionComposition.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-client.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Total amount for user 42 (for-comp) = 64.28329174307896
Total amount for user 42 (flatMap) = 91.08188790925797
Total amount for user 42 = 91.08188790925797
◆ rac@PTPMRT07 10-11-2025 %

```

Q4. Combine Futures with Dependencies — MapN Style

Code:

```

import scala.concurrent._
import scala.concurrent.ExecutionContext

object CombineFuturesWithDependenciesMapNStyle extends App {

```

```

implicit val ec: ExecutionContext = ExecutionContext.Implicits.global

val f1 = Future { Thread.sleep(1000); 10 }
val f2 = Future { Thread.sleep(800); 20 }
val f3 = Future { Thread.sleep(500); 30 }

// 1: zip + map (cats-style mapN without libraries)
val combined1: Future[String] =
  f1.zip(f2).zip(f3).map { case ((a, b), c) =>
    val sum = a + b + c
    val avg = sum / 3
    s"Sum = $sum, Average = $avg"
  }
combined1.foreach(println)

// 2: Future.sequence + map
val combined2: Future[String] =
  Future.sequence(List(f1, f2, f3)).map { xs =>
    val sum = xs.sum
    val avg = sum / xs.size
    s"Sum = $sum, Average = $avg"
  }
combined2.foreach(println)

Thread.sleep(1500)
}

```

Output:

```

rac@PTPMRT07 10-11-2025 % scala ChainingDependentFuturesFunctionComposition.scala
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Total amount for user 42 (for-comp) = 64.28329174307896
Total amount for user 42 (flatMap) = 91.08188790925797
Total amount for user 42 = 91.08188790925797
● rac@PTPMRT07 10-11-2025 % scala CombineFuturesWithDependenciesMapNStyle.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-client.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Sum = 60, Average = 20
Sum = 60, Average = 20
◆ rac@PTPMRT07 10-11-2025 %

```

Q5. Coordinated Retry — Asynchronous Resilience

Code:

```

import scala.concurrent.{Future, ExecutionContext}
import scala.concurrent.ExecutionContext.Implicits.global
import scala.util.Random

object CoordinatedRetryAsynchronousResilience extends App {

    def fetchDataFromServer(
        server: String
    )(implicit ec: ExecutionContext): Future[String] = Future {
        if (Random.nextBoolean())
            throw new RuntimeException(s"Failed to fetch from $server")
        else s>Data from $server"
    }

    def fetchWithRetry(server: String, maxRetries: Int)(implicit
        ec: ExecutionContext
    ): Future[String] = {
        fetchDataFromServer(server).recoverWith {
            case _ if maxRetries > 0 =>
                println(s"Retrying... remaining attempts: $maxRetries")
                fetchWithRetry(server, maxRetries - 1)
        }
    }
}

fetchWithRetry("Server-1", 3).onComplete(println)

}

```

Output:

```

● rac@PTPMRT07 10-11-2025 % scala CombineFuturesWithDependenciesMapNStyle.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cli.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Compiling project (Scala 3.7.3, JVM (25))
Compiled project (Scala 3.7.3, JVM (25))
Sum = 60, Average = 20
Sum = 60, Average = 20
● rac@PTPMRT07 10-11-2025 % scala CoordinatedRetryAsynchronousResilience.scala
WARNING: A terminally deprecated method in sun.misc.Unsafe has been called
WARNING: sun.misc.Unsafe::objectFieldOffset has been called by scala.runtime.LazyVals$ (file:/opt/homebrew/Cellar/scala/3.7.3/libexec/libexec/scala-cli.jar)
WARNING: Please consider reporting this to the maintainers of class scala.runtime.LazyVals$
WARNING: sun.misc.Unsafe::objectFieldOffset will be removed in a future release
Success(Data from Server-1)
◆ rac@PTPMRT07 10-11-2025 %

```