

Bowen Jiang (NUID: 001582174)

CSYE 7200

Big-Data Sys Engr Using Scala

Sprint 2021

Assignment 3

## Task:

As mentioned in class there are three implementations to create in *Movie.scala* (lines 101, 124, 204). This time it is OK to edit the Spec file (but after this assignment, you should not edit any Spec files). But I want to give you some familiarity with Spec files (i.e. ScalaTest with Matchers). The most useful documentation for this sort of thing is here: [http://www.scalatest.org/user\\_guide/using\\_matchers](http://www.scalatest.org/user_guide/using_matchers).

The purpose of this code is to read the movie database file into *Movie* instances.

The module name for this assignment is *assignment-movie-database*. Let me clarify a little: you only need to work with *Ingest.scala*, *Movie.scala* and *IngestSpec.scala*, in addition, *MovieSpec.scala* is available for you to verify your implementation. These are all in the *asstmd* package (the spec file is under test of course). You will also need *movie\_metadata.csv* which is under test/resources of the project (use this version).

## Function Implement

The function I implement:

Movie.scala

```
/**
 * Method to yield a Try[Movie] from a String representing a line of input of the movie database file.
 *
 * TODO 11 points.
 *
 * @param w a line of input.
 * @return a Try[Movie]
 */
def parse(w: String): Try[Movie] = Try(
  Movie(w.split(",").toSeq)) // TO BE IMPLEMENTED
}
```

```

*/
def elements(list: Seq[String], indices: Int*): List[String] = {
  // Hint: form a new list which is consisted by the elements in list in position indices. Int* means array of Int.
  // 6 points
  val result: Seq[String] =
    // TO BE IMPLEMENTED
    for(index <- indices) yield list(index)
  result.toList
}

```

```

def apply(s: String): Rating =
  s match {
    case rRating(code, _, null) => Rating(code, None)
    case rRating(code, _, age) => Rating(code, Some(age.toInt))
    case _ => throw ParseException(s"parse error in Rating: $s")
  }

```

IngestSpec.scala

```

val xys = ingester(source).toSeq
// check that xys has exactly one element, consisting of Success(42) -- 10 points
// TO BE IMPLEMENTED
xys.length shouldBe 1
xys(0) shouldBe Success(42)

```

## Unit Test Screenshot

The screenshot displays the IntelliJ IDEA IDE interface. The top toolbar includes menus like File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, Git, Window, SBT Commands, and Help. The title bar shows the project name 'assignment-movie-database' and the file 'IngestSpec.scala'.

The left sidebar shows the Project view with the following structure:

- assignment-movie-database
  - idea
  - out
  - project
  - src
    - main.scala.edu.neu.coe.csye7200.asstmd
      - Ingest
      - Movie
    - test
      - resources
      - scala.edu.neu.coe.csye7200.asstmd
        - IngestSpec
        - MovieSpec
    - target
      - bg-jobs
      - global-logging
      - scala-2.13
      - streams
      - task-temp-directory
      - test-reports
      - Assignment3 Report.docx
      - assignment-movie-database.iml
      - build.sbt
      - UnitTest.png
    - External Libraries
    - Scratches and Consoles

The main editor shows the code for `IngestSpec.scala`:

```
1 import ...
2
3 /**
4  * Created by scalaprof on 9/13/16.
5  */
6
7 class IngestSpec extends AnyFlatSpec with Matchers {
8
9   behavior of "ingest"
10
11   it should "work for Int" in {
12     trait ParsableInt$ extends Parsable[Int] {
13       def parse(w: String): Try[Int] = Try(w.toInt)
14     }
15     implicit object ParsableInt$ extends ParsableInt$
16     val source = Source.fromChars(Array('x', '\n', '4', '2'))
17     val ingester = new Ingest[Int]()
18     val xys = ingester(source).toSeq
19     // check that xys has exactly one element, consisting of Success(42) -- 10 points
20     // TO BE IMPLEMENTED
21     xys.length shouldBe 1
22     xys(0) shouldBe Success(42)
23   }
24
25   it should "work for movie database" in {
26     implicit val codec: Codec = Codec("UTF-8")
27     // NOTE that you expect to see a number of exceptions thrown. That's OK. We expect that some lines will not parse correctly.
28   }
29 }
```

The bottom panel shows the Terminal output:

```
edu.neu.coe.csye7200.asstmd.ParseException: parse error in Name: Ana de la Reguera (parsed as None)
[info] IngestSpec:
[info] ingest
[info] - should work for Int
[info] - should work for movie database
[info] Run completed in 616 milliseconds.
[info] Total number of tests run: 18
[info] Suites: completed 2, aborted 0
[info] Tests: succeeded 18, failed 0, canceled 0, ignored 0, pending 0
[info] All tests passed.
[success] Total time: 11 s, completed 2021年2月16日 5:57:27
[info] running edu.neu.coe.csye7200.asstmd.Movie
[success] Total time: 1 s, completed 2021年2月16日 5:57:28
```

The bottom status bar shows 'IntelliJ IDEA 2020.3.2 available // Update... (a minute ago)' and the system clock '17:57 2021/2/16'.