Qinyun Lin – SEC01 (NUID 001582464)

Big Data System Engineering with Scala
Spring 2022
Assignment No. 5

Task

1. **Implement 13 methods in Function.scala**
2. **Implement 2 methods in Movie.scala**

Solution

1. Implement 13 methods in Function.scala

```scala
def map2[T1, T2, R](t1y: Try[T1], t2y: Try[T2])(f: (T1, T2) => R): Try[R] =
  for(t1 <- t1y; t2 <- t2y) yield f(t1, t2)
```

```scala
def map3[T1, T2, T3, R](t1y: Try[T1], t2y: Try[T2], t3y: Try[T3])(f: (T1, T2, T3) => R): Try[R] =
  for(t1 <- t1y; t2 <- t2y; t3 <- t3y) yield f(t1, t2, t3)
```

```scala
def map7[T1, T2, T3, T4, T5, T6, T7, R](t1y: Try[T1], t2y: Try[T2], t3y: Try[T3], t4y: Try[T4],
                                        t5y: Try[T5], t6y: Try[T6], t7y: Try[T7])
                                       (f: (T1, T2, T3, T4, T5, T6, T7) => R): Try[R] =
  for(t1 <- t1y; t2 <- t2y; t3 <- t3y; t4 <- t4y; t5 <- t5y; t6 <- t6y ; t7 <- t7y)
    yield f(t1, t2, t3, t4, t5, t6, t7)
```

```scala
def lift[T, R](f: T => R): Try[T] => Try[R] = _ map f
```

```scala
def lift2[T1, T2, R](f: (T1, T2) => R): (Try[T1], Try[T2]) => Try[R] = map2(_, _)(f)
```

```scala
def lift3[T1, T2, T3, R](f: (T1, T2, T3) => R): (Try[T1], Try[T2], Try[T3]) => Try[R] =
  map3(_, _, _)(f)
```

```scala
def lift7[T1, T2, T3, T4, T5, T6, T7, R](f: (T1, T2, T3, T4, T5, T6, T7) => R):
(Try[T1], Try[T2], Try[T3], Try[T4], Try[T5], Try[T6], Try[T7]) => Try[R] =
  map7(_, _, _, _, _, _, _)(f)
```

```scala
def invert2[T1, T2, R](f: T1 => T2 => R): T2 => T1 => R = t2 => t1 => f(t1)(t2)
```

```scala
def invert3[T1, T2, T3, R](f: T1 => T2 => T3 => R): T3 => T2 => T1 => R =
  t3 => t2 => t1 => f(t1)(t2)(t3)
```

```scala
def invert4[T1, T2, T3, T4, R](f: T1 => T2 => T3 => T4 => R): T4 => T3 => T2 => T1 => R =
  t4 => t3 => t2 => t1 => f(t1)(t2)(t3)(t4)
```

```scala
def uncurried2[T1, T2, T3, R](f: T1 => T2 => T3 => R): (T1, T2) => T3 => R =
  (t1, t2) => t3 => f(t1)(t2)(t3)
```

```scala
def uncurried3[T1, T2, T3, T4, R](f: T1 => T2 => T3 => T4 => R): (T1, T2, T3) => T4 => R =
  (t1, t2, t3) => t4 => f(t1)(t2)(t3)(t4)
```

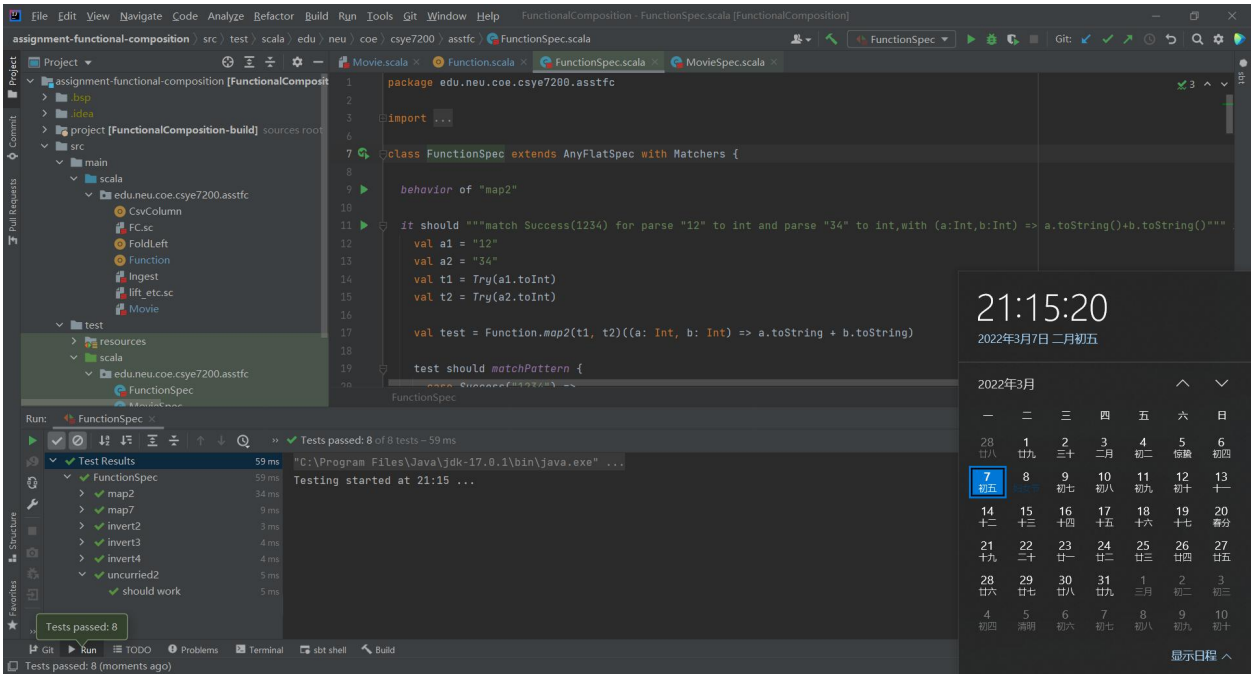
```scala
def uncurried7[T1, T2, T3, T4, T5, T6, T7, T8, R](f: T1 => T2 => T3 => T4 => T5 => T6 => T7 => T8 => R):
(T1, T2, T3, T4, T5, T6, T7) => T8 => R =
(t1, t2, t3, t4, t5, t6, t7) => t8 => f(t1)(t2)(t3)(t4)(t5)(t6)(t7)(t8)
```

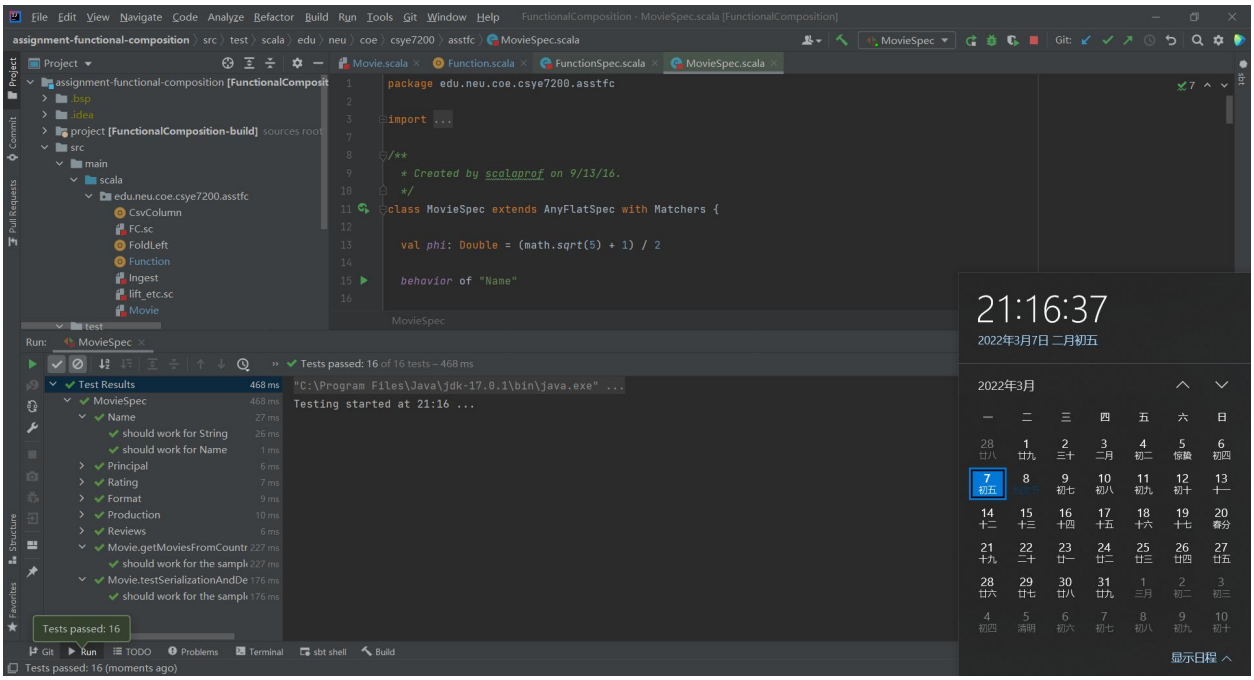
2. Implement 2 methods in Movie.scala

```scala
object MoviesProtocol extends DefaultJsonProtocol {
  // 20 points
  // TO BE IMPLEMENTED
  implicit val formatFormat: RootJsonFormat[Format] = jsonFormat4(Format.apply)
  implicit val productionFormat: RootJsonFormat[Production] = jsonFormat4(Production.apply)
  implicit val ratingFormat: RootJsonFormat[Rating] = jsonFormat2(Rating.apply)
  implicit val reviewsFormat: RootJsonFormat[Reviews] = jsonFormat7(Reviews.apply)
  implicit val nameFormat: RootJsonFormat[Name] = jsonFormat4(Name.apply)
  implicit val principalFormat: RootJsonFormat[Principal] = jsonFormat2(Principal.apply)
  implicit val movieFormat: RootJsonFormat[Movie] = jsonFormat11(Movie.apply)
}
```

```scala
def testSerializationAndDeserialization(ms: Seq[Movie]): Boolean = {
  // 5 points
  // TO BE IMPLEMENTED
  import MoviesProtocol._
  val SerializationJson: Seq[JsValue] = ms.map(m => m.toJson)
  val DeserializationJson = SerializationJson.map(js => js.convertTo[Movie])
  ms.equals(DeserializationJson)
}
```

Unit Test Screenshot





Project Source

https://github.com/MrNiro/CSYE7200/tree/Spring2022/assignment-functional-composition/src/main/scala/edu/neu/coe/csye7200/asstfc