Hrishikesh Sanjay Pawar – SEC01 (NUID 002707307)

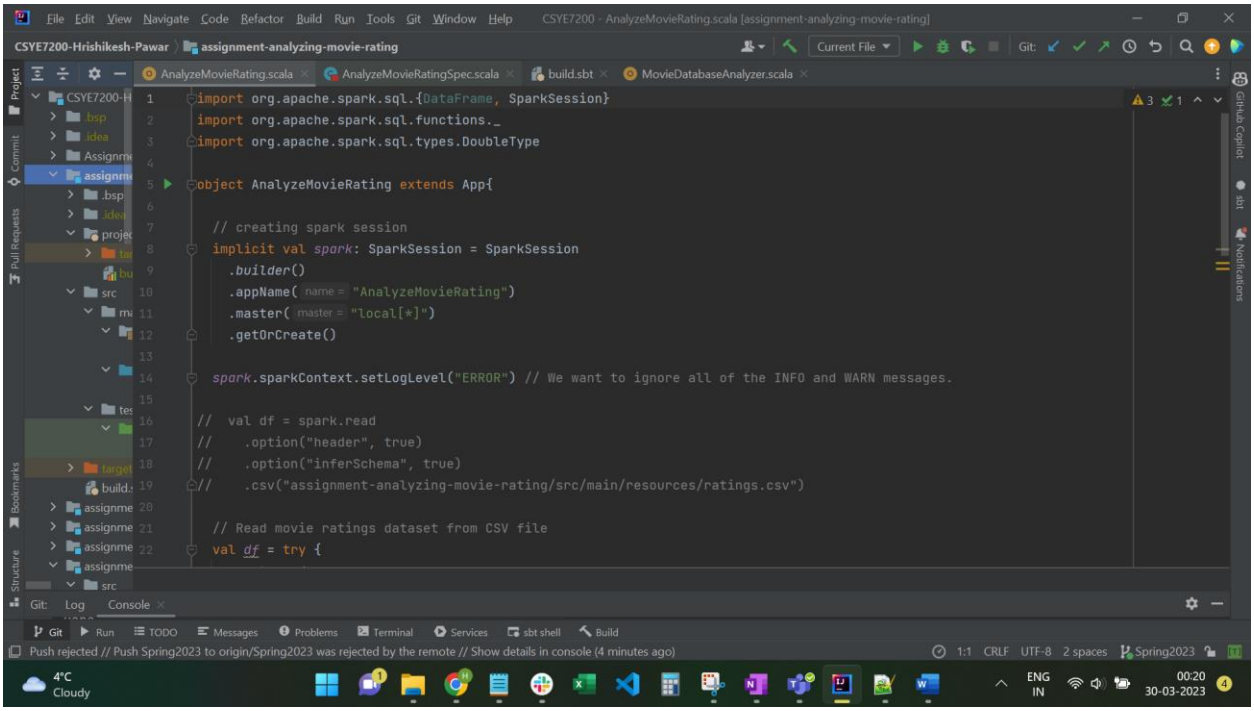
Big Data System Engineering with Scala
Spring 2023
Assignment No. 7

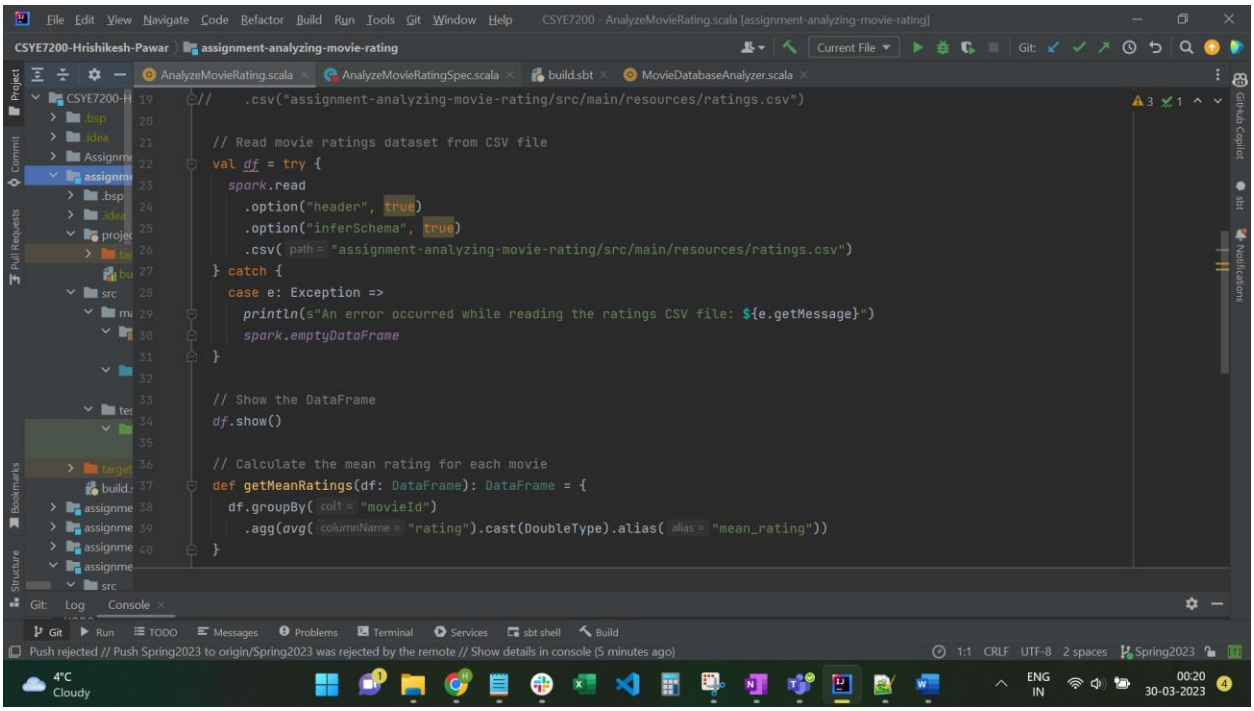
-List of Tasks Implemented

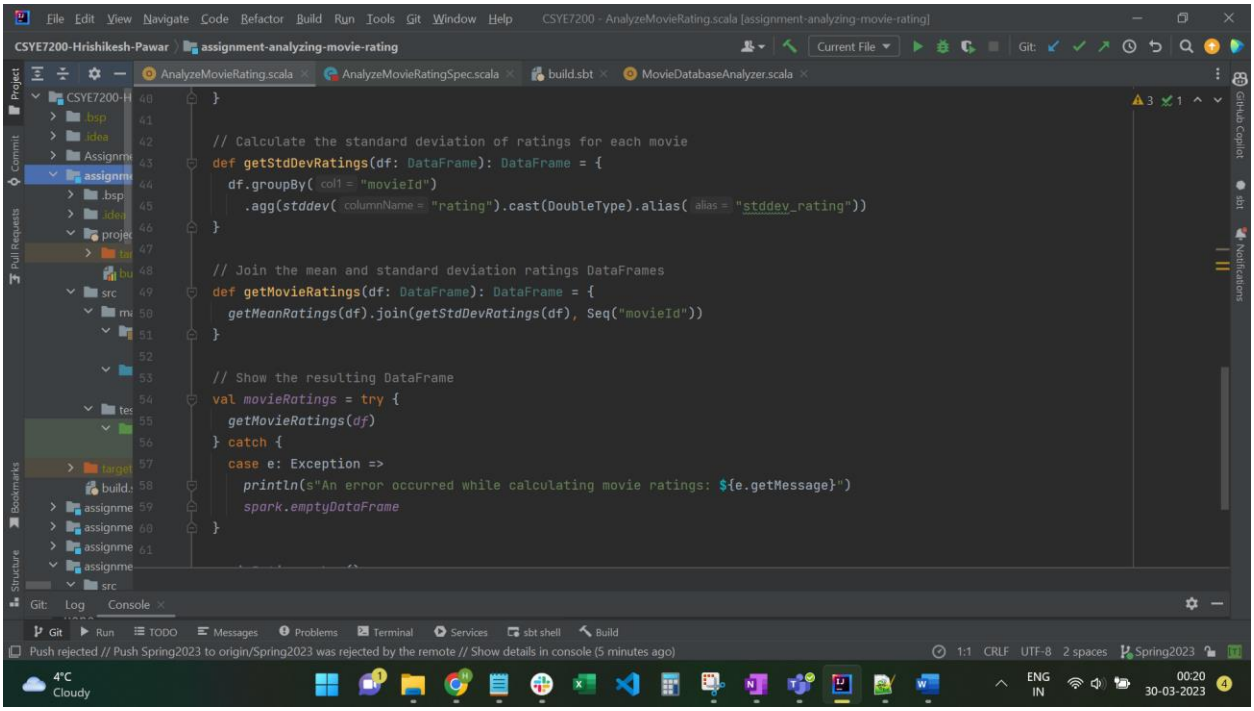
1. Read the movie rating csv file into spark dataframe
2. Show the dataframe records on console
2. Calculate mean of movie ratings for each movie
3. Calculate standard deviation for each movie
4. Join the two results to form a final dataframe with both results
5. Add exception handling
6. Write test cases in Spec file

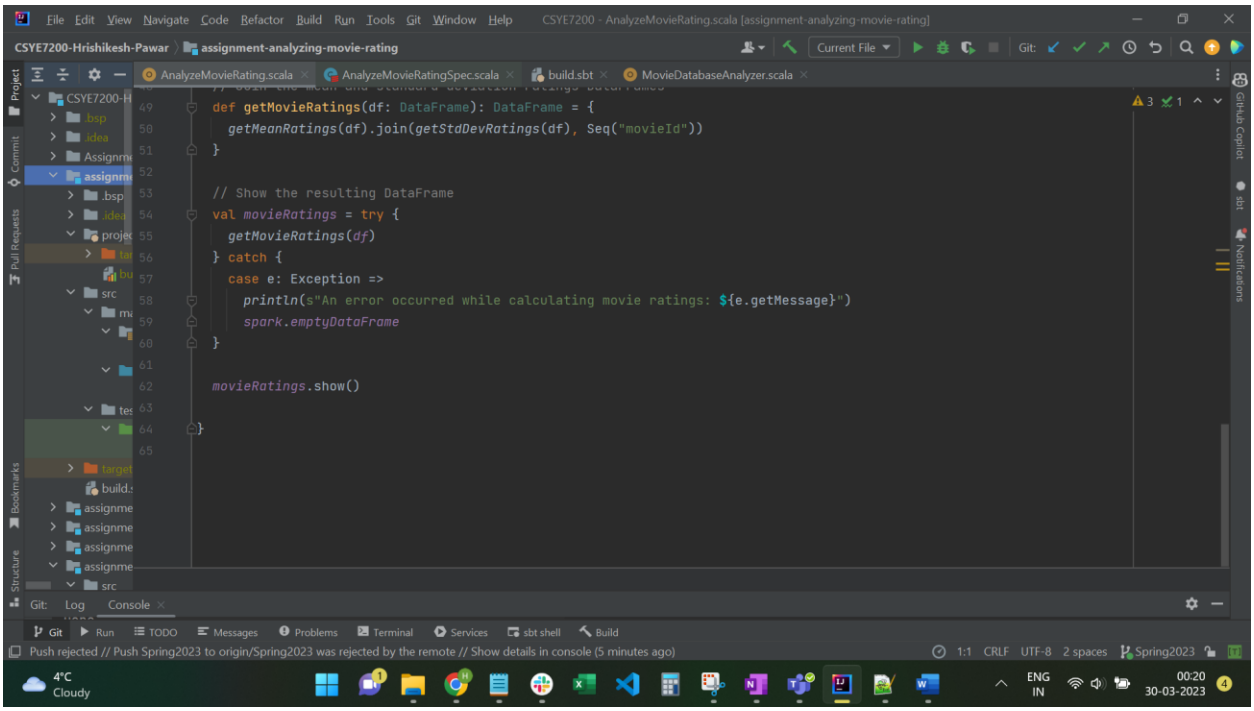
-Code

AnalyzeMovieRating.scala

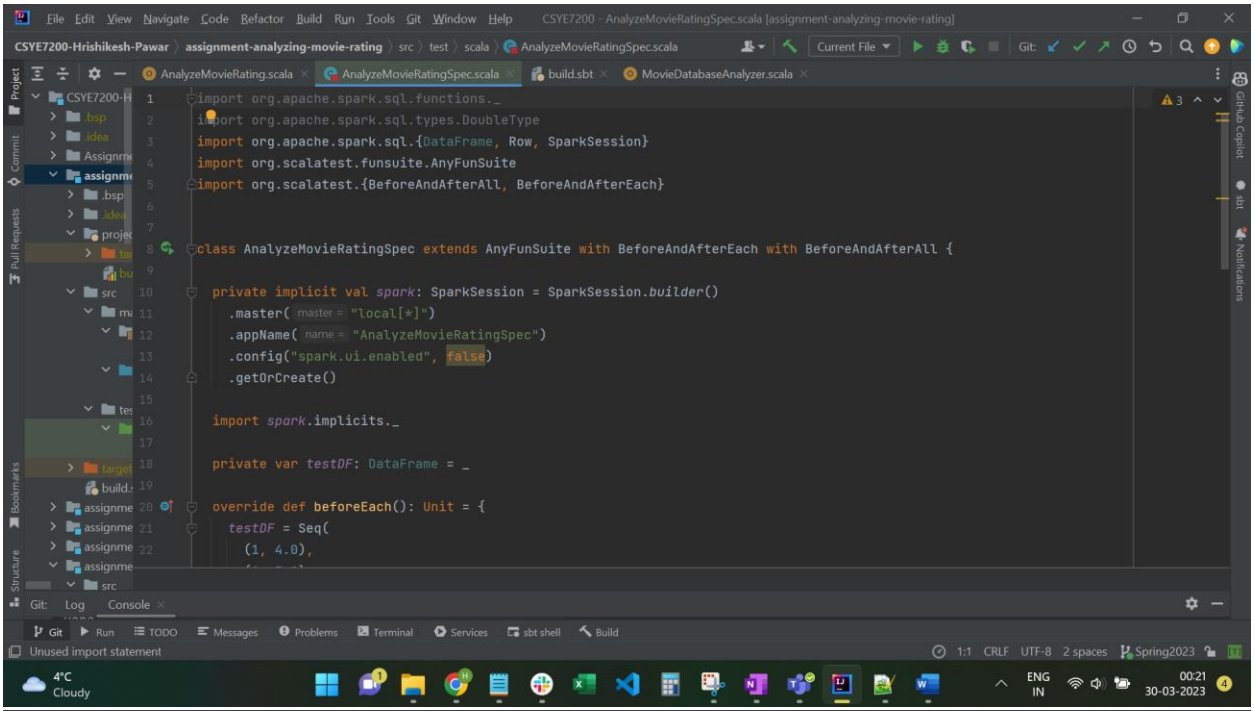


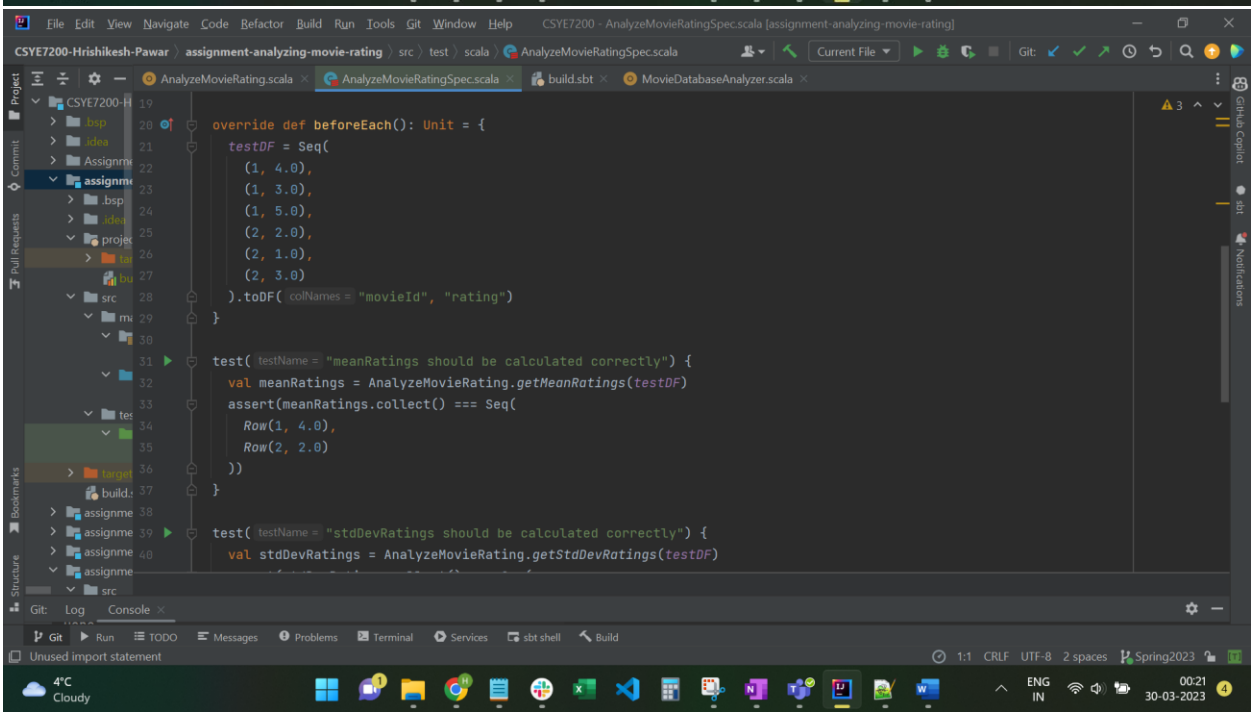

File Edit View Navigate Code Refactor Build Run Tools Git Window Help
CSYE7200 - AnalyzeMovieRating.scala [assignment-analyzing-movie-rating]
CSYE7200-Hrishikesh-Pawar
assignment-analyzing-movie-rating
Current File
Git:
AnalyzeMovieRating.scala
AnalyzeMovieRatingSpec.scala
build.sbt
MovieDatabaseAnalyzer.scala
//   .csv("assignment-analyzing-movie-rating/src/main/resources/ratings.csv")
// Read movie ratings dataset from CSV file
val df = try {
spark.read
.option("header", true)
.option("inferSchema", true)
.csv(path = "assignment-analyzing-movie-rating/src/main/resources/ratings.csv")
} catch {
case e: Exception =>
println(s"An error occurred while reading the ratings CSV file: ${e.getMessage}")
spark.emptyDataFrame
}
// Show the DataFrame
df.show()
// Calculate the mean rating for each movie
def getMeanRatings(df: DataFrame): DataFrame = {
df.groupBy(col1 = "movieId")
.agg(avg(columnName = "rating").cast(DoubleType).alias(alias = "mean_rating"))
}
Git: Log Console
Git Run TODO Messages Problems Terminal Services sbt shell Build
Push rejected // Push Spring2023 to origin/Spring2023 was rejected by the remote // Show details in console (5 minutes ago)
1:1 CRLF UTF-8 2 spaces Spring2023
4°C Cloudy
ENG IN
00:20 30-03-2023

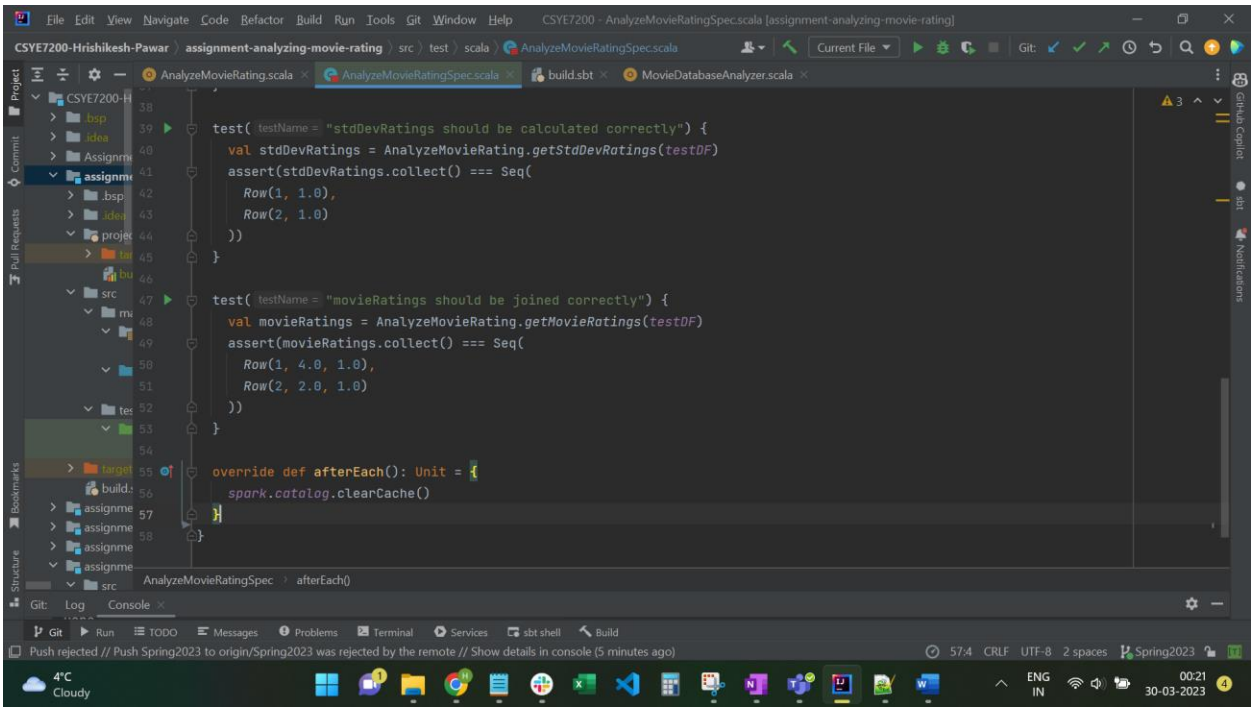

File Edit View Navigate Code Refactor Build Run Tools Git Window Help
CSYE7200 - AnalyzeMovieRating.scala [assignment-analyzing-movie-rating]
CSYE7200-Hrishikesh-Pawar
assignment-analyzing-movie-rating
Current File
Git:
AnalyzeMovieRating.scala
AnalyzeMovieRatingSpec.scala
build.sbt
MovieDatabaseAnalyzer.scala
}
// Calculate the standard deviation of ratings for each movie
def getStdDevRatings(df: DataFrame): DataFrame = {
df.groupBy(col1 = "movieId")
.agg(stddev(columnName = "rating").cast(DoubleType).alias(alias = "stddev_rating"))
}
// Join the mean and standard deviation ratings DataFrames
def getMovieRatings(df: DataFrame): DataFrame = {
getMeanRatings(df).join(getStdDevRatings(df), Seq("movieId"))
}
// Show the resulting DataFrame
val movieRatings = try {
getMovieRatings(df)
} catch {
case e: Exception =>
println(s"An error occurred while calculating movie ratings: ${e.getMessage}")
spark.emptyDataFrame
}
Git: Log Console
Git Run TODO Messages Problems Terminal Services sbt shell Build
Push rejected // Push Spring2023 to origin/Spring2023 was rejected by the remote // Show details in console (5 minutes ago)
1:1 CRLF UTF-8 2 spaces Spring2023
4°C Cloudy
ENG IN
00:20 30-03-2023


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
CSYE7200 - AnalyzeMovieRating.scala [assignment-analyzing-movie-rating]
CSYE7200-Hrishikesh-Pawar
assignment-analyzing-movie-rating
Current File
Git:
AnalyzeMovieRating.scala
AnalyzeMovieRatingSpec.scala
build.sbt
MovieDatabaseAnalyzer.scala
def getMovieRatings(df: DataFrame): DataFrame = {
getMeanRatings(df).join(getStdDevRatings(df), Seq("movieId"))
}
// Show the resulting DataFrame
val movieRatings = try {
getMovieRatings(df)
} catch {
case e: Exception =>
println(s"An error occurred while calculating movie ratings: ${e.getMessage}")
spark.emptyDataFrame
}
movieRatings.show()
}
Git: Log Console
Git Run TODO Messages Problems Terminal Services sbt shell Build
Push rejected // Push Spring2023 to origin/Spring2023 was rejected by the remote // Show details in console (5 minutes ago)
1:1 CRLF UTF-8 2 spaces Spring2023
4°C
Cloudy
ENG
IN
00:20
30-03-2023

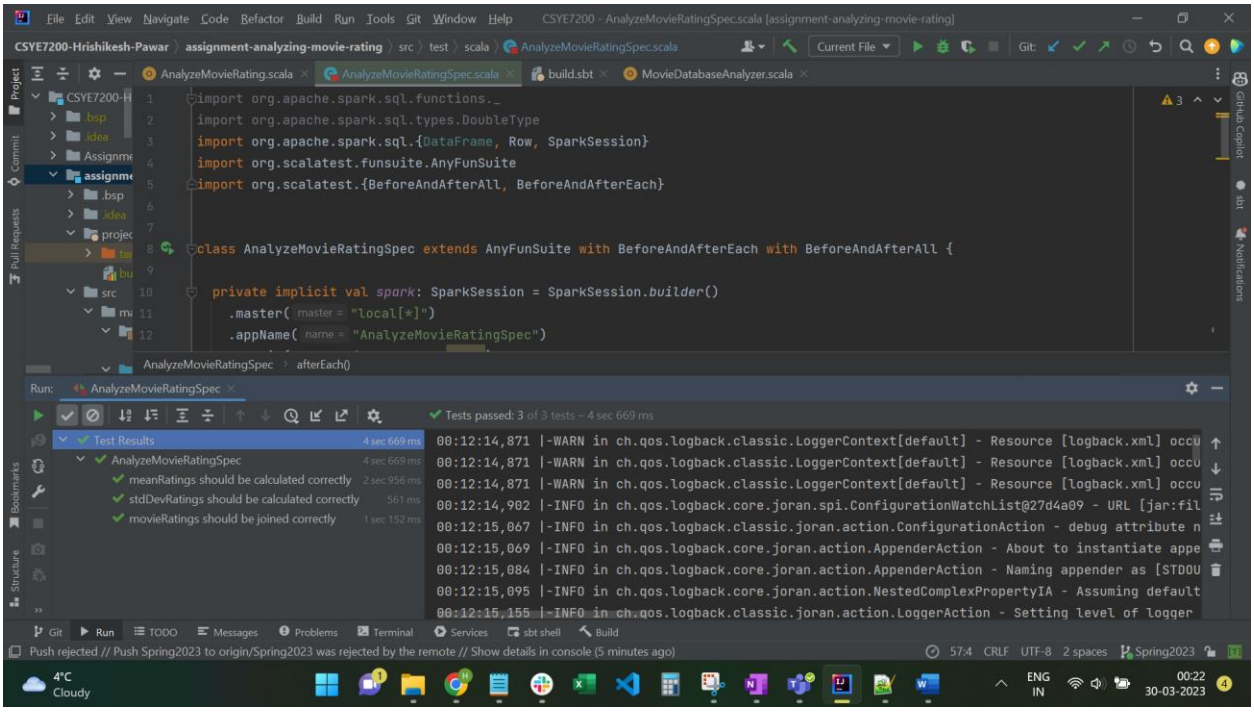
AnalyzeMovieRatingSpec.scala







-Unit tests



- **GitHub Repo**

https://github.com/hrishikesh-pawar/CSYE7200-Hrishikesh-Pawar

- **Dataset**

https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset

ratings.csv file from the above link

GitHub does not allow a file over 100 MB so the push was getting rejeceted. For this reason, I have not pushed the ratings.csv file on GitHub. But to run the project, you can paste the ratings.csv file in this path:

assignment-analyzing-movie-rating/src/main/resources/ratings.csv