```
001,56,CSE,aditi
002,67,DSE,raju
003,78,CCE,abhi
004,89,DSE,varalika
005,40,CSE,kanha
006,51,CCE,ishan
007,59,DSE,rakshit
008,63,CCE,naina
```

Code:

```scala
// Define a case class for Student
case class Student(id: String, marks: Int, department: String, name: String)

// Load the input data from input.txt
val inputData = spark.read.textFile("input.txt").rdd.collect()

// Convert each line into a Student object
val students = inputData.map(line => {
val parts = line.split(",")
Student(parts(0), parts(1).toInt, parts(2), parts(3))
})

// Convert RDD to DataFrame
val studentDF = spark.createDataFrame(students)

// Convert DataFrame to CSV and display all rows
studentDF.write
.option("header", "true")
.csv("output.csv")

// Display all students who got marks greater than 70
val highScorers = studentDF.filter($"marks" > 70)
highScorers.show()
```

CODE :

```scala
import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.functions._

// Create SparkSession
val spark = SparkSession.builder()
.appName("StudentMarksFilter")
.master("local[*]")
.getOrCreate()
```

```scala
// Create DataFrame from input data
val data = Seq(
("001", 56, "CSE", "aditi"),
("002", 67, "DSE", "raju"),
("003", 78, "CCE", "abhi"),
("004", 89, "DSE", "varalika"),
("005", 40, "CSE", "kanha"),
("006", 51, "CCE", "ishan"),
("007", 59, "DSE", "rakshit"),
("008", 63, "CCE", "naina")
)

val df = spark.createDataFrame(data).toDF("id", "marks", "department", "name")

// Write DataFrame to CSV
df.write
.mode("overwrite") // Overwrite existing file if exists
.option("header", "true")
.csv("output.csv")

// Display all students who got marks greater than 70
val highScorers = df.filter(col("marks") > 70)
highScorers.show()
```