you can leverage `lambda`, `Stream`, and `Predicate` to perform sorting operations with custom comparators in a more concise and functional style

A Student class is given with the following properties with appropriate getters, setters and toString overridden method
rollno - int
name - String
address - String
rank - int

Now the list of menu options is available for access. You need to write the following custom comparators
sortByName
sortByRollno
sortByRank

For the sortByRank comparator, implement the option for choosing the ascending and descending order for the rank.
sortByRank (boolean);
the Boolean value denotes whether the comparator should sort the options in ascending or descending order

The user should provide either ASC / DES (or) asc / des as the option for the sortByName comparator

Note: Write the logic only for the comparators, all the logic is already written.

Input format :

Enter the details based on the choice

1. Add User
2. Sort Student List by Roll no
3. Sort Student List by Name
4. Sort Students by Rank
5. Exit from System

Output format :

Refer to the sample output for reference.

Sample test cases :

Input 1 :

```
1
1,Gokul,Address1,2
1
2,Arima,Address2,5
1
3,Rhenal,Address3,1
2
3
4
asc
4
dsc
6
```

```
5
```

Output 1 :

```
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Enter the rollno, name, address and rank (separated by comma)
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Enter the rollno, name, address and rank (separated by comma)
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Enter the rollno, name, address and rank (separated by comma)
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :

Students List sorted by rollno
1 Gokul Address1 2
2 Arima Address2 5
3 Rhenal Address3 1
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :

Students List sorted by rollno
2 Arima Address2 5
1 Gokul Address1 2
3 Rhenal Address3 1
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
```

```
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Sort by ascending or descending ( asc / des)

Students List sorted by Rank
3 Rhenal Address3 1
1 Gokul Address1 2
2 Arima Address2 5
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Sort by ascending or descending ( asc / des)

Students List sorted by Rank
2 Arima Address2 5
1 Gokul Address1 2
3 Rhenal Address3 1
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :

Invalid Input Try again !!!

Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Exiting ....
```

Input 2 :

```
1
1,Gokul,Address1,4
6
5
```

Output 2 :

```
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Enter the rollno, name, address and rank (separated by comma)
```

```
Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :

Invalid Input Try again !!!

Student Interactive Console :
1) Add User
2) Sort Student List by Roll no
3) Sort Student List by Name
4) Sort Students by Rank
5) Exit from System

Enter your choice :
Exiting ....
```

Note :

The program will be evaluated only after the “Submit Code” is clicked.

Extra spaces and new line characters in the program output will result in the failure of the test case.