Assessment Instructions

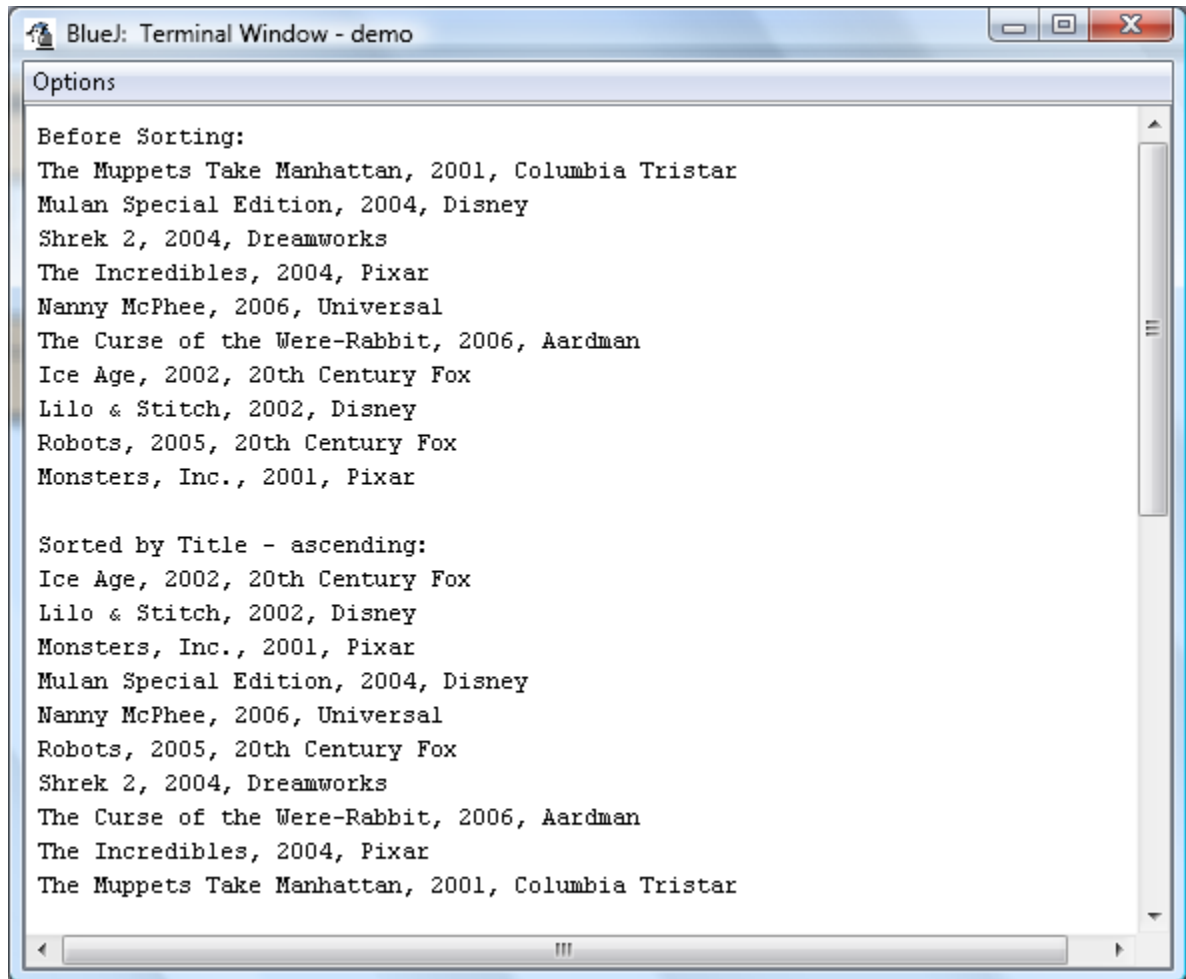**Instructions**: For this assessment, you are going to create some sorting methods that use Insertion sort.

1. Create a folder called **Assessment** in your Unit 7 assessments folder.
   a. Create a class called **Movie2**.
   b. **Movie2** will need instance variables **title, year,** and **studio** (of types **String**, **int**, and **String**, respectively).
   c. **Movie2** will need appropriate methods and constructors. Make sure to have a **toString()** method that prints the movie in the following format:
      Title, Year, Studio
   d. Save the class as **Movie2.java**.
2. You are to create a class called T**testMovie2** and save it as **TestMovie2.java**.
   a. Make sure that you create an array called **myMovies**. Add the following movies.

Title	Year	Studio
The Muppets Take Manhattan	2001	Columbia Tristar
Mulan Special Edition	2004	Disney
Shrek 2	2004	Dreamworks
The Incredibles	2004	Pixar
Nanny McPhee	2006	Universal
The Curse of the Were-Rabbit	2006	Aardman
Ice Age	2002	20th Century Fox
Lilo & Stitch	2002	Disney
Robots	2005	20th Century Fox
Monsters Inc.	2001	Pixar

   b. Create a method called **printMovies**() that traverses through the array and prints out each element.
   c. Create a method called **sortTitles()** that insertion sorts the array according to **title**. It should have two arguments: the array and an int parameter that if it is 1 means ascending, and 2 means descending. Your method should sort appropriately depending on the value of the second parameter (ascending or descending). It should return the sorted array.
   d. Create a method called **sortYears()** that insertion sorts the array according to **year**. It should have two arguments: the array and an int parameter that if it is 1 means ascending, and 2 means descending. Your method should sort appropriately depending on the value of the second parameter (ascending or descending). It should return the sorted array.
   e. Create a method called **sortStudios()** that insertion sorts the array according to **studio**. It should have two arguments: the array and an int parameter that if it is 1 means ascending, and 2 means descending. Your method should sort

appropriately depending on the value of the second parameter (ascending or descending). It should return the sorted array.

f. Test your methods, showing array without sort first, then sort by title ascending and show output, then sort by year descending and show output, and finally sort by studio ascending and show output. Your output will be similar to output shown below:

```
BlueJ: Terminal Window - demo
Options

Before Sorting:
The Muppets Take Manhattan, 2001, Columbia Tristar
Mulan Special Edition, 2004, Disney
Shrek 2, 2004, Dreamworks
The Incredibles, 2004, Pixar
Nanny McPhee, 2006, Universal
The Curse of the Were-Rabbit, 2006, Aardman
Ice Age, 2002, 20th Century Fox
Lilo & Stitch, 2002, Disney
Robots, 2005, 20th Century Fox
Monsters, Inc., 2001, Pixar

Sorted by Title - ascending:
Ice Age, 2002, 20th Century Fox
Lilo & Stitch, 2002, Disney
Monsters, Inc., 2001, Pixar
Mulan Special Edition, 2004, Disney
Nanny McPhee, 2006, Universal
Robots, 2005, 20th Century Fox
Shrek 2, 2004, Dreamworks
The Curse of the Were-Rabbit, 2006, Aardman
The Incredibles, 2004, Pixar
The Muppets Take Manhattan, 2001, Columbia Tristar
```

```
BlueJ:  Terminal Window - demo

Options

Sorted by Year - descending:
The Curse of the Were-Rabbit, 2006, Aardman
Nanny McPhee, 2006, Universal
Robots, 2005, 20th Century Fox
The Incredibles, 2004, Pixar
Shrek 2, 2004, Dreamworks
Mulan Special Edition, 2004, Disney
Lilo & Stitch, 2002, Disney
Ice Age, 2002, 20th Century Fox
The Muppets Take Manhattan, 2001, Columbia Tristar
Monsters, Inc., 2001, Pixar

Sorted by Studio - ascending:
Ice Age, 2002, 20th Century Fox
Robots, 2005, 20th Century Fox
The Curse of the Were-Rabbit, 2006, Aardman
The Muppets Take Manhattan, 2001, Columbia Tristar
Lilo & Stitch, 2002, Disney
Mulan Special Edition, 2004, Disney
Shrek 2, 2004, Dreamworks
Monsters, Inc., 2001, Pixar
The Incredibles, 2004, Pixar
Nanny McPhee, 2006, Universal
```