ABSOLUTE NUMBER SORTING

In this coding exercise you will be expected to develop a method that sorts an array of numbers according to their **absolute** values in descending order. Your method ***void sortAbsolute(Number[] numbers)*** should call Arrays.sort() internally. Your method should sort all types of numbers.

1. The main() method and the incomplete ***void sortAbsolute(Number[] numbers)*** method are as below

```
public class Main {

   public static void main(String[] args) {
      Integer[] integers={-1,2,-3,4,6,5,9,-7,8,-10};
      Float[] floats={-1.5f,2.5f,-3.5f,4.5f,6.5f,5.5f,9.5f,-7.5f,8.5f,-10.5f};
      Double [] doubles={-1.5,2.5,-3.5,4.5,6.5,5.5,9.5,-7.5,8.5,-10.5};
      Short[] shorts={-1,2,-3,4,6,5,9,-7,8,-10};
      Byte[] bytes={-1,2,-3,4,6,5,9,-7,8,-10};
       sortAbsolute(integers);
       sortAbsolute(floats);
       sortAbsolute(doubles);
       sortAbsolute(shorts);
       sortAbsolute(bytes);
       System.out.println("integers: "+Arrays.toString(integers));
       System.out.println("floats: "+Arrays.toString(floats));
       System.out.println("doubles: "+Arrays.toString(doubles));
       System.out.println("shorts: "+Arrays.toString(shorts));
       System.out.println("bytes: "+Arrays.toString(bytes));
   }

   public static void sortAbsolute(Number[] numbers) {
      //Please use Arrays.sort()

   }
}
```

2. Output of the above main() method is as below

```
integers: [-10, 9, 8, -7, 6, 5, 4, -3, 2, -1]
floats: [-10.5, 9.5, 8.5, -7.5, 6.5, 5.5, 4.5, -3.5, 2.5, -1.5]
doubles: [-10.5, 9.5, 8.5, -7.5, 6.5, 5.5, 4.5, -3.5, 2.5, -1.5]
shorts: [-10, 9, 8, -7, 6, 5, 4, -3, 2, -1]
bytes: [-10, 9, 8, -7, 6, 5, 4, -3, 2, -1]
```