

Sorting algorithms:

<http://maven.smith.edu/~thiebaut/java/sort/>

http://en.wikipedia.org/wiki/Sorting_algorithm#Selection_sort

<http://www.javaworld.com/javaworld/jw-12-2002/jw-1227-sort.html>

<http://www.lepoint.net/notes-java/index.html>

```

// demonstrates sorting objects (uses insertion sort)
class Person
{
    private String lastName;
    private String firstName;
    private int age;
    //-----
    public Person(String last, String first, int a)
    {
        // constructor
        lastName = last;
        firstName = first;
        age = a;
    }
    //-----
    public void displayPerson()
    {
        System.out.print("  Last name: " + lastName);
        System.out.print(", First name: " + firstName);
        System.out.println(", Age: " + age);
    }
    //-----
    public String getLast()      // get last name
    { return lastName; }
} // end class Person
////////////////////////////////////

class ArrayInOb
{
    private Person[] a;      // ref to array a
    private int nElems;     // number of data items
    //-----
    public ArrayInOb(int max) // constructor
    {
        a = new Person[max]; // create the array
        nElems = 0;          // no items yet
    }
    //-----
    // put person into array
    public void insert(String last, String first, int age)
    {
        a[nElems] = new Person(last, first, age);
    }
}

```

```

    nElems++;          // increment size
}
//-----
public void display()    // displays array contents
{
    for(int j=0; j<nElems; j++)    // for each element,
        a[j].displayPerson();    // display it
    System.out.println("");
}
//-----
public void insertionSort()
{
    int in, out;
    for(out=1; out<nElems; out++) // out is dividing line
    {
        Person temp = a[out];    // remove marked person
        in = out;                // start shifting at out

        while(in>0 &&           // until smaller one found,
            a[in-1].getLast().compareTo(temp.getLast())>0)
        {
            a[in] = a[in-1];    // shift item to the right
            --in;                // go left one position
        }
        a[in] = temp;           // insert marked item
    } // end for
} // end insertionSort()
//-----
} // end class ArrayInOb
////////////////////////////////////
class ObjectSortApp
{
    public static void main(String[] args)
    {
        int maxSize = 100;      // array size
        ArrayInOb arr;          // reference to array
        arr = new ArrayInOb(maxSize); // create the array
        arr.insert("Evans", "Patty", 24);
        arr.insert("Smith", "Doc", 59);
        arr.insert("Smith", "Lorraine", 37);
    }
}

```

```
arr.insert("Smith", "Paul", 37);
arr.insert("Yee", "Tom", 43);
arr.insert("Hashimoto", "Sato", 21);
arr.insert("Stimson", "Henry", 29);
arr.insert("Velasquez", "Jose", 72);
arr.insert("Vang", "Minh", 22);
arr.insert("Creswell", "Lucinda", 18);

System.out.println("Before sorting:");
arr.display();           // display items
```

Here's the output of this program:

Before sorting:

```
Last name: Evans, First name: Patty, Age: 24
Last name: Smith, First name: Doc, Age: 59
Last name: Smith, First name: Lorraine, Age: 37
Last name: Smith, First name: Paul, Age: 37
Last name: Yee, First name: Tom, Age: 43
Last name: Hashimoto, First name: Sato, Age: 21
Last name: Stimson, First name: Henry, Age: 29
Last name: Velasquez, First name: Jose, Age: 72
Last name: Vang, First name: Minh, Age: 22
Last name: Creswell, First name: Lucinda, Age: 18
```

After sorting:

```
Last name: Creswell, First name: Lucinda, Age: 18
Last name: Evans, First name: Patty, Age: 24
Last name: Hashimoto, First name: Sato, Age: 21
Last name: Smith, First name: Doc, Age: 59
Last name: Smith, First name: Lorraine, Age: 37
Last name: Smith, First name: Paul, Age: 37
Last name: Stimson, First name: Henry, Age: 29
Last name: Vang, First name: Minh, Age: 22
Last name: Velasquez, First name: Jose, Age: 72
Last name: Yee, First name: Tom, Age: 43
```