/\*\*

\* Sergey Ganous

\* 012278381

\* 2/4/16

\* Lab 4 : Interface/Comparable

\*/

**public** **class** Person **implements** Measurable, Comparable {

**private** String lname;

**private** String fname;

**private** **double** age;

**public** Person(String inlname, String infname, **int** inage) {

lname=inlname;

fname=infname;

age=inage;

}

**public** **double** getMeasure() {

**return** age;

}

**public** String toString() {

**return** fname + "'" + lname + " is " + age + " years old.";

}

**public** **int** compareTo(Object someObject)

{

Person other = (Person)someObject;

**return** fname.compareTo(other.fname);

}

}

**public** **class** Data

{

/\*\*

Computes the average of the measures of the given objects.

**@param** objects an array of Measurable objects

**@return** the average of the measures

\*/

**public** **static** Measurable smaller(Measurable[] obj)

{

Measurable item = obj[0];

**for**(**int** i = 1;i<obj.length;i++){

**if**(item.getMeasure() > obj[i].getMeasure())

{

item = obj[i];

}

}

**return** item;

}

}

**import** java.lang.reflect.Array;

**import** java.util.Arrays;

**public** **class** PersonTester {

**public** **static** **void** main(String[] args){

Measurable[] people = **new** Measurable[3];

people[0] = **new** Person("Jones","Bill",25);

people[1] = **new** Person("Walker","Julie",20);

people[2] = **new** Person("Hopkins","Mary",15);

String youngest;

Measurable smallest = Data.*smaller*(people);

Person smallPerson = (Person)smallest;

youngest = smallPerson.toString();

System.***out***.println(youngest);

Arrays.*sort*(people);

}

}

**OUTPUT**

Mary'Hopkins is 15.0 years old.