

```
import java.util.Comparator;

public class SortLastNameDescending implements Comparator<Employee> {
    @Override
    public int compare(Employee a, Employee b){
        return b.last.compareTo(a.last);
    }
}
```

```
import java.util.*;
/**
 * This is the tester class that creates 9 instances of employees
 * -3 are staff
 * -3 are faculty
 * -3 are partime workers
 * in the tester we add the 9 instances employees into an arraylist and print all their toStrings and monthly earnings
 * @author Liam Abalos
 * @author Mark Fastner
 * CECS 277 Lab (Lab #3)
 */
import java.util.ArrayList;
```

```
public class Tester{
    public static void main(String[] args){
        double total_monthly_salary_partime = 0;
        double total_monthly_salary_employees = 0;
        //creates 9 objects
        Staff s1 = new Staff("Allen", "Paita", "123", 50.00);
        Staff s2 = new Staff("Zapata", "Steven", "456", 35.00);
        Staff s3 = new Staff("Rios", "Enrique", "789", 40.00);

        Education e1 = new Education("Ph.D", "Engineering", 3);
        Faculty f1 = new Faculty("Johnson", "Anne", "243", Faculty.Level.FU, e1);
        Education e2 = new Education("Ph.D", "English", 1);
        Faculty f2 = new Faculty("Bouris", "William", "791", Faculty.Level.AO, e1);
        Education e3 = new Education("MS", "Physical Education", 0);
        Faculty f3 = new Faculty("Andrade", "Christopher", "623", Faculty.Level.AS, e1);

        Partime p1 = new Partime("Guzman", "Augusto", "455", 35.00, 30);
        Partime p2 = new Partime("Depirro", "Martin", "678", 30.00, 15);
        Partime p3 = new Partime("Aldaco", "Marque", "945", 20.00, 35);

        //creates arraylist and adds object to list
        ArrayList<Employee> employees = new ArrayList<Employee>(9);
        employees.add(s1);
        employees.add(s2);
        employees.add(s3);
        employees.add(f1);
        employees.add(f2);
        employees.add(f3);
        employees.add(p1);
```

```

employees.add(p2);
employees.add(p3);

//goes through arraylist and prints out hte toString as well as monthly earnings of each employee
for(Employee temp: employees){
    total_monthly_salary_employees += temp.monthlyEarning();
    if(temp instanceof Partime){
        total_monthly_salary_partime += temp.monthlyEarning();
    }
    System.out.println(temp);
}

System.out.println("The total monthly salary for all part-time staff is: " + total_monthly_salary_partime);
System.out.println("The total monthly salary for all employees is: " + total_monthly_salary_employees);

//item d - Sort id in ascending order using Comparable (5 points)
ArrayList<String> id_list = new ArrayList<String>(employees.size());
for(Employee temp: employees){
    id_list.add(temp.getId());
}
Collections.sort(id_list);
System.out.println("Sorted id in ascending order using Comparable");
System.out.println(id_list);

//item e - Sort last name in descending order using Comparator (5 points)
Collections.sort(employees, new SortLastNameDescending());
System.out.println("Sorted last names descending order");
for(Employee temp: employees){
    System.out.println(temp.getLastName());
}
}
}

```

Run: Tester x

```

Sorted id in ascending order using Comparable
[123, 243, 455, 456, 623, 678, 789, 791, 945]
Sorted last names descending order
Zapata
Rios
Johnson
Guzman
Depirro
Bouris
Andrade
Allen
Aldaco

```

