Combining Data Structures
Sets of Objects

Arrays of Objects: one way of representing a set of objects

Problem: Create a list of employees of a department. First, ask how many employees there are, then compute the total amount earned by all employees.

1. Create an array of numEmployees Employee variables.

   ```java
   Employee[] emp = new Employee[numEmployees];
   ```

   Note that this does not create any objects of Employee class. The operator `new` is used to create an array!

2. Objects have to be created separately:

   ```java
   String empName;
   double empRate, hrs;
   
   for (int i = 0; i < numEmployees; i++)
   {
       // read next employee data
       System.out.print ("Please enter the name of the next employee");
       empName = SavitchIn.readLine();
       
       System.out.print ("Please enter the pay rate");
       empRate = SavitchIn.readLineDouble();
       
       // Create the employee object and store it (the reference)
       // in the array
       emp[i] = new Employee(empName, empRate);
       
       System.out.print ("Please enter how many hours worked today?");
       hrs = SavitchIn.readLineDouble();
       
       emp[i].IncrementHoursBy(hrs);
   }
   ```

3. Here’s an example of a traversal of this array: computing the total amount to be paid

   ```java
   double total = 0;
   for (int i = 0; i < numEmployees; i++)
   {
       total = total + emp[i].AmtDue();
   }
   ```

Container Objects

Problem: define a Department class to represent a department. Each department is a collection of employees.