

An ABC company, which is growing fast, has a need for storing the data in a database table.

For this they already have defined the structure of the table. We call this “employee” table. Following is the structure of the table.

| Data Type | Attribute | Short Description |
| --- | --- | --- |
| int | id | To identify each row uniquely |
| string | name | Name of an employee |
| int | age | Age of an employee |
| int | salary | Annual salary of an employee |
| string | department | Name of the department of an employee |
| string | city | Name of the city an employee lives in |

You have been engaged as a programmer and have to write the code for the following requirements.

* Display the list of all employees in the company
* Display the name of all the employees in alphabetical order
* Fetch a count of employees from each city
* Calculate the monthly salary of each employee

**Note**: Use a suitable data structure to implement the above requirements

To further ease the requirements the company has also shared the required sample output..

**List of all the employees**

*1 Aman 20 1100000 IT Delhi*

*2 Bobby 22 500000 Hr Bombay*

*3 Zoe 20 750000 Admin Delhi*

*4 Smitha 21 1000000 IT Chennai*

*5 Smitha 24 1200000 Hr Bengaluru*

**Name of all the employees in alphabetical order**

*[Aman, Bobby, Smitha, Smitha, Zoe]*

**Count of Employees from each city**

*{Bengaluru=1, Bombay=1, Chennai=1, Delhi=2}*

**Monthly Salary of employee along with their ids**

*{1=91666.0, 2=41666.0, 3=62500.0, 4=83333.0, 5=100000.0}*

****

**The company has defined certain guidelines which every programmer has to use. Following is that list of the same that you too have to adhere to**:

1. Use setters and getters to set or get the values of each attribute.
2. Follow Naming conventions (What kind of naming convention) throughout the code.
3. Create a Separate class for each of the data structure implementations.
4. Throw Illegal Argument Exception when

a) Employee Id < 0

b) Name is empty or null

c) Age < 0

d) Salary < 0

e) Department is empty or null

f) City is empty or null.

**Stub Code**

public class Employee {

int id;

String name;

int age;

int salary; // per annum

String department;

String city;

.

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ public class DataStructureA{

public void sortingNames(ArrayList<Employee> employees) {

.

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

public class DataStructureB {

public void cityNameCount(ArrayList<Employee> employees) {

.

.

public void monthlySalary(ArrayList<Employee> employees) {

.

.

. try{

}

Catch{}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

public static void main(String[] args) {

ArrayList<Employee> employees = new ArrayList<>();

Employee e1 = new Employee();

e1.setId(1);

.

.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Learning objective**: