

```
import java.util.ArrayList;
import java.util.List;
import java.util.Objects;

class Employee {

    private String firstname;
    private String lastname;
    private String id;

    public Employee(String firstname, String lastname, String id) {
        this.firstname = firstname;
        this.lastname = lastname;
        this.id = id;
    }

    public String getFirstname() {

        return firstname;
    }

    public void setFirstname(String firstname) {

        this.firstname = firstname;
    }

    public String getLastname() {
        return lastname;
    }

    public void setLastname(String lastname) {

        this.lastname = lastname;
    }

    public String getId() {

        return id;
    }

    public void setId(String id) {

        this.id = id;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        Employee employee = (Employee) o;
        return Objects.equals(firstname, employee.firstname) &&
            Objects.equals(lastname, employee.lastname) &&
            Objects.equals(id, employee.id);
    }

    @Override
    public int hashCode() {
        return Objects.hash(firstname, lastname, id);
    }

    @Override
    public String toString() {
        return "Employee{" +
            "firstname='" + firstname + '\'' +
            ", lastname='" + lastname + '\'' +
            ", id='" + id + '\'' +
            '}';
    }
}
```

```
}  
}  
  
public class lists {  
    public static void main(String[] args) {  
  
        // List<Integer> intList = new ArrayList<>();  
        //      or  
        // List<Integer> intList = new ArrayList<Integer>();  
  
        List<Employee> employeeList = new ArrayList<>();  
  
        // Directly adding instances to the arraylist without creating objects  
  
        employeeList.add(new Employee("Nayeem", "Mahmood", "C161026"));  
        employeeList.add(new Employee("Kawsar", "Chakladar", "C161001"));  
        employeeList.add(new Employee("Sa-ad", "Mahmood", "C161022"));  
        employeeList.add(new Employee("Abir", "Shanto", "C161007"));  
  
        employeeList.forEach(employee -> System.out.println(employee)); //Using Lamda Consumer  
        System.out.println();  
  
        // Creating Objects  
  
        Employee addedEmployee = new Employee("Mahir", "Shahrier", "C161023");  
  
        // Adding objects to the array list  
  
        employeeList.add(addedEmployee);  
  
        employeeList.forEach(employee -> System.out.println(employee)); //Using Lamda Consumer  
        System.out.println();  
  
        // Checking weather an instance or object is available in the list or not  
  
        System.out.println("Check if arraylist contains Mahir or not\t" + employeeList.contains(addedEmployee));  
        System.out.println("Check if arraylist contains Shanto or not\t" + employeeList.contains(new Employee("Abir",  
        "Shanto", "C161007")));  
        System.out.println();  
        // With the equal method Shanto will check true.  
        // Without the equal method Shanto will check false as this one is another distinct instance created for Shanto  
        // because we did not implement  
        // the equals method  
        // With or without the equal method Mahir will check true as there was an object of Employee created for Mahir  
        // and we searched using  
        // the object name therefore no need of implementing the equal method  
  
        // Equals method lets us use the index finding method to find the index or position  
  
        System.out.println("Shanto is in position\t" + employeeList.indexOf(new Employee("Abir", "Shanto", "C161007")));  
        System.out.println("Mahir is in position\t" + employeeList.indexOf(addedEmployee));  
  
    }  
}
```