**LISTS**

1. write a Java program to create an ArrayList, add all the months of a year and print the same.

**CODE**:

import java.util.ArrayList;

public class MonthsOfYear {

public static void main(String[] args) {

ArrayList<String> months = new ArrayList<>();

months.add("January");

months.add("February");

months.add("March");

months.add("April");

months.add("May");

months.add("June");

months.add("July");

months.add("August");

months.add("September");

months.add("October");

months.add("November");

months.add("December");

System.out.println("Months of the Year:");

for (String month : months) {

System.out.println(month);

}

}

}

**OUTPUT:**

Months of the Year:

January

February

March

April

May

June

July

August

September

October

November

December

1. Create an application for employee management with the following classes:

**a)** Create an Employee class with following attributes and behaviors:

i) int empId

ii)String empName

iii)String email

iv)String gender

v)float salary

vi) void GetEmployeeDetails()-> prints employee details

**CODE:**

import java.util.ArrayList;

import java.util.Enumeration;

import java.util.Iterator;

import java.util.Vector;

class Employee {

int empId;

String empName;

String email;

String gender;

float salary;

public Employee(int empId, String empName, String email, String gender, float salary) {

this.empId = empId;

this.empName = empName;

this.email = email;

this.gender = gender;

this.salary = salary;

}

public void GetEmployeeDetails() {

System.out.println("Employee ID : " + empId);

System.out.println("Name : " + empName);

System.out.println("Email : " + email);

System.out.println("Gender : " + gender);

System.out.println("Salary : " + salary);

System.out.println("-----------------------------");

}

}

public class EmployeeManagement {

public static void main(String[] args) {

Vector<Employee> employeeList = new Vector<>();

employeeList.add(new Employee(101, "Alice Smith", "alice@example.com", "Female", 55000));

employeeList.add(new Employee(102, "Bob Johnson", "bob@example.com", "Male", 60000));

employeeList.add(new Employee(103, "Charlie Davis", "charlie@example.com", "Male", 50000));

Iterator<Employee> iterator = employeeList.iterator();

count = 1;

while (iterator.hasNext()) {

System.out.println("Employee " + count + ":");

Employee emp = iterator.next();

emp.GetEmployeeDetails();

count++;

}

}

}

**OUTPUT:**

Employee 1:

Employee ID : 101

Name : Alice Smith

Email : alice@example.com

Gender : Female

Salary : 55000.0

-----------------------------

Employee 2:

Employee ID : 102

Name : Bob Johnson

Email : bob@example.com

Gender : Male

Salary : 60000.0

-----------------------------

Employee 3:

Employee ID : 103

Name : Charlie Davis

Email : charlie@example.com

Gender : Male

Salary : 50000.0

-----------------------------

**b)** Create one more class EmployeeDe with the following attributes and behaviors.

i)ArrayList list;

ii) boolean add Employee (Employee e) ->adds the employee object to the collection

iii) boolean delete Employee(int empId)->delete the employee object from the collection with the given empid

iv) String showPaySlip(int empId)->returns the payslip of the employee with the given empid

Provide implementation for all the methods and test your program

**CODE:**

import java.util.ArrayList;

class Employee {

int empId;

String empName;

String email;

String gender;

float salary;

public Employee(int empId, String empName, String email, String gender, float salary) {

this.empId = empId;

this.empName = empName;

this.email = email;

this.gender = gender;

this.salary = salary;

}

public void GetEmployeeDetails() {

System.out.println("Employee ID : " + empId);

System.out.println("Name : " + empName);

System.out.println("Email : " + email);

System.out.println("Gender : " + gender);

System.out.println("Salary : ₹" + salary);

System.out.println("-----------------------------");

}

}

class EmployeeDe {

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

public boolean addEmployee(Employee e) {

return list.add(e);

}

public boolean deleteEmployee(int empId) {

for (Employee e : list) {

if (e.empId == empId) {

list.remove(e);

return true;

}

}

return false;

}

public String showPaySlip(int empId) {

for (Employee e : list) {

if (e.empId == empId) {

return "Pay slip for Employee ID " + empId + ": Salary = ₹" + e.salary;

}

}

return "Employee not found!";

}

public void displayAllEmployees() {

if (list.isEmpty()) {

System.out.println("No employees available.");

} else {

int count = 1;

for (Employee e : list) {

System.out.println("Employee " + count + ":");

e.GetEmployeeDetails();

count++;

}

}

}

}

public class EmployeeManagementApp {

public static void main(String[] args) {

EmployeeDe empDe = new EmployeeDe();

empDe.addEmployee(new Employee(201, "Alice Smith", "alice@example.com", "Female", 55000));

empDe.addEmployee(new Employee(202, "Bob Johnson", "bob@example.com", "Male", 60000));

empDe.addEmployee(new Employee(203, "Catherine Lee", "cat@example.com", "Female", 65000));

empDe.displayAllEmployees();

System.out.println("\n=== PaySlip ===");

System.out.println(empDe.showPaySlip(202));

System.out.println("\n=== Delete Employee ID 202 ===");

boolean deleted = empDe.deleteEmployee(202);

System.out.println(deleted ? "Employee deleted successfully." : "Employee not found.");

System.out.println("\n=== Employees After Deletion ===");

empDe.displayAllEmployees();

}

}

**OUTPUT:**

Employee 1:

Employee ID : 201

Name : Alice Smith

Email : alice@example.com

Gender : Female

Salary : ₹55000.0

-----------------------------

Employee 2:

Employee ID : 202

Name : Bob Johnson

Email : bob@example.com

Gender : Male

Salary : ₹60000.0

-----------------------------

Employee 3:

Employee ID : 203

Name : Catherine Lee

Email : cat@example.com

Gender : Female

Salary : ₹65000.0

-----------------------------

=== PaySlip ===

Pay slip for Employee ID 202: Salary = ₹60000.0

=== Delete Employee ID 202 ===

Employee deleted successfully.

=== Employees After Deletion ===

Employee 1:

Employee ID : 201

Name : Alice Smith

Email : alice@example.com

Gender : Female

Salary : ₹55000.0

-----------------------------

Employee 2:

Employee ID : 203

Name : Catherine Lee

Email : cat@example.com

Gender : Female

Salary : ₹65000.0

1. Create an ArrayList that can store only Strings.

Create a printAll method that will print all the elements of the ArrayList using an Iterator.

Source code:

import java.util.ArrayList;

import java.util.Iterator;

import java.util.ListIterator;

public class StringListDemo {

public static void printAllUsingIterator(ArrayList<String> list) {

System.out.println("=== Printing using Iterator ===");

Iterator<String> iterator = list.iterator();

while (iterator.hasNext()) {

String element = iterator.next();

System.out.println(element);

}

}

public static void printAllUsingListIterator(ArrayList<String> list) {

System.out.println("\n=== Printing Forward using ListIterator ===");

ListIterator<String> listIterator = list.listIterator();

while (listIterator.hasNext()) {

System.out.println(listIterator.next());

}

System.out.println("\n=== Printing Backward using ListIterator ===");

while (listIterator.hasPrevious()) {

System.out.println(listIterator.previous());

}

}

public static void main(String[] args) {

ArrayList<String> stringList = new ArrayList<>();

stringList.add("Apple");

stringList.add("Banana");

stringList.add("Cherry");

stringList.add("Date");

stringList.add("Elderberry");

printAllUsingIterator(stringList);

printAllUsingListIterator(stringList);

}

}

Output:

=== Printing using Iterator ===

Apple

Banana

Cherry

Date

Elderberry

=== Printing Forward using ListIterator ===

Apple

Banana

Cherry

Date

Elderberry

=== Printing Backward using ListIterator ===

Elderberry

Date

Cherry

Banana

Apple

1. Create an ArrayList that can store only numbers like int, float, double, etc, but not any other data type.

Source code:

import java.util.ArrayList;

public class NumberListDemo {

public static void main(String[] args) {

ArrayList<Number> numberList = new ArrayList<>();

numberList.add(10);

numberList.add(15.5);

numberList.add(23.7f);

numberList.add(100L);

numberList.add((short)5);

numberList.add((byte)2);

System.out.println("Stored numbers in the list:");

for (Number num : numberList) {

System.out.println(num + " (type: " + num.getClass().getSimpleName() + ")");

}

}

}

**OUTPUT:**

Stored numbers in the list:

10 (type: Integer)

15.5 (type: Double)

23.7 (type: Float)

100 (type: Long)

5 (type: Short)

2 (type: Byte)

1. Implement the assignment 1 using Linked List

**CODE**

import java.util.LinkedList;

class Employee {

int empId;

String empName;

String email;

String gender;

float salary;

public Employee(int empId, String empName, String email, String gender, float salary) {

this.empId = empId;

this.empName = empName;

this.email = email;

this.gender = gender;

this.salary = salary;

}

public void GetEmployeeDetails() {

System.out.println("Employee ID : " + empId);

System.out.println("Name : " + empName);

System.out.println("Email : " + email);

System.out.println("Gender : " + gender);

System.out.println("Salary : ₹" + salary);

System.out.println("-----------------------------");

}

}

class EmployeeDe {

LinkedList<Employee> list = new LinkedList<>();

public boolean addEmployee(Employee e) {

return list.add(e);

}

public boolean deleteEmployee(int empId) {

for (Employee e : list) {

if (e.empId == empId) {

list.remove(e);

return true;

}

}

return false;

}

public String showPaySlip(int empId) {

for (Employee e : list) {

if (e.empId == empId) {

return "Pay slip for Employee ID " + empId + ": Salary = ₹" + e.salary;

}

}

return "Employee not found!";

}

public void displayAllEmployees() {

if (list.isEmpty()) {

System.out.println("No employees found.");

} else {

int count = 1;

for (Employee e : list) {

System.out.println("Employee " + count + ":");

e.GetEmployeeDetails();

count++;

}

}

}

}

public class LinkedListEmployeeApp {

public static void main(String[] args) {

EmployeeDe empDe = new EmployeeDe();

empDe.addEmployee(new Employee(101, "Alice Johnson", "alice@example.com", "Female", 56000));

empDe.addEmployee(new Employee(102, "Bob Williams", "bob@example.com", "Male", 60000));

empDe.addEmployee(new Employee(103, "Clara Martinez", "clara@example.com", "Female", 62000));

empDe.displayAllEmployees();

System.out.println("\n=== Pay Slip ===");

System.out.println(empDe.showPaySlip(102));

System.out.println("\n=== Deleting Employee with ID 102 ===");

if (empDe.deleteEmployee(102)) {

System.out.println("Employee deleted successfully.");

} else {

System.out.println("Employee not found.");

}

System.out.println("\n=== Employees After Deletion ===");

empDe.displayAllEmployees();

}

}

**OUTPUT:**

Employee 1:

Employee ID : 101

Name : Alice Johnson

Email : alice@example.com

Gender : Female

Salary : ₹56000.0

-----------------------------

Employee 2:

Employee ID : 102

Name : Bob Williams

Email : bob@example.com

Gender : Male

Salary : ₹60000.0

-----------------------------

Employee 3:

Employee ID : 103

Name : Clara Martinez

Email : clara@example.com

Gender : Female

Salary : ₹62000.0

-----------------------------

=== Pay Slip ===

Pay slip for Employee ID 102: Salary = ₹60000.0

=== Deleting Employee with ID 102 ===

Employee deleted successfully.

=== Employees After Deletion ===

Employee 1:

Employee ID : 101

Name : Alice Johnson

Email : alice@example.com

Gender : Female

Salary : ₹56000.0

-----------------------------

Employee 2:

Employee ID : 103

Name : Clara Martinez

Email : clara@example.com

Gender : Female

Salary : ₹62000.0

1. Implement the assignment 1 using Vector

**CODE**

import java.util.Vector;

class Employee {

int empId;

String empName;

String email;

String gender;

float salary;

public Employee(int empId, String empName, String email, String gender, float salary) {

this.empId = empId;

this.empName = empName;

this.email = email;

this.gender = gender;

this.salary = salary;

}

public void GetEmployeeDetails() {

System.out.println("Employee ID : " + empId);

System.out.println("Name : " + empName);

System.out.println("Email : " + email);

System.out.println("Gender : " + gender);

System.out.println("Salary : ₹" + salary);

System.out.println("-----------------------------");

}

}

class EmployeeDe {

Vector<Employee> list = new Vector<>();

public boolean addEmployee(Employee e) {

return list.add(e);

}

public boolean deleteEmployee(int empId) {

for (Employee e : list) {

if (e.empId == empId) {

list.remove(e);

return true;

}

}

return false;

}

public String showPaySlip(int empId) {

for (Employee e : list) {

if (e.empId == empId) {

return "Pay slip for Employee ID " + empId + ": Salary = ₹" + e.salary;

}

}

return "Employee not found!";

}

public void displayAllEmployees() {

if (list.isEmpty()) {

System.out.println("No employees found.");

} else {

int count = 1;

for (Employee e : list) {

System.out.println("Employee " + count + ":");

e.GetEmployeeDetails();

count++;

}

}

}

}

public class VectorEmployeeApp {

public static void main(String[] args) {

EmployeeDe empDe = new EmployeeDe();

empDe.addEmployee(new Employee(201, "John Doe", "john@example.com", "Male", 50000));

empDe.addEmployee(new Employee(202, "Jane Smith", "jane@example.com", "Female", 60000));

empDe.addEmployee(new Employee(203, "Mark Taylor", "mark@example.com", "Male", 55000));

System.out.println("=== All Employees ===");

empDe.displayAllEmployees();

System.out.println("\n=== Pay Slip ===");

System.out.println(empDe.showPaySlip(202));

System.out.println("\n=== Deleting Employee ID 202 ===");

if (empDe.deleteEmployee(202)) {

System.out.println("Employee deleted successfully.");

} else {

System.out.println("Employee not found.");

}

System.out.println("\n=== Remaining Employees ===");

empDe.displayAllEmployees();

}

}

OUTPUT:

=== All Employees ===

Employee 1:

Employee ID : 201

Name : John Doe

Email : john@example.com

Gender : Male

Salary : ₹50000.0

-----------------------------

Employee 2:

Employee ID : 202

Name : Jane Smith

Email : jane@example.com

Gender : Female

Salary : ₹60000.0

-----------------------------

Employee 3:

Employee ID : 203

Name : Mark Taylor

Email : mark@example.com

Gender : Male

Salary : ₹55000.0

-----------------------------

=== Pay Slip ===

Pay slip for Employee ID 202: Salary = ₹60000.0

=== Deleting Employee ID 202 ===

Employee deleted successfully.

=== Remaining Employees ===

Employee 1:

Employee ID : 201

Name : John Doe

Email : john@example.com

Gender : Male

Salary : ₹50000.0

-----------------------------

Employee 2:

Employee ID : 203

Name : Mark Taylor

Email : mark@example.com

Gender : Male

Salary : ₹55000.0

1. Write a program that will have a Vector which is capable of storing Employee objects. Use an Iterator and enumeration to list all the elements of the Vector.

**CODE**

import java.util.\*;

class Employee {

int empId;

String empName;

float salary;

public Employee(int empId, String empName, float salary) {

this.empId = empId;

this.empName = empName;

this.salary = salary;

}

public void printDetails() {

System.out.println("ID: " + empId + ", Name: " + empName + ", Salary: ₹" + salary);

}

}

public class VectorEmployeeExample {

public static void main(String[] args) {

Vector<Employee> empVector = new Vector<>();

empVector.add(new Employee(101, "Alice", 50000));

empVector.add(new Employee(102, "Bob", 60000));

empVector.add(new Employee(103, "Charlie", 55000));

System.out.println("=== Using Iterator ===");

Iterator<Employee> iterator = empVector.iterator();

while (iterator.hasNext()) {

Employee emp = iterator.next();

emp.printDetails();

}

Enumeration<Employee> enumeration = empVector.elements();

while (enumeration.hasMoreElements()) {

Employee emp = enumeration.nextElement();

emp.printDetails();

}

}

}

**OUTPUT:**

ID: 101, Name: Alice, Salary: ₹50000.0

ID: 102, Name: Bob, Salary: ₹60000.0

ID: 103, Name: Charlie, Salary: ₹55000.0

=== Using Enumeration ===

ID: 101, Name: Alice, Salary: ₹50000.0

ID: 102, Name: Bob, Salary: ₹60000.0

ID: 103, Name: Charlie, Salary: ₹55000.0