|  |  |  |
| --- | --- | --- |
| **Name of Student: Pushkar Prasad Sane** | | |
| **Roll Number: 45** | | **Lab Assignment Number: 1** |
| **Title of Lab Assignment: Assignment based on Collection.** | | |
| **DOP: 23-08-2023** | | **DOS: 06-09-2023** |
| **CO Mapped:**  **CO1, CO2** | **PO Mapped:**  **PO1, PO2, PO3, PSO1** | **Signature:** |

**PRACTICAL 1**

**Aim:**

1. **List Interface**
2. Create an ArrayList of type Integer, add elements into it, traverse the arraylist and print the elements.

**Code:**

package demo;

import java.util.ArrayList;

import java.util.Iterator;

public class demo{

public static void main(String args[]) {

ArrayList<Integer> arr = new ArrayList <Integer>();

arr.add(100);

arr.add(101);

arr.add(102);

Iterator<Integer> itr = arr.iterator();

while(itr.hasNext()) {

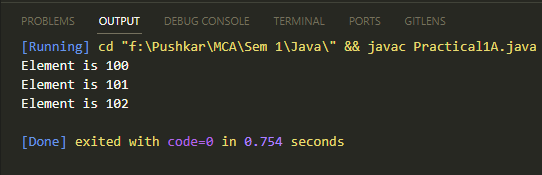
System.out.println("Element is "+ itr.next());

}

}

}

**Output:**

****

1. Create a LinkedList of type String, add 5 elements and traverse the list from both sides.

**Code:**

package demo;

import java.util.\*;

public class demo{

public static void main(String args[]) {

LinkedList<String> al1 = new LinkedList<String>();

al1.add("Peter");

al1.add("Harry");

al1.add("Hagrid");

al1.add("Ron");

al1.add("John");

System.out.println("Forward:");

ListIterator<String> itr = al1.listIterator();

while(itr.hasNext()) {

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

}

System.out.println("\nReverse:");

while(itr.hasPrevious()) {

while(itr.hasPrevious()) {

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

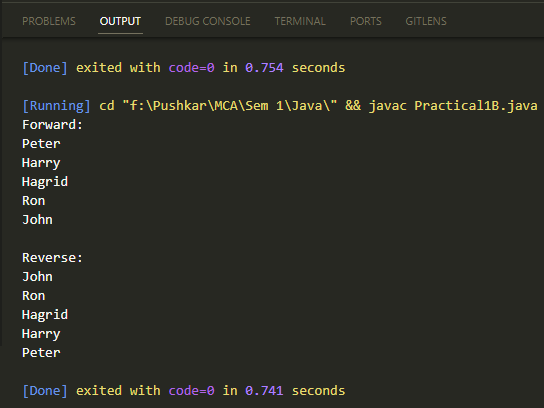
}

}

}

}

**Output:**

****

1. Create an employee class (id, name, salary) create an Arralist of type employee, add 5 employees, traverse the ArrayList and print the elements, Remove one element and print the list.

**Code:**

package demo;

import java.util.ArrayList;

import java.util.Iterator;

class employee{

int id;

String name;

int salary;

employee(int id, String name, int salary){

this.id = id;

this.name = name;

this.salary = salary;

}

}

public class demo{

public static void main(String args[]) {

employee e1 = new employee(1, "Peter", 30000);

employee e2 = new employee(2, "John", 40000);

employee e3 = new employee(3, "Harry", 20000);

employee e4 = new employee(4, "Hagrid", 70000);

employee e5 = new employee(5, "Ron", 10000);

ArrayList<employee> al1 = new ArrayList<employee>();

al1.add(e1);

al1.add(e2);

al1.add(e3);

al1.add(e4);

al1.add(e5);

Iterator itr = al1.iterator();

while(itr.hasNext()) {

employee emp = (employee)itr.next();

System.out.println(emp.id + emp.name + emp.salary);

}

al1.remove(e4);

Iterator itr1 = al1.iterator();

while(itr1.hasNext()) {

employee emp1 = (employee)itr1.next();

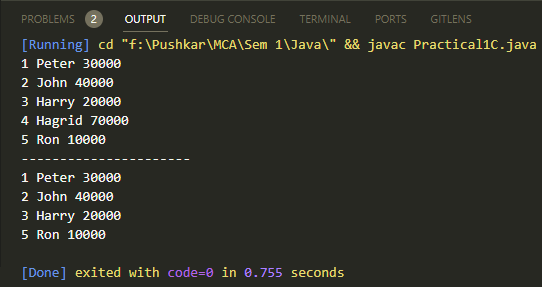
System.out.println(emp1.id + emp1.name + emp1.salary);

}

}

}

**Output:**

****

1. **Set Interface**
2. **Write a Java program using Set interface containing list of items and perform the following operations:** 
   1. **Add items in the set.**
   2. **Insert items of one set into another set.**
   3. **Remove items from the set.**
   4. **Search the specified item in the set.**

**Code:**

package practicals;

import java.util.HashSet;

public class SetInterfaceExample {

public static void main(String[] args) {

HashSet<String> colors = new HashSet<>();

HashSet<String> shapes = new HashSet<>();

colors.add("Red");

colors.add("Green");

colors.add("Brown");

colors.add("Yellow");

shapes.add("Square");

shapes.add("Triangle");

shapes.add("Circle");

System.out.println("Colors: " + colors);

System.out.println("Shapes: " + shapes);

colors.addAll(shapes);

System.out.println("Colors and shapes: " + colors);

colors.remove("Yellow");

System.out.println("Updated colors: " + colors);

String search = "Circle";

if (colors.contains(search)) {

System.out.println(search + " is found in colors.");

} else {

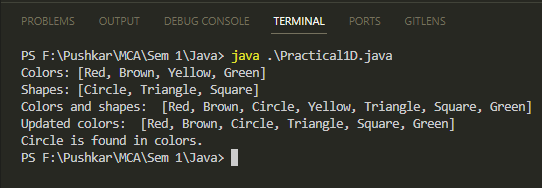
System.out.println(search + " is not found in colors.");

}

}

}

**Output:**

****

1. **Map Interface**
2. **Create a class Customer(Account\_no Integer, Name Sting), Create a HashMap of type Customer put elements, print elements, check if element with account number 101 is present or not? What is the value for Customer 101.**

**Code:**

package practicals;

import java.util.HashMap;

import java.util.Iterator;

class Customer {

private Integer Account\_no;

private String Name;

public Customer(Integer accountNo, String name) {

this.Account\_no = accountNo;

this.Name = name;

}

public Integer getAccountNo() {

return Account\_no;

}

public String getName() {

return Name;

}

@Override

public String toString() {

return "Account\_no = " + Account\_no + ", Name = " + Name;

}

}

public class MapInterfaceExample {

public static void main(String[] args) {

HashMap<Integer, Customer> cMap = new HashMap<>();

// Adding customers

cMap.put(101, new Customer(101, "Saitama"));

cMap.put(102, new Customer(102, "Genos"));

cMap.put(103, new Customer(103, "King"));

// Print the elements in the HashMap

System.out.println("Customers :");

Iterator<Customer> customerIterator = cMap.values().iterator();

while (customerIterator.hasNext()) {

Customer customer = customerIterator.next();

System.out.println(customer);

}

// Check if Account\_no 101 is present

int Check = 101;

if (cMap.containsKey(Check)) {

System.out.println("Customer with Account\_no " + Check + " is present.");

} else {

System.out.println("Customer with Account\_no " + Check + " is not present.");

}

// print the value of Customer 101

Customer customer101 = cMap.get(Check);

if (customer101 != null) {

System.out.println("Value for Customer 101: " + customer101);

} else {

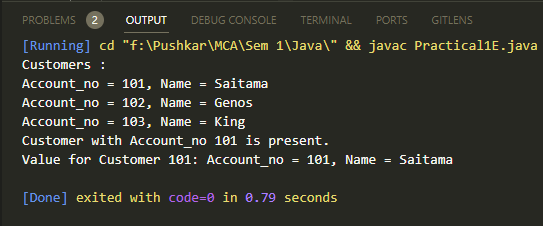
System.out.println("Customer 101 not found in the HashMap.");

}

}

}

**Output:**

****