Md Yaseen (mdyaseeny589@gmail.com)

Day – 3 Assignment

Task – 1:

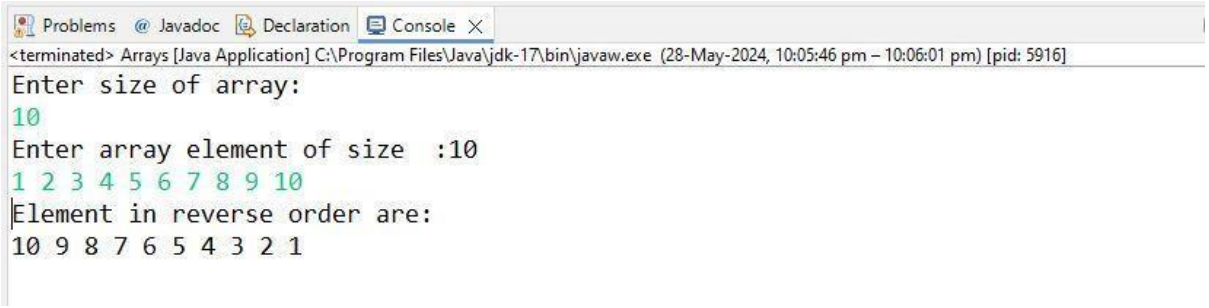
Arrays - Declaration, Initialization, and Usage

Create a program that declares an array of integers, initializes it with consecutive numbers, and prints the array in reverse order.

CODE:

```
package com.assignment.day3;

import java.util.Scanner;

public class Arrays {

	public static void main(String[] args) {

		Scanner sc = new Scanner(System.in);

		// Asking user to enter size of array
		System.out.println("Enter size of array:");
		int n = sc.nextInt();

		// declaring array of size n;
		int[] array = new int[n];

		// taking array element input from user
		System.out.println("Enter array element of size
:" + n);

		// using for loop to store the element
		for(int i=0;i<n;i++) {
			array[i] = sc.nextInt();
		}

		// printing element in reverse order
	System.out.println("Element in reverse order are: ");
		for(int i=n-1;i>=0;i--) {
			System.out.print(array[i] + " ");
		}
	}
}
```

OUTPUT:



Task – 2:

List interface

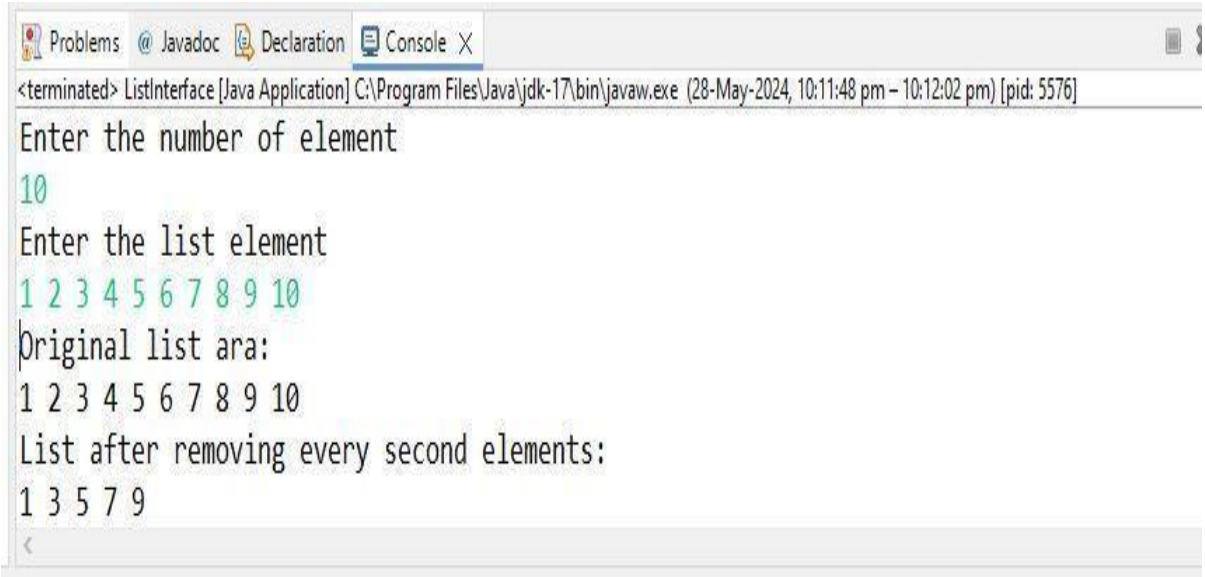
Implement a method that takes a List as an argument and removes every second element from the list, then prints the resulting list.

CODE:

```java
package com.assignment.day3;

import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.Scanner;

public class ListInterface {


	public static void main(String[] args) {
		Scanner sc = new Scanner(System.in);

		System.out.println("Enter the number of element");
		int n = sc.nextInt();

		System.out.println("Enter the list element");
		// Declaring list of integers
		List<Integer> list = new ArrayList<>();

		// Storing users input in list
		for(int i=0;i<n;i++) {
			list.add(sc.nextInt());
		}

		// Displaying original list
		System.out.println("Original list ara: ");
		display(list);

		// calling method to remove every 2nd element from the
list
		removeEverySecondElement(list);

		// displaying modified list
		System.out.println("List after removing every second
elements: ");
		display(list);
	}
```

```
// function to display list element
	private static void display(List<Integer> list) {
		// using forEach loop to access the element
		for(int num : list)
			System.out.print(num + " ");

		System.out.println();
	}

	// function to delete the every second element from the list
	private static void removeEverySecondElement(List<Integer>
list) {
		// creating iterator for list
		Iterator<Integer> it = list.iterator();
		int idx = 0;
		while(it.hasNext()) { // check whether next iterator
exists or not
			it.next(); // move to next iterator
			if(idx%2 != 0) { // checking current idx is second
index or not
				it.remove(); // removing element
			}
			idx++; // incrementing index
		}
	}
}
```

OUTPUT:



Task – 3:

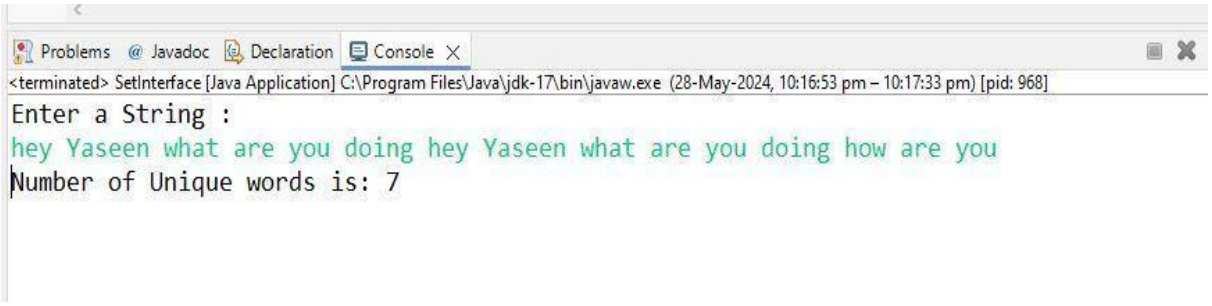
Set interface

Write a program that reads words from a String variable into a Set and prints out the number of unique words, demonstrating the unique property of sets.

CODE:

```
package com.assignment.day3;

import java.util.HashSet;
import java.util.Scanner;
import java.util.Set;

public class SetInterface {
	public static void main(String[] args) {
		Scanner sc = new Scanner(System.in);

		// taking String input from user
		System.out.println("Enter a String : ");
		String str = sc.nextLine();

		String[] strArray = str.split("\\s+");

		// Declaring Set
		Set<String> set = new HashSet<>();

		// Adding words in String to count unique number of
words

		for(String word: strArray) {
			set.add(word);
		}

		// Displaying result
		System.out.println("Number of Unique words is: " +
set.size());


	}

}
```

OUTPUT:



Task – 4:

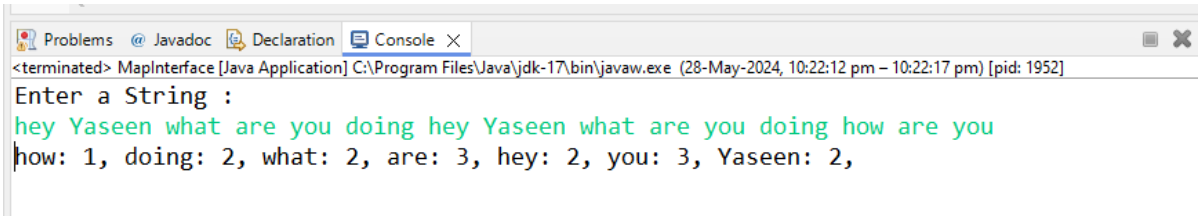
Map interface

Create a Java class that uses a Map to store the frequency of each word that appears in a given string.

CODE:

```
package com.assignment.day3;

import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class MapInterface {
	public static void main(String[] args) {
		Scanner sc = new Scanner(System.in);

		// taking String input from user
		System.out.println("Enter a String : ");
		String str = sc.nextLine();

		// converting string to string array
		String[] strArray = str.split("\\s+");

		// declaring a Map of String:Integer type
		Map<String,Integer> map = new HashMap<>();

		// storing the frequency of each words
		for(String word : strArray) {
			map.put(word, map.getOrDefault(word, 0)+1);
		}

		// displaying frequency of each words
		for(String key : map.keySet()) {
			System.out.print(key + ": " + map.get(key) + ", ");
		}

	}
}
```

OUTPUT:



Task – 5:

Iterators and Comparators

Write a custom Comparator to sort a list of Employee objects by their salary and then by name if the salary is the same.

CODE:

```
package com.assignment.day3;

import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;

class Employee {
	private int id;
	private String name;
	private int salary;
	private int age;

	public Employee(int id, String name, int salary, int age) {
		this.id = id;
		this.name = name;
		this.salary = salary;
		this.age = age;
	}

	@Override
	public String toString() {
		return "Employee [id=" + id + ",name=" + getName() + ",
salary=" + getSalary() + ",age=" + age +  " ]";
	}

	public String getName() {
		return name;
	}

	public int getSalary() {
		return salary;
	}

}
```

```
class EmployeeComparator implements Comparator<Employee> {
     @Override
     public int compare(Employee e1,Employee e2) {
          int sal = Integer.compare(e2.getSalary(),
e1.getSalary());
          if(sal != 0) return sal;

          return e1.getName().compareTo(e2.getName());
     }
}
public class ComparatorIteratorEx {
     public static void main(String[] args) {
          List<Employee> emp = new ArrayList<>();
          emp.add(new Employee(1,"Yaseen",7500000,23));
          emp.add(new Employee(2,"Alok",450000,24));
          emp.add(new Employee(3,"Shadan",450000,26));
          emp.add(new Employee(4,"Kuldeep",750000,25));

          Collections.sort(emp,new EmployeeComparator());
          for(Employee em : emp) {
               System.out.println(em);
          }

     }

}
```

OUTPUT: