***Dt : 25/10/2022***

***\*imp***

***Spliterator<T>:***

***=>Spliterator<T> is an interface introduced by Java8 version and which is***

***used to retrieve elements from Array Objects and Collection<E> objects.***

***=>The following is one important method of Spliterator<T>:***

***public default void forEachRemaining(java.util.function.Consumer<? super T>);***

***=>we use spliterator() method from java.util.Arrays class to create***

***implementation object for Spliterator<T> interface.***

***syntax:***

***Spliterator<T> ob = Arrays.spliterator(arr\_var);***

***=>This Spliterator<T> object will hold the reference of Array object.***

***define Consumer<T>?***

***=>Consumer<T> is a Functional interface introduced by Java8 version and***

***which provide abstract method "accept()" to hold LambdaExpression passed as***

***parameter to forEachRemaining() method.***

***Structure of Consumer<T>:***

***public interface java.util.function.Consumer<T>***

***{***

***public abstract void accept(T);***

***}***

***LambdaExpression syntax:***

***Consumer<T> obj = (T)->***

***{***

***...***

***};***

***Ex-Program:***

***LambdaExpression3.java(MainClass)***

***package maccess;***

***import java.util.\*;***

***public class LambdaExpression3 {***

***public static void main(String[] args) {***

***Scanner s = new Scanner(System.in);***

***System.out.println("Enter the size of Array:");***

***int n = s.nextInt();***

***Integer a[] = new Integer[n];***

***System.out.println("Enter "+n+" Integer elements");;***

***for(int i=0;i<a.length;i++)***

***{***

***a[i] = new ~~Integer~~(s.nextInt());***

***}//end of loop***

***Spliterator<Integer> ob = Arrays.spliterator(a);***

***//Implementation object of Spliterator Interface***

***System.out.println("====Spliterator<T>====");***

***ob.forEachRemaining((k)->***

***{***

***System.out.print(k.toString()+" ");***

***});***

***s.close();***

***}***

***}***

***o/p:***

***Enter the size of Array:***

***5***

***Enter 5 Integer elements***

***11***

***12***

***13***

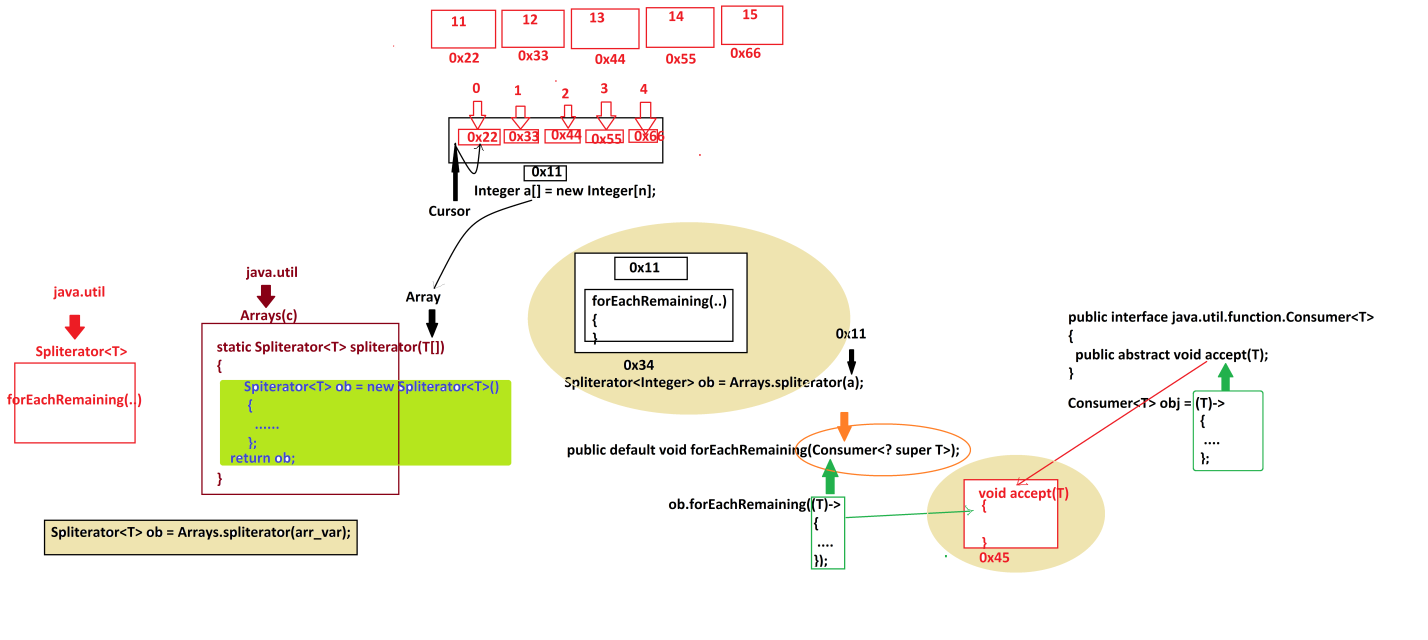
***14***

***15***

***====Spliterator<T>====***

***11 12 13 14 15***

***Diagram:***

******

***=============================================================***