|  |  |
| --- | --- |
| **is** | **CSC-150 – Object Oriented Programming** |
| **Semester (Summer 2024)**  **Course Instructors: Dr M Hussain Mughal** |

|  |
| --- |
| **Lab 03**  **Arrays and Java Class Libraries** |

**Objectives:**

1. Arrays in JAVA
2. Java Math Class

|  |
| --- |
| **1: Arrays in JAVA** |

An array stores a sequence of values that are all of the same type. The length of an array is established when the array is created. After creation, its length is fixed. Each item in an array is called an *element,* and each element is accessed by its numerical *index*.

The method that we use to refer to individual values in an array is to number and then index them—if we have n values, we think of them as being numbered from 0 to n−1.

Making an array in a Java program involves three distinct steps:

* Declare the array name.
* Declaration and memory allocation.
* Declare and Initialize the array values.

We refer to an array element by putting its index in square brackets after the array name.

To use an array in a program, you must declare a variable to reference the array and specify the array’s *element type*.

*Syntax:*

elementType[] arrayRefVar;

The **elementType** can be any data type, and all elements in the array will have the same data type.

Unlike declarations for primitive data type variables, the declaration of an array variable does not allocate any space in memory for the array. It creates only a storage location for the reference to an array. If a variable does not contain a reference to an array, the value of the variable is **null**. You cannot assign elements to an array unless it has already been created. After an array variable is declared, you can create an array by using the **new** operator and assign its reference to the variable with the following **syntax**:

arrayRefVar = new elementType[arraySize];

Java has a shorthand notation, known as the *array initializer*, which combines the declaration, creation, and initialization of an array in one statement using the following **syntax**:

elementType[] arrayRefVar = {value0, value1, ..., valuek};

**Arrays Class**

Arrays class which is in java.util.Arrays package, is a provision by Java that provides you a number of methods through which arrays can be manipulated. This class also lets you perform sorting and searching operations on an array.

**Array Declaration**

Example: //Declaring Array

double myArray[];

**Array Instantiation**

Example: //Instantiation of an Array;

int myArrary[] = new int[2];

**Array Initialization**

Example: //Intialization of an Array

myArrary[0] = 1;

myArrary[1] = 2;

**Example: //Accessing Array elements using Loop**

for(int a = 0; a<myArrary.length; a++){

System.out.println(myArrary[a]); }

**//Array using foreach loop**

**Example:** int numbers[]={1,2,3,4};

for(int j:numbers){

System.out.println(j);

}

|  |
| --- |
| **2: JAVA Math Class** |

The Java programming language supports basic arithmetic with its arithmetic operators: +, -, \*, /, and %. The [Math](https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html" \t "_blank)class provides methods and constants for doing more advanced mathematical computation.

The Math is located in the java.lang package, and not in the java.math package. Thus, the fully qualified class name of the Math class is java.lang.Math

The methods in the Math class are all static, so you call them directly from the class, like this:

Math.cos(angle);

## **Constants**

The Math class includes two constants:

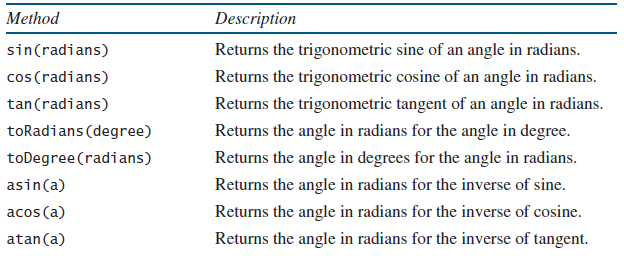
* Math.E, which is the base of natural logarithms, and
* Math.PI, which is the ratio of the circumference of a circle to its diameter.

**Basic Math Methods**

The Math class includes more than 40 static methods. They can be categorized as *trigonometric methods*, *exponent methods*, and *service methods*. Service methods include the rounding, min, max, absolute, and random methods.

**Trigonometric Methods**

The Mathclass contains the following methods.

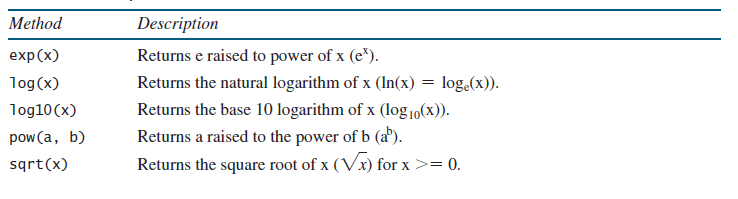


The parameter for sin, cos, and tanis an angle in radians. The return value for asin, acos, and atanis a degree in radians in the range between -pi/2 and pi/2.

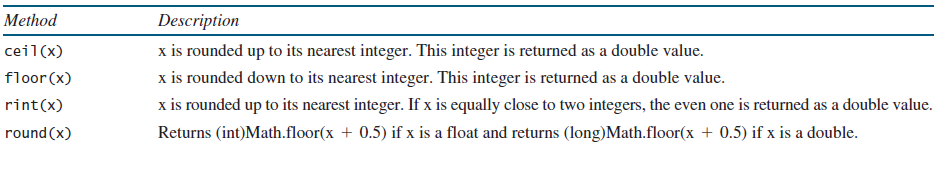
* One degree is equal to pi/180 in radians,
* 90 degrees is equal to pi/2 in radians
* 30 degrees is equal to pi/6 in radians.

**Exponent Methods**

There are five methods related to exponents in the Math class.



**The Rounding Methods**

The Mathclass contains five rounding methods

**The Service Methods**

The **min**, **max**, and **abs** Methods

The **min** and **max** methods return the minimum and maximum numbers of two numbers (**int**, **long**, **float**, or **double**).

For example, **max(4.4, 5.0)** returns **5.0**, and **min(3, 2)** returns **2**.

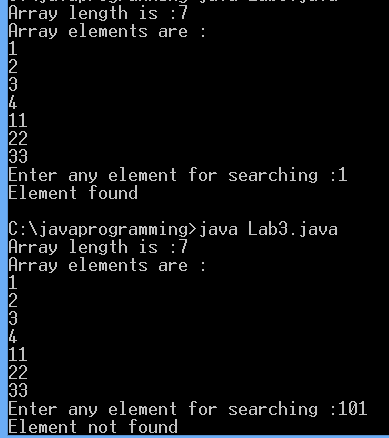
The **abs** method returns the absolute value of the number (**int**, **long**, **float**, or **double**).

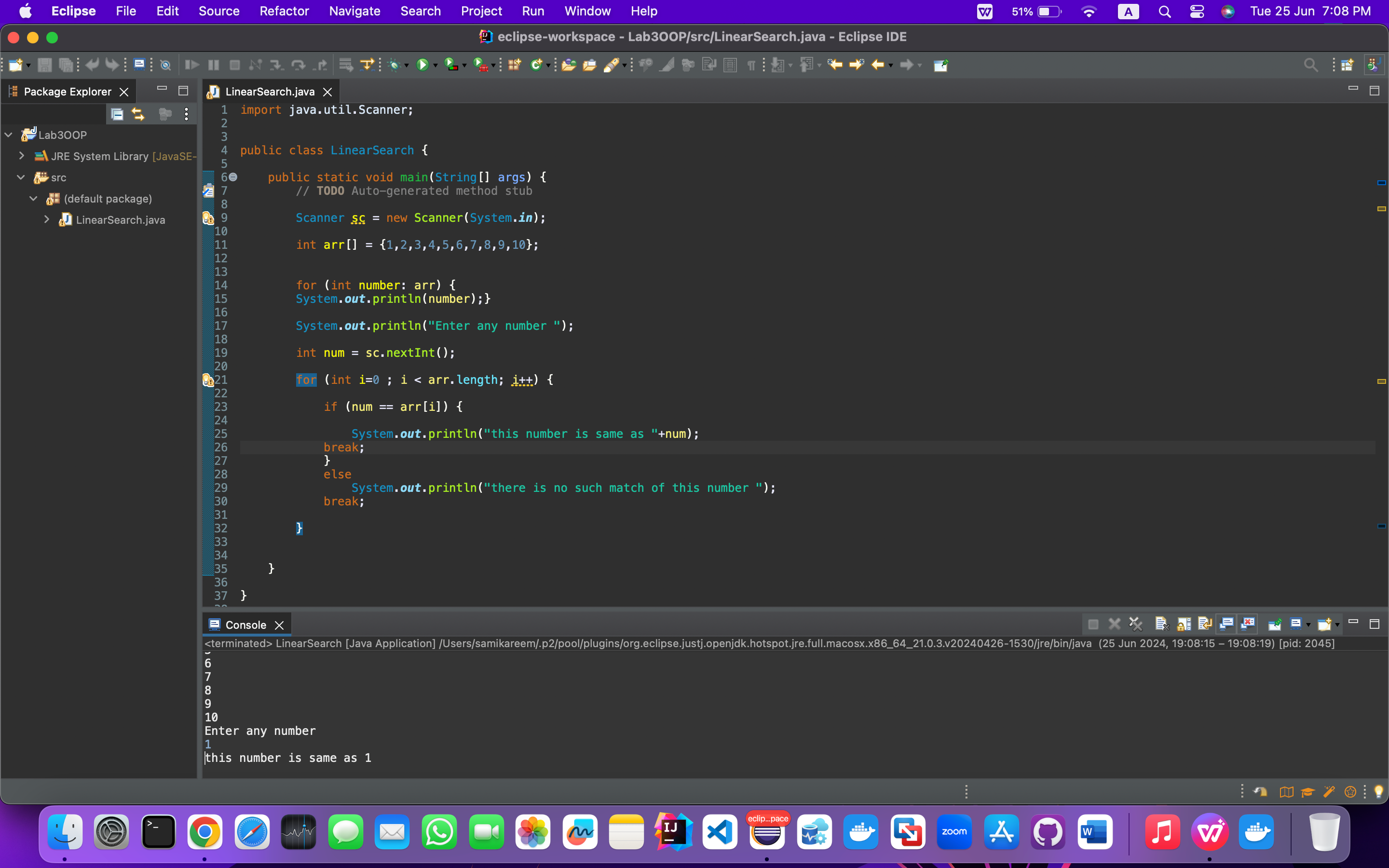
This method generates a random **double** value greater than or equal to 0.0 and less than 1.0 (**0 <= Math.random() < 1.0**). You can use it to write a simple expression to generate random numbers in any range.

**Lab Tasks:**

Exercises

1. Write a Java program which program linear search. It takes user input; if the elment is found in the array it display element is found or not found. As shown in the following figure.

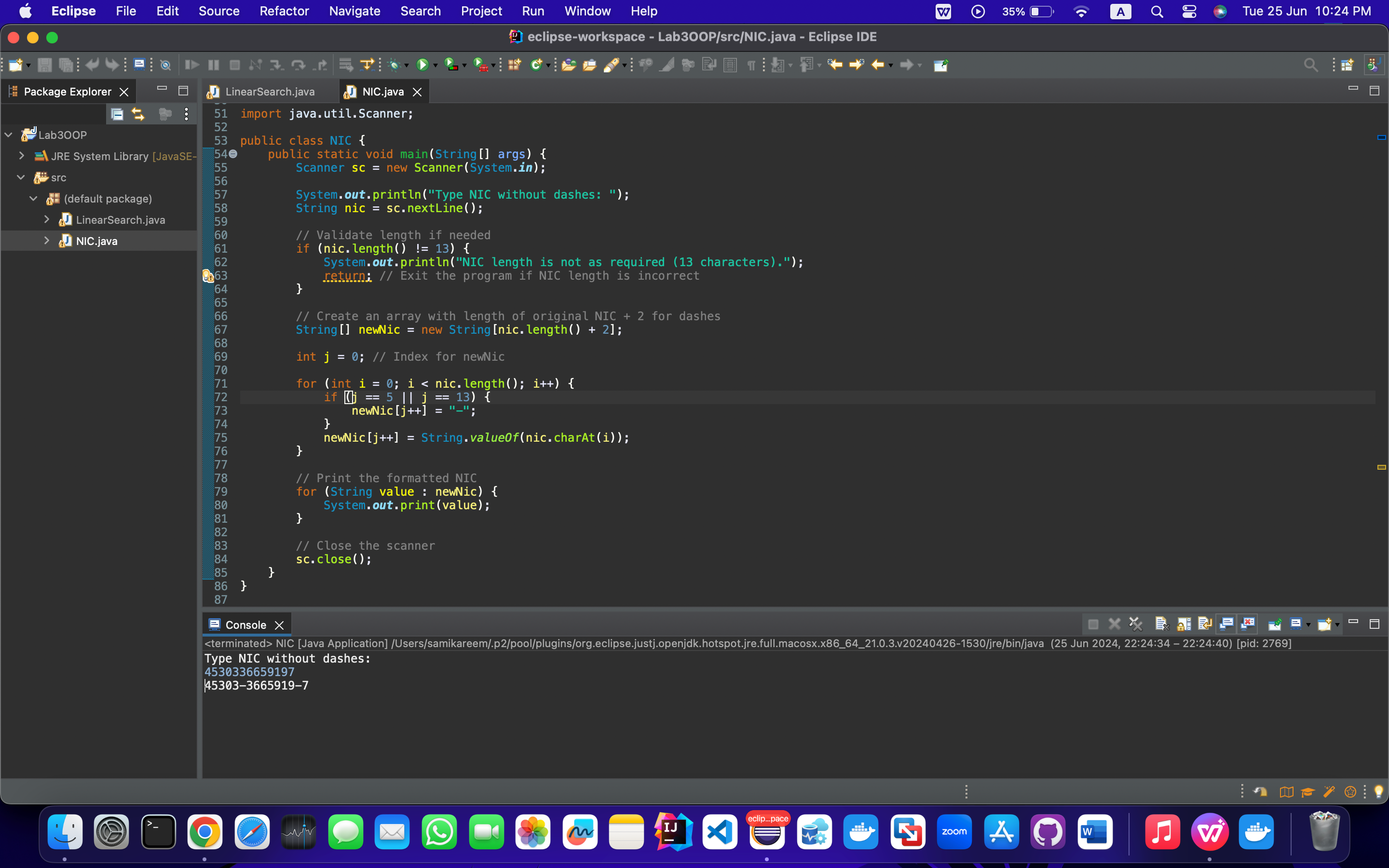




1. Write a Java program which takes user CNIC as an input without dashesh. It display CNIC with dashes. As shown below

Enter any CNIC : 4511122334459

Output: 45111-2233445-9



1. Write a Java program that creates an array which takes odd numbers from 1-10 and display total sum of odd numbers.

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

Odd Numbers 1-10

Enter odd numebr: 1

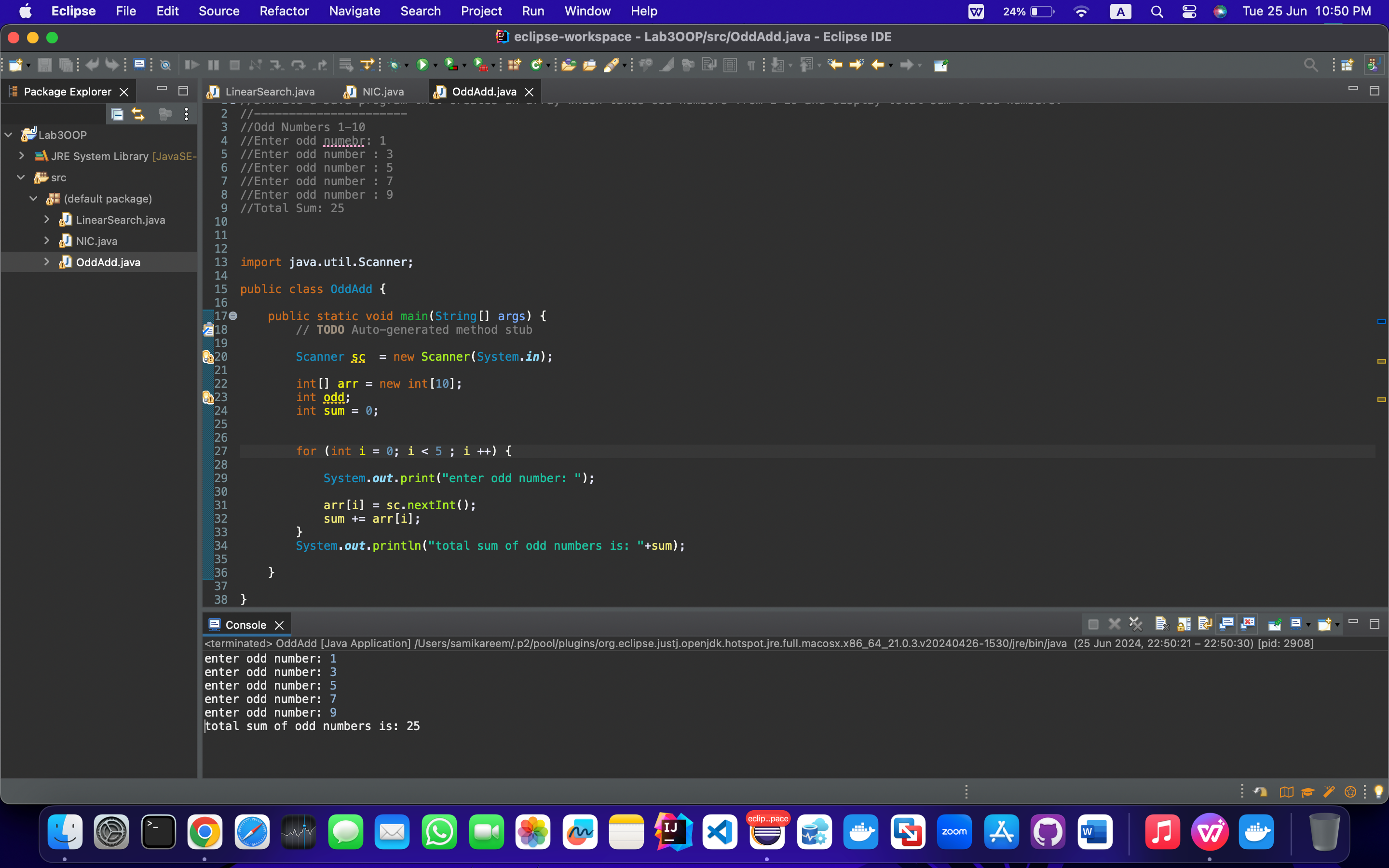
Enter odd number : 3

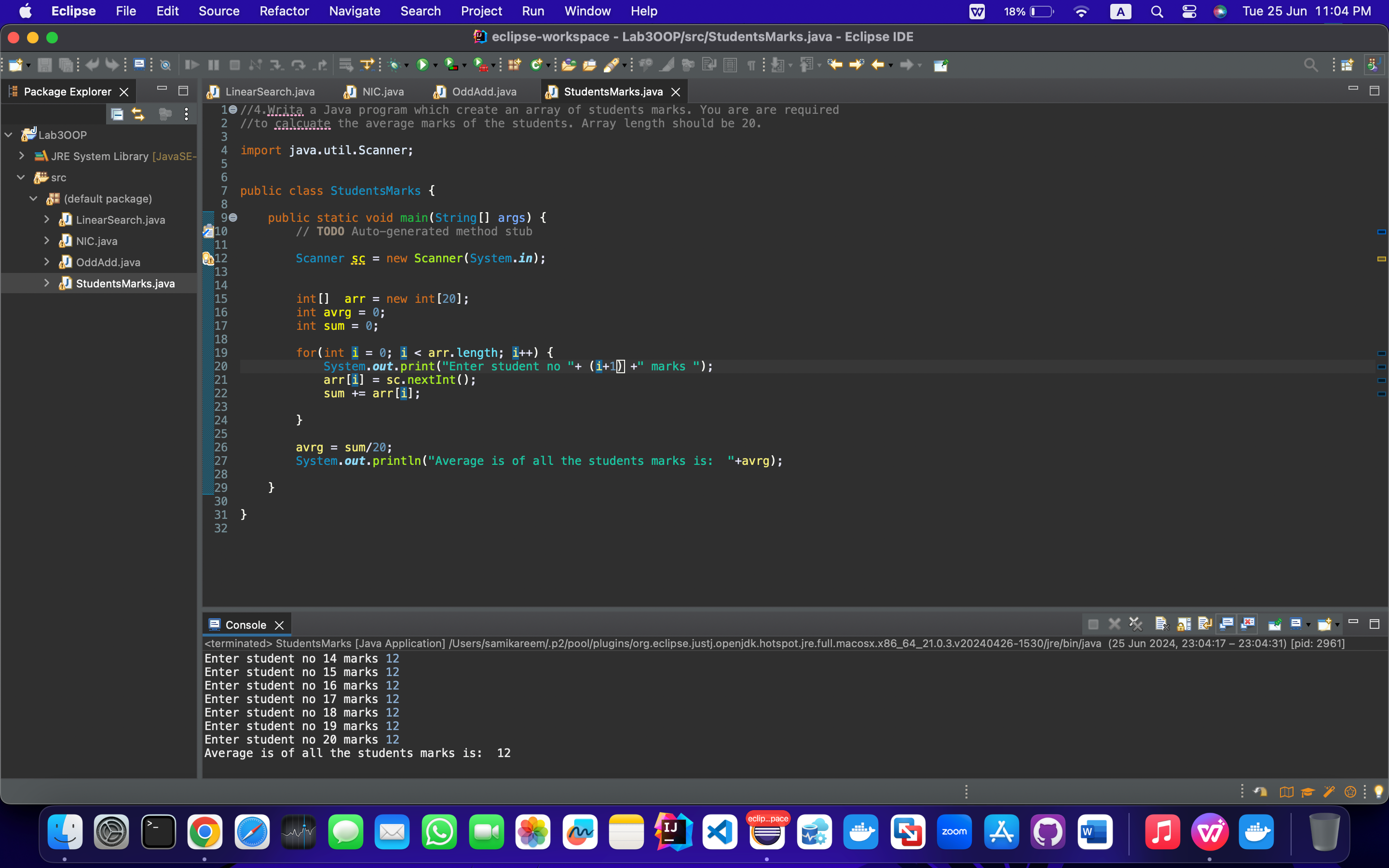
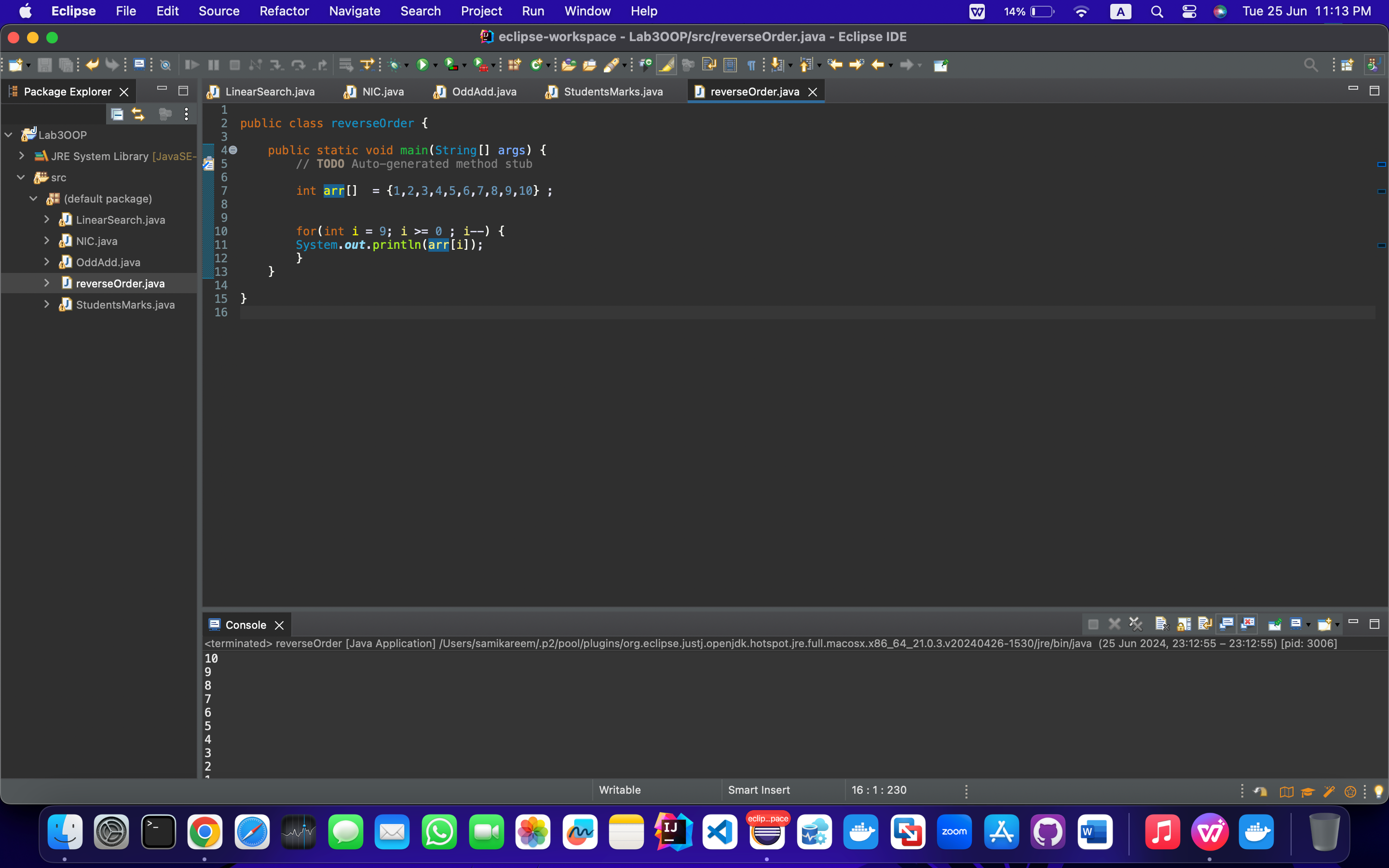
Enter odd number : 5

Enter odd number : 7

Enter odd number : 9

Total Sum: 25



1. Writa a Java program which create an array of students marks. You are are required to calcuate the average marks of the students. Array length should be 20.
2. 
3. Write a Java program to print the elements of an array in reverse order.
4. 
5. Write a Java program which convert degrees into radians and radians into degrees.

