# CS500-01 Fundamentals of Programming

# Assignment 2

Tharinda Embuldeniya

S01996041

# Question 6.35

*Code*

**import** java.util.Scanner;

**public** **class** AreaOfPentagon {

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

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

System.***out***.println("Enter the side" );

// user input data

**double** side = input.nextDouble();

//output the result + calling area method

System.***out***.println("The area of the pentagon is " + *area*(side) );

}

// define the area while returning a value

**public** **static** **double** area(**double** side1) {

**double** result = 0;

//final double PI = 3.14159;

// subs math formula given

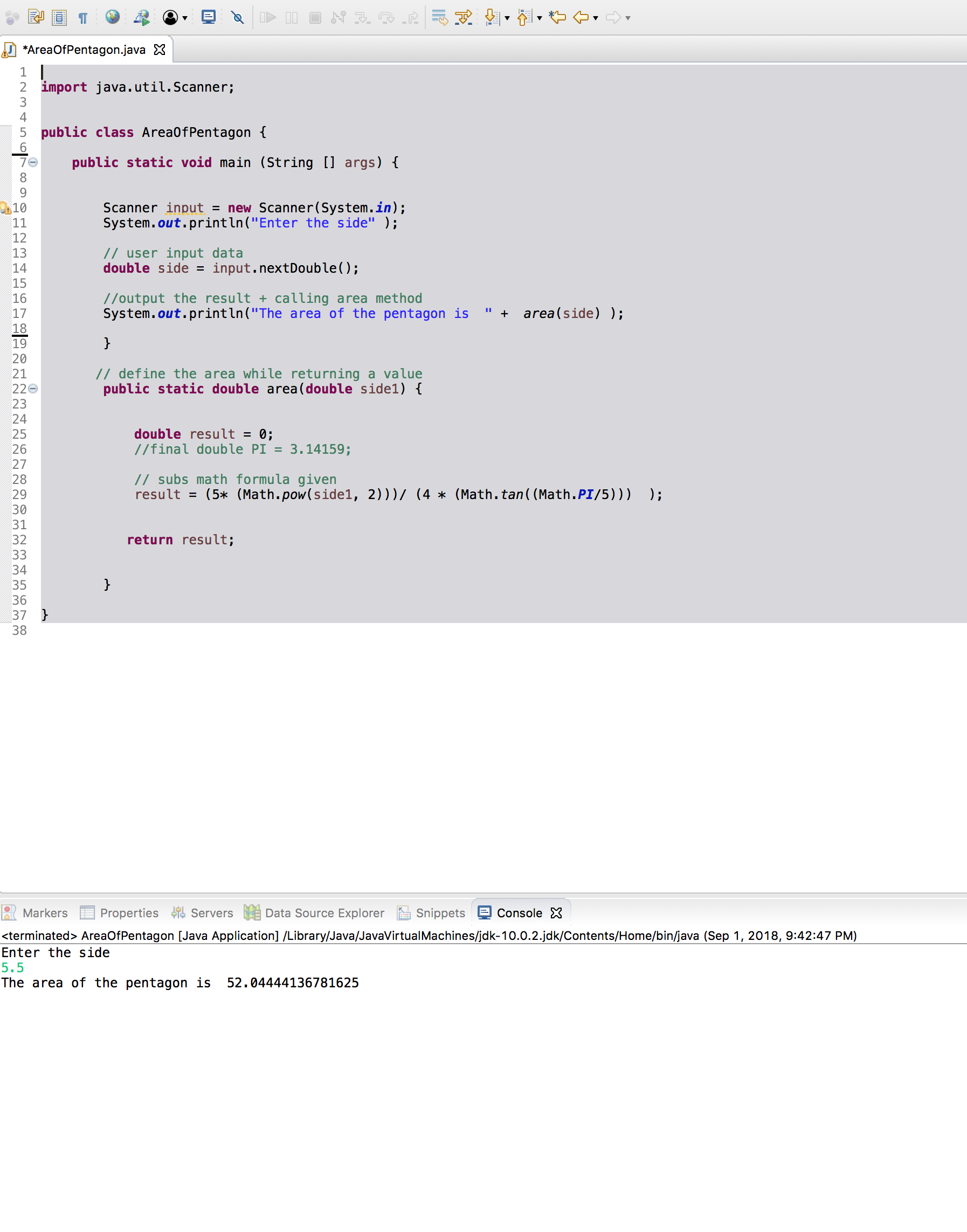
result = (5\* (Math.*pow*(side1, 2)))/ (4 \* (Math.*tan*((Math.***PI***/5))) );

**return** result;

}

}

*Screenshot*

**

# Question 7.31

*Code*

**import** java.util.Scanner;

**public** **class** MergerTwoList {

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

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

//user input for list 1 size and content

System.***out***.println("Enter list1 size and contents");

//intilize the array

**int** n1 = input.nextInt();

**int** [] list1 = **new** **int**[n1];

//input the content into the array

**for** (**int** i = 0; i<n1; i++) {

list1[i] = input.nextInt();

}

//Using bulit in fun to sort

java.util.Arrays.*sort*(list1);

//user input for list 2 size and content

System.***out***.println("Enter list2 size and contents");

//intilize the array

**int** n2 = input.nextInt();

**int** [] list2 = **new** **int**[n2];

//input the content into the array

**for** (**int** i = 0; i<n2; i++) {

list2[i] = input.nextInt();

}

//calling method to sort list2

*SelectionSort*(list2);

//output result for merge list 1 & 2

System.***out***.println("list1 is " + java.util.Arrays.*toString*(list1));

System.***out***.println("list2 is " + java.util.Arrays.*toString*(list2));

//new array for mergelist and calling the method

**int** [] listMerge = *merge*(list1 , list2);

//sort the merge list

*SelectionSort*(listMerge);

//display the result of sorted merge list

System.***out***.println("The merged list is " + java.util.Arrays.*toString*(listMerge));

}

// method for merging two list arrays

**public** **static** **int**[] merge(**int**[] list1, **int** [] list2) {

//intilize merge array with adding length of list1 array and length of list2

**int** n3 = list1.length + list2.length;

**int** [] list3 = **new** **int**[n3];

//intilizing counter for value

**int** count = 0;

//add each value of list1 to list3 & add a counter

**for**(**int** i = 0; i<list1.length; i++) {

list3[i] = list1[i];

count++;

}

//add each value of list2 to list3 where above counter ends from

**for**(**int** j = 0;j<list2.length;j++) {

list3[count++] = list2[j];

}

**return** list3;

}

//based on Book Listing 7.8 and modified the code

// this method for sorting the array

**public** **static** **void** SelectionSort(**int**[] list) {

// go through each element in list

**for** (**int** i = 0; i < list.length - 1; i++) {

// find the minimum value from the list

**int** currentMin = list[i];

**int** currentMinIndex = i;

**for** (**int** j = i + 1 ; j < list.length; j++) {

**if** (currentMin > list[j]) {

currentMin = list[j];

currentMinIndex = j;

}

}

// change the currentMinindex , if not equal

**if**(currentMinIndex !=i) {

list[currentMinIndex] = list[i];

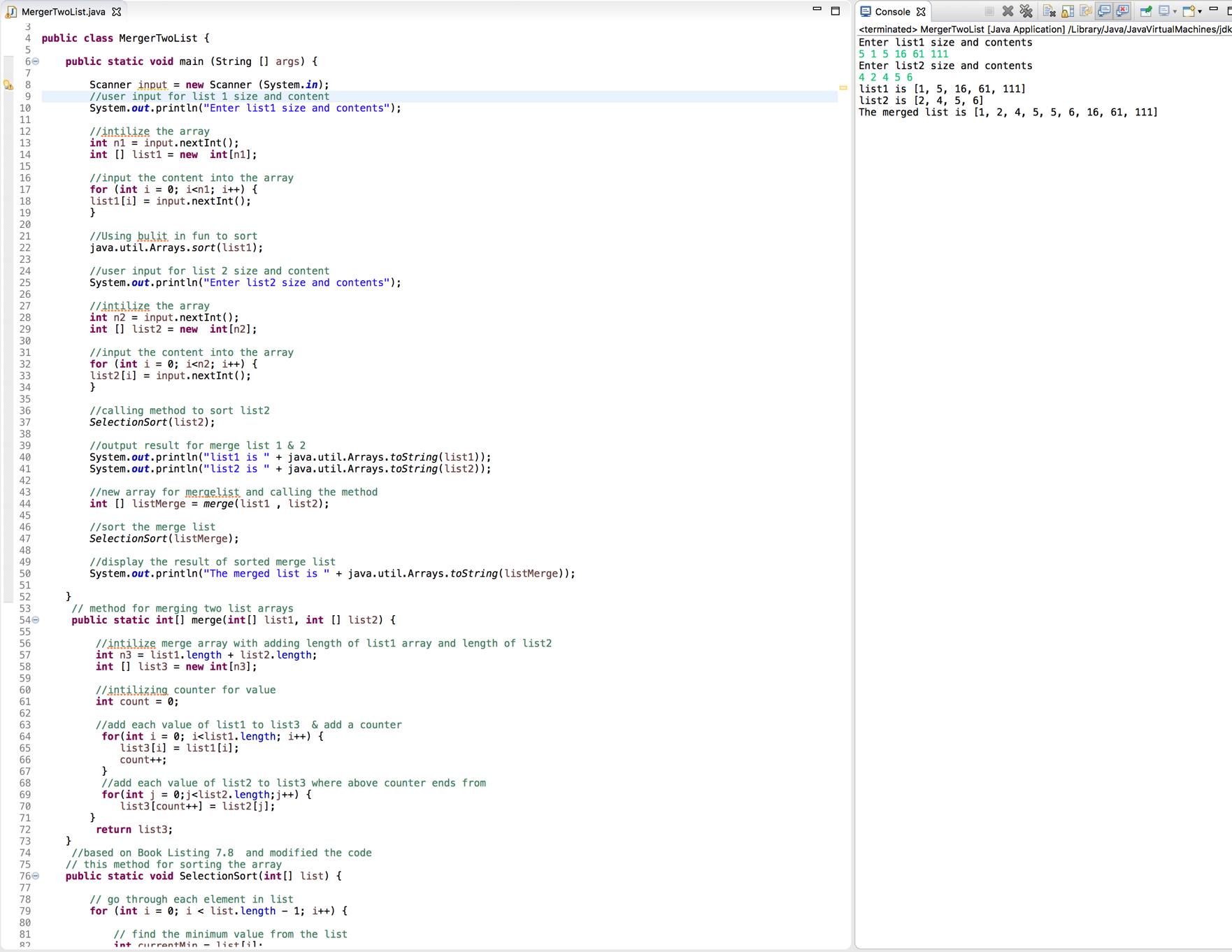
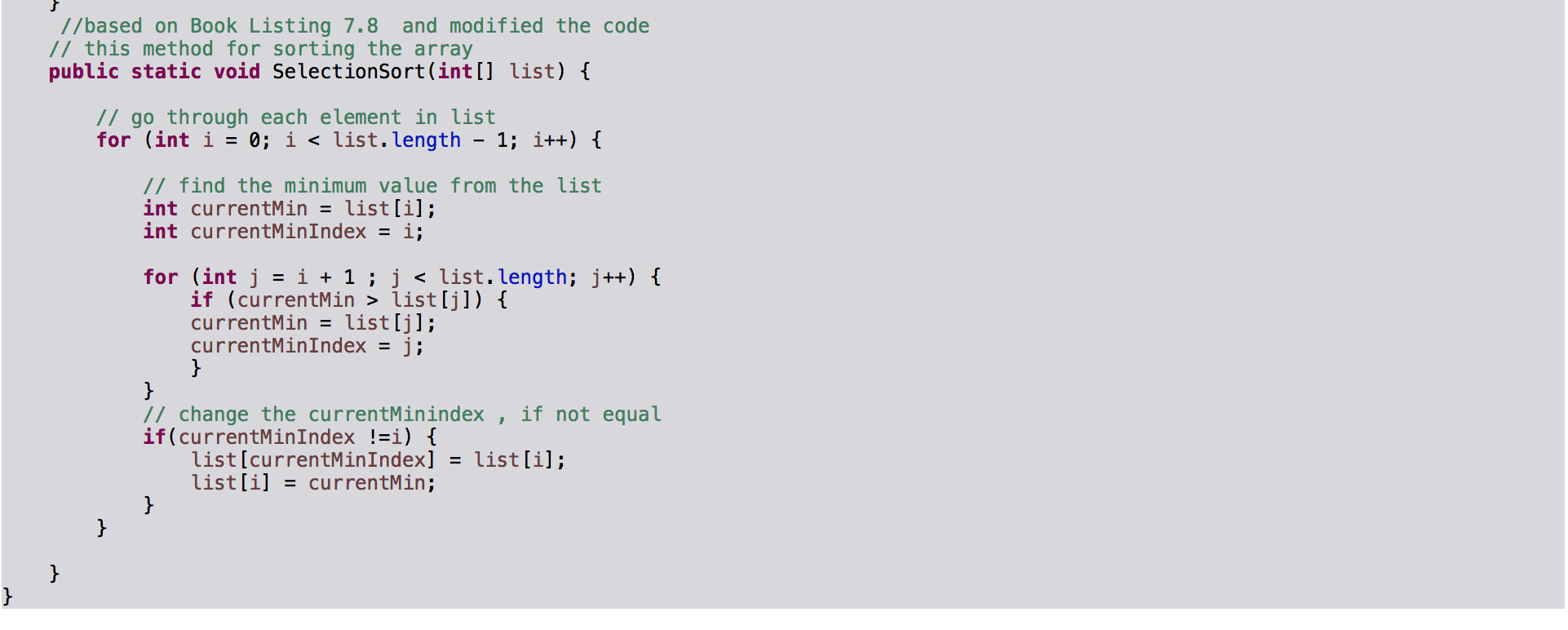
list[i] = currentMin;

}

}

}

}

*Screenshot*