[2CEIT5PE5: MOBILE APPLICATION DEVELOPMENT]

Practical: 1

AIM – To Study Kotlin Programs.

Submitted By: Divy Patel
Enrollment number: 21012011072

Ganpat University
॥ विद्यया समाजोत्कर्षः ॥
U.V. Patel College of Engineering

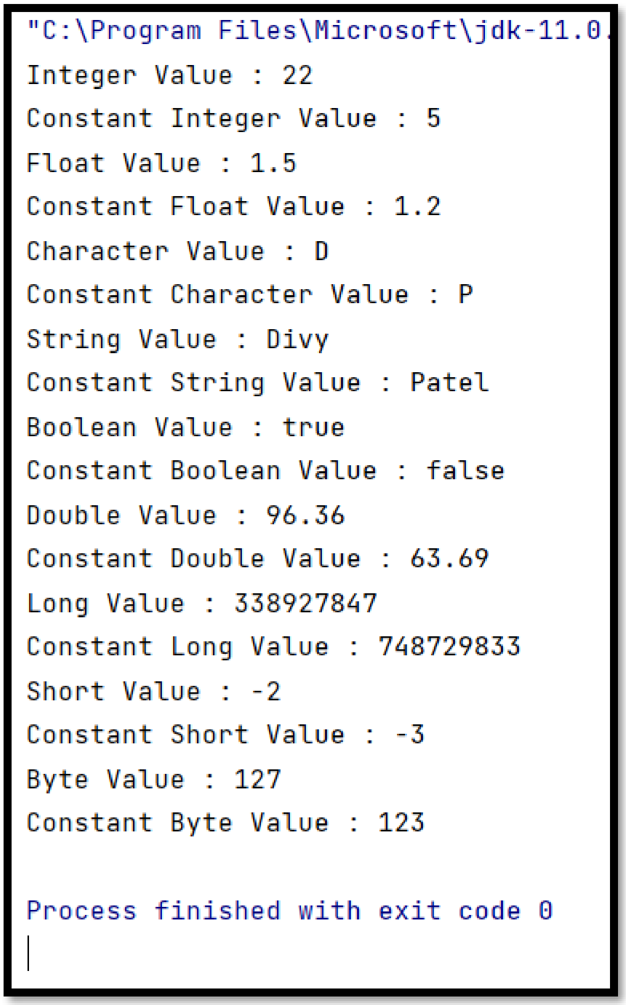
Department of Computer Engineering/Information Technology

1. Store & display values in different variable of different data type.

Code :-

```
fun main() {
    var a : Int = 22
    val a1 : Int = 5
    var b : Float = 1.5f
    val b1 : Float = 1.2f
    var c : Char = 'D'
    val c1 : Char = 'P'
    var d : String = "Divy"
    val d1 : String = "Patel"
    var e : Boolean = true
    val e1 : Boolean = false
    var f : Double = 96.36
    val f1 : Double = 63.69
    var g : Long = 338927847
    val g1 : Long = 748729833
    var h : Short = -2
    val h1 : Short = -3
    var i : Byte = 127
    val i1 : Byte = 123

    println("Integer Value : "+a)
    println("Constant Integer Value : "+a1)
    println("Float Value : "+b)
    println("Constant Float Value : "+b1)
    println("Character Value : "+c)
    println("Constant Character Value : "+c1)
    println("String Value : "+d)
    println("Constant String Value : "+d1)
    println("Boolean Value : "+e)
    println("Constant Boolean Value : "+e1)
    println("Double Value : "+f)
    println("Constant Double Value : "+f1)
    println("Long Value : "+g)
    println("Constant Long Value : "+g1)
    println("Short Value : "+h)
    println("Constant Short Value : "+h1)
    println("Byte Value : "+i)
    println("Constant Byte Value : "+i1)
}
```

Output :-



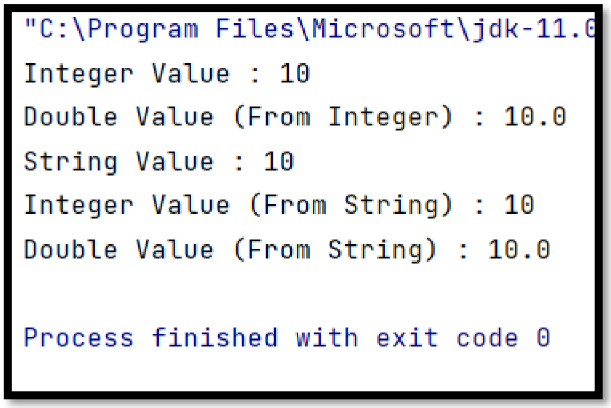
```
"C:\Program Files\Microsoft\jdk-11.0.
Integer Value : 22
Constant Integer Value : 5
Float Value : 1.5
Constant Float Value : 1.2
Character Value : D
Constant Character Value : P
String Value : Divy
Constant String Value : Patel
Boolean Value : true
Constant Boolean Value : false
Double Value : 96.36
Constant Double Value : 63.69
Long Value : 338927847
Constant Long Value : 748729833
Short Value : -2
Constant Short Value : -3
Byte Value : 127
Constant Byte Value : 123

Process finished with exit code 0
```

2. Type Conversion : Integer to Double, String to Integer or Double.

Code :-

```
fun main() {
    var a : Int = 10
    var b = a.toDouble()
    var c : String = 10.toString()
    var d = c.toInt()
    var e = c.toDouble()
    println("Integer Value : "+a)
    println("Double Value (From Integer) : "+b)
    println("String Value : "+c)
    println("Integer Value (From String) : "+d)
    println("Double Value (From String) : "+e)
}
```

Output :-



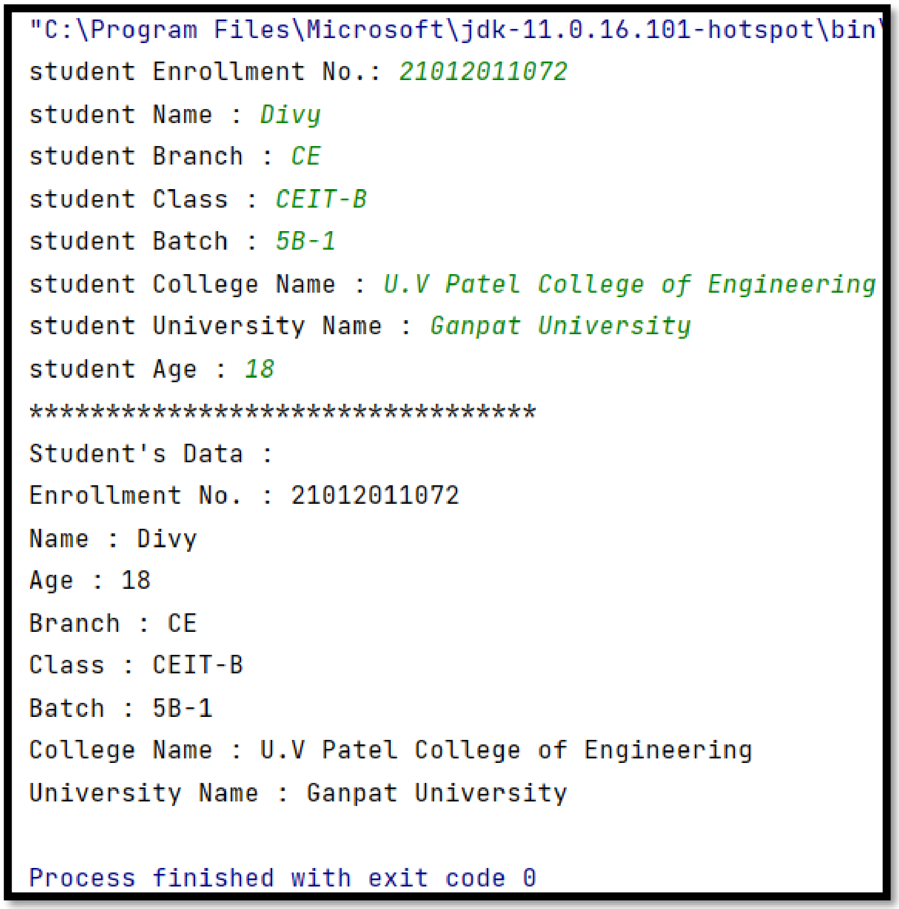
```
"C:\Program Files\Microsoft\jdk-11.0
Integer Value : 10
Double Value (From Integer) : 10.0
String Value : 10
Integer Value (From String) : 10
Double Value (From String) : 10.0

Process finished with exit code 0
```

3. Scan student information & display all the data.

Code :-

```
fun main() {
    print("student Enrollment No.: ")
    var a = readLine()
    print("student Name : ")
    var b = readLine()
    print("student Branch : ")
    var c = readLine()
    print("student Class : ")
    var d = readLine()
    print("student Batch : ")
    var e = readLine()
    print("student College Name : ")
    var f = readLine()
    print("student University Name : ")
    var g = readLine()
    print("student Age : ")
    var h = readLine()

    println("*********************************")

    println("Student's Data : ")
    println("Enrollment No. : "+a)
    println("Name : "+b)
    println("Age : "+h)
    println("Branch : "+c)
    println("Class : "+d)
    println("Batch : "+e)
    println("College Name : "+f)
    println("University Name : "+g)
}
```

Output :-



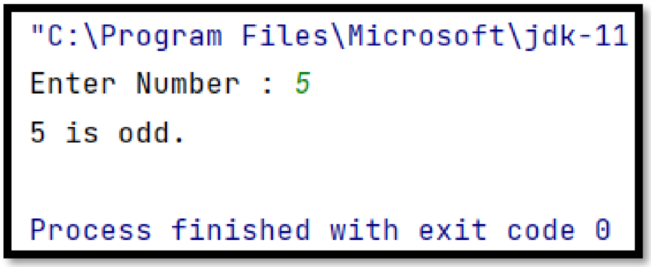
```
"C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot\bin
student Enrollment No.: 21012011072
student Name : Divy
student Branch : CE
student Class : CEIT-B
student Batch : 5B-1
student College Name : U.V Patel College of Engineering
student University Name : Ganpat University
student Age : 18
*********************************
Student's Data :
Enrollment No. : 21012011072
Name : Divy
Age : 18
Branch : CE
Class : CEIT-B
Batch : 5B-1
College Name : U.V Patel College of Engineering
University Name : Ganpat University

Process finished with exit code 0
```

4. Find the Number is odd or even by using Control Folw.

Code :-

```
fun main() {
   println("Enter Number : ")
   val num = readLine()!!.toInt()


   if (num % 2 == 0){
      println("$num is even.")
   }
   else {
      println("$num is odd.")
   }
}
```

Output :-



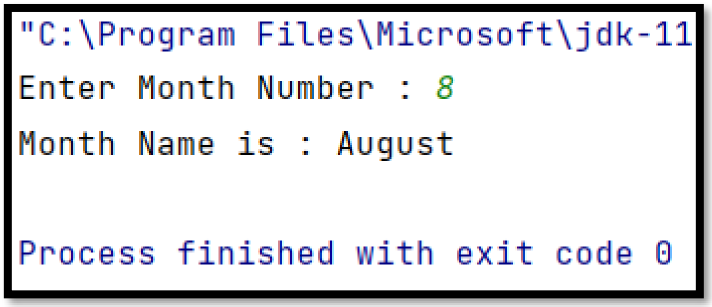
```
"C:\Program Files\Microsoft\jdk-11
Enter Number : 5
5 is odd.

Process finished with exit code 0
```

5. Display month name by entering the month number using When.

Code :-

```
fun main() {
   print("Enter Month Number : ")
   val mn = readLine()!!.toInt()

   when(mn) {
      1 -> println("Month Name is : January")
      2 -> println("Month Name is : February")
      3 -> println("Month Name is : March")
      4 -> println("Month Name is : April")
      5 -> println("Month Name is : May")
      6 -> println("Month Name is : June")
      7 -> println("Month Name is : July")
      8 -> println("Month Name is : August")
      9 -> println("Month Name is : September")
      10 -> println("Month Name is : October")
      11 -> println("Month Name is : November")
      12 -> println("Month Name is : December")
      else -> {
         println("Please enter proper number.")
      }
   }
}
```

Output :-



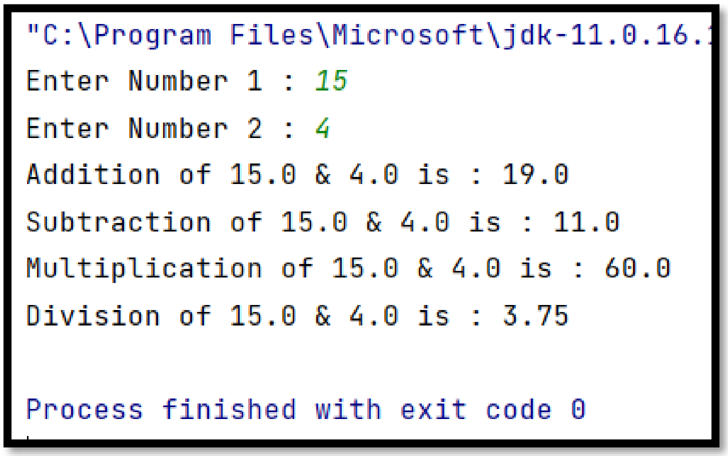
```
"C:\Program Files\Microsoft\jdk-11
Enter Month Number : 8
Month Name is : August

Process finished with exit code 0
```

6. By using a user defined function perform all arithmetic operations.

Code :-

```
fun main() {
   print("Enter Number 1 : ")
   val num1 = readLine()!!.toDouble()
   print("Enter Number 2 : ")
   val num2 = readLine()!!.toDouble()


   println("Addition of $num1 & $num2 is : ${add(num1, num2)}")
   println("Subtraction of $num1 & $num2 is : ${sub(num1, num2)}")
   println("Multiplication of $num1 & $num2 is : ${mul(num1, num2)}")
   println("Division of $num1 & $num2 is : ${div(num1, num2)}")
}

fun add (a: Double, b: Double): Double = a + b
fun sub (a: Double, b: Double): Double = a - b
fun mul (a: Double, b: Double): Double = a * b
fun div (a: Double, b: Double): Double = a / b
```

Output :-



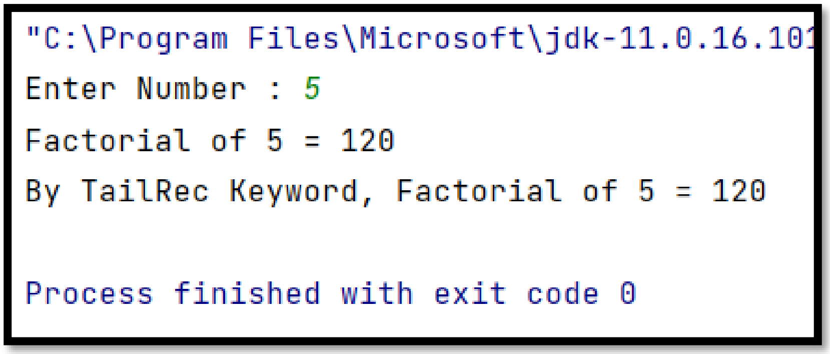
```
"C:\Program Files\Microsoft\jdk-11.0.16.
Enter Number 1 : 15
Enter Number 2 : 4
Addition of 15.0 & 4.0 is : 19.0
Subtraction of 15.0 & 4.0 is : 11.0
Multiplication of 15.0 & 4.0 is : 60.0
Division of 15.0 & 4.0 is : 3.75

Process finished with exit code 0
```

7. Find the factorial number by recursion.

Code :-

```
fun main() {
    print("Enter Number : ")
    var num = readLine()!!.toInt()
    var result : Int
    result = fact(num)
    println("Factorial of $num = $result")
    println("By TailRec Keyword, Factorial of $num = $result")
}

tailrec fun fact(n: Int): Int {
    return if (n == 1){
        n
    }
    else {
        n * fact(n-1)
    }
}
```

Output :-



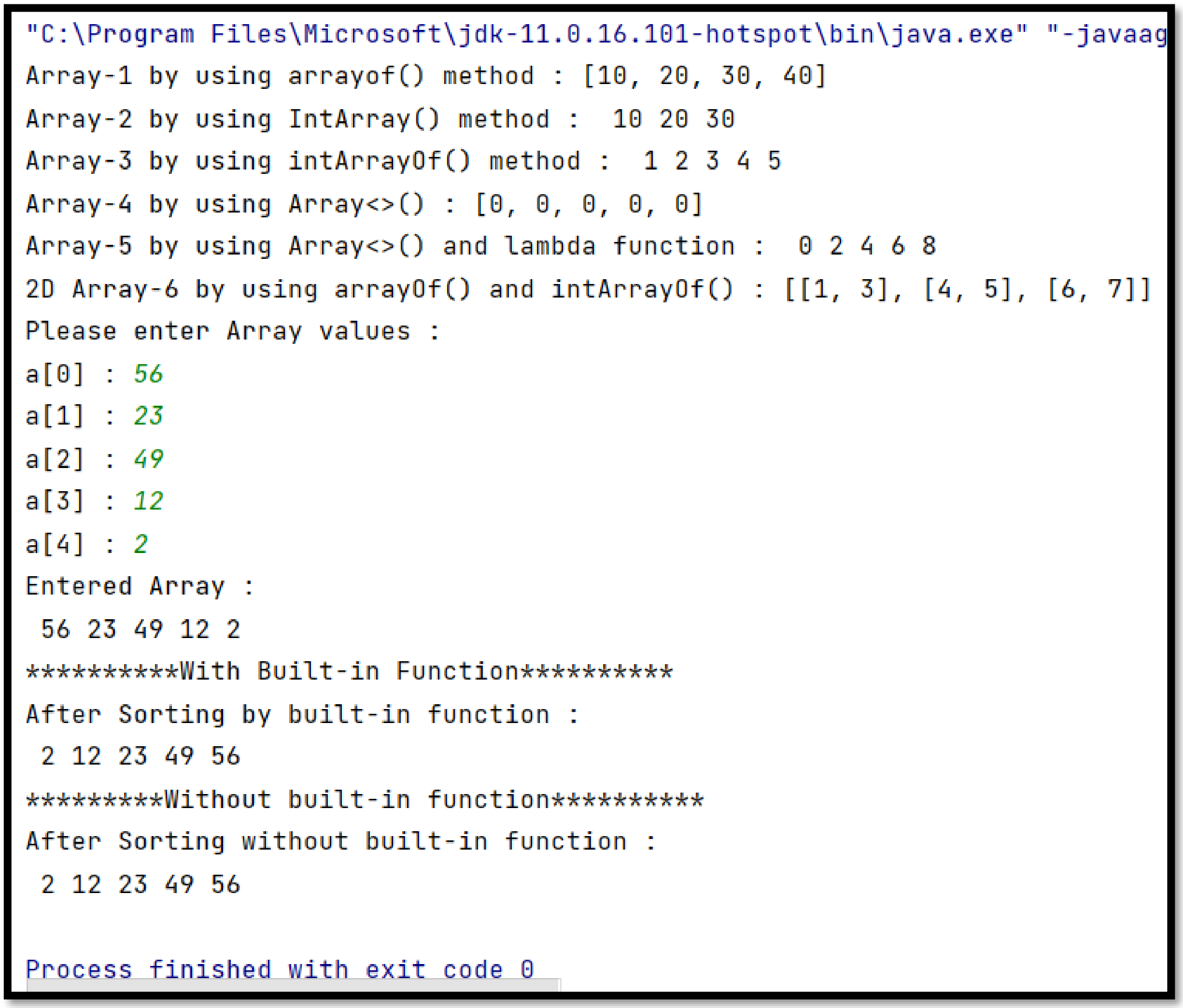
```
"C:\Program Files\Microsoft\jdk-11.0.16.101
Enter Number : 5
Factorial of 5 = 120
By TailRec Keyword, Factorial of 5 = 120

Process finished with exit code 0
```

8. Create different types of array and Sort an array without using inbuilt function & with using inbuilt function.

Code :-

```
import org.w3c.dom.ranges.Range
import java.awt.font.NumericShaper
import java.util.*
import kotlin.collections.ArrayList

fun main() {
    var a = arrayOf(10, 20, 30, 40)
    println("Array-1 by using arrayof() method : "+ Arrays.deepToString(a))

    var a2 = IntArray(size = 3)
    a2[0] = 10
    a2[1] = 20
    a2[2] = 30
    print("Array-2 by using IntArray() method : ")
    for (i in a2) {
        print(" "+i)
    }
    println("")

    var a3 = intArrayOf(1,2,3,4,5)
    print("Array-3 by using intArrayOf() method : ")
    for (j in a3) {
        print(" "+j)
    }
    println("")

    var a4 = Array<Int>(size = 5){0}
    println("Array-4 by using Array<>() : "+Arrays.deepToString(a4))

    val a5 = Array<Int>(size = 5) {index -> index * 2}
    print("Array-5 by using Array<>() and lambda function : ")
    for (i in a5) {
        print(" "+i)
    }
    println("")

    var a6 = arrayOf(
        intArrayOf(1, 3),
        intArrayOf(4, 5),
        intArrayOf(6, 7)
    )
    println("2D Array-6 by using arrayOf() and intArrayOf() : "+Arrays.deepToString(a6))

    var b = IntArray(size = 5)
    println("Please enter Array values : ")
    for (i in 0 until 5) {
        print("a[$i] : ")
        var input = readLine()!!.toInt()
        b[i] = input
    }

    println("Entered Array : ")
    for (elements in b){
        print(" "+elements)
    }
    println("")

    println("**********With Built-in Function**********")
    b.sort()
    println("After Sorting by built-in function : ")
    for (elements in b){
        print(" "+elements)
    }
    println("")

    println("*********Without built-in function*********")
    bubbleSort(b)
    println("After Sorting without built-in function : ")
    for (elements in b){
        print(" "+elements)
    }
    println("")
}

fun bubbleSort(b: IntArray) {
    var n = b.size
    for (i in 0 until n-1) {
        for (j in 0 until n-i-1) {
            if (b[j] > b[j+1]) {
                var temp = b[j]
                b[j] = b[j+1]
                b[j+1] = temp
            }
        }
    }
}
```

Output :-



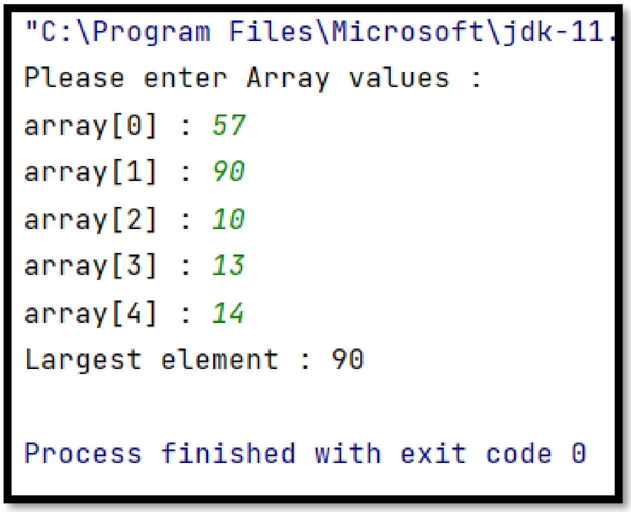
```
"C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot\bin\java.exe" "-javaag
Array-1 by using arrayof() method : [10, 20, 30, 40]
Array-2 by using IntArray() method :  10 20 30
Array-3 by using intArrayOf() method :  1 2 3 4 5
Array-4 by using Array<>() : [0, 0, 0, 0, 0]
Array-5 by using Array<>() and lambda function :  0 2 4 6 8
2D Array-6 by using arrayOf() and intArrayOf() : [[1, 3], [4, 5], [6, 7]]
Please enter Array values :
a[0] : 56
a[1] : 23
a[2] : 49
a[3] : 12
a[4] : 2
Entered Array :
 56 23 49 12 2
**********With Built-in Function**********
After Sorting by built-in function :
 2 12 23 49 56
*********Without built-in function**********
After Sorting without built-in function :
 2 12 23 49 56

Process finished with exit code 0
```

9. Find the maximum number from ArrayList.

Code :-

```
fun main() {
    var array = ArrayList<Int>()
    print("Please enter Array values : ")
    for (i in 0 until 5) {
        print("array[$i] : ")
        var input = readLine()!!.toInt()
        array.add(input)
    }
    var max = Int.MIN_VALUE
    for (num in array) {
        if (num > max) {
            max = num
        }
    }
    println("Largest element : $max")
}
```

Output :-



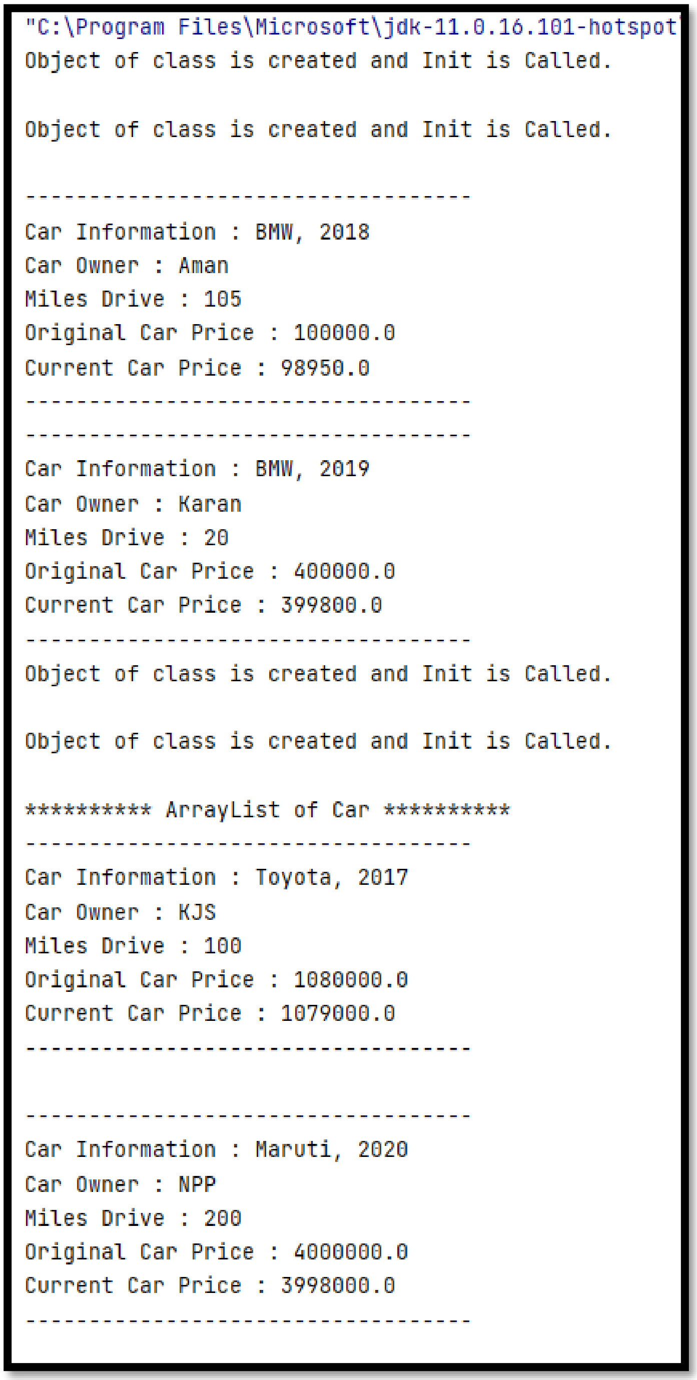
```
"C:\Program Files\Microsoft\jdk-11.
Please enter Array values :
array[0] : 57
array[1] : 90
array[2] : 10
array[3] : 13
array[4] : 14
Largest element : 90


Process finished with exit code 0
```

10. Create a class Car and set various members like type,model,price,owner,milesdrive. Add the function getOriginalCarPrice(),getCurrentPrice,displayCarInfo() in it. Create an object of Car class and access property of it.

Code :-

```
class Car (
    var type : String,
    var model : Int,
    var Oprice : Double,
    var Cprice : Double,
    var owner : String,
    var milesdrive : Int)
{
    init {
        println("Object of class is created and Init is Called.")
        println()
    }
    fun getOriginalCarPrice(): Double {
        return Oprice
    }

    fun getCurrentCarPrice(): Double {
        return Cprice
    }

    fun displayCarInfo() {
        val originalPrice = getOriginalCarPrice()
        val currentPrice = getCurrentCarPrice()
        println("-----------------------------------")
        println("Car Information : $type, $model")
        println("Car Owner : $owner")
        println("Miles Drive : $milesdrive")
        println("Original Car Price : $originalPrice")
        println("Current Car Price : $currentPrice")
        println("-----------------------------------")
    }
}

fun main() {
    var car1 = Car("BMW",2018, 100000.0,98950.0, "Aman",105 )
    var car2 = Car("BMW",2019, 400000.0,399800.0, "Karan",20 )
    car1.displayCarInfo()
    car2.displayCarInfo()

    var cars = ArrayList<Car>()
    cars.add(Car("Toyota",2017, 1080000.0,1079000.0, "KJS",100 ))
    cars.add(Car("Maruti",2020, 4000000.0,3998000.0, "NPP",200 ))
    println("********** ArrayList of Car **********")
    for (car in cars) {
        car.displayCarInfo()
        println()
    }
}
```

Output :-



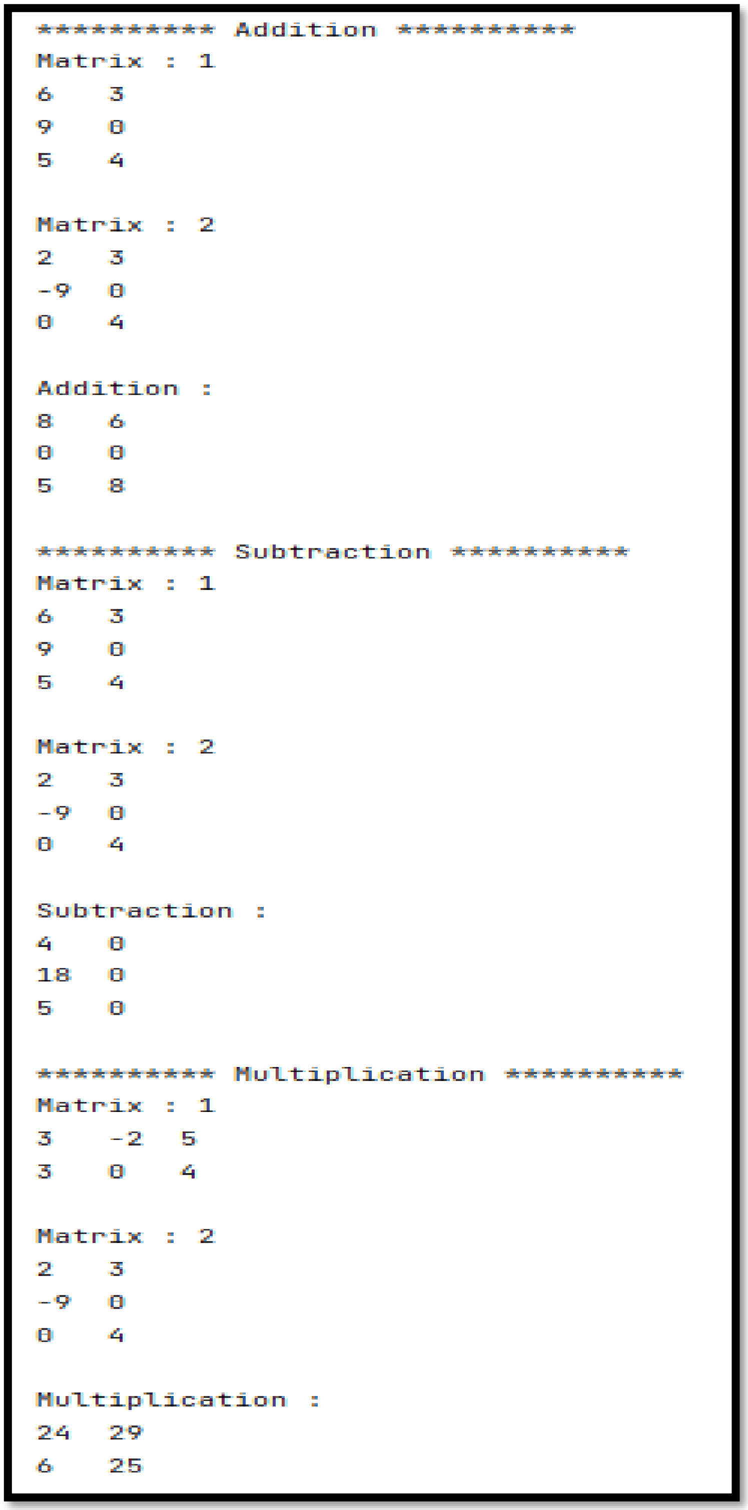
```
"C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot
Object of class is created and Init is Called.

Object of class is created and Init is Called.

-----------------------------------
Car Information : BMW, 2018
Car Owner : Aman
Miles Drive : 105
Original Car Price : 100000.0
Current Car Price : 98950.0
-----------------------------------
-----------------------------------
Car Information : BMW, 2019
Car Owner : Karan
Miles Drive : 20
Original Car Price : 400000.0
Current Car Price : 399800.0
-----------------------------------
Object of class is created and Init is Called.

Object of class is created and Init is Called.

********** ArrayList of Car **********
-----------------------------------
Car Information : Toyota, 2017
Car Owner : KJS
Miles Drive : 100
Original Car Price : 1080000.0
Current Car Price : 1079000.0
-----------------------------------


-----------------------------------
Car Information : Maruti, 2020
Car Owner : NPP
Miles Drive : 200
Original Car Price : 4000000.0
Current Car Price : 3998000.0
-----------------------------------
```

11. Perform Matrix addition, subtraction & multiplication using Class Matrix and operator overloading. Overload toString() function in Matrix class.

Code :-

```
import javax.print.attribute.standard.MediaSize.Other

class Matrix(var data: Array<IntArray>) {
    val rows: Int = data.size
    val columns: Int = data[0].size
    operator fun plus(other: Matrix): Matrix {
        val resultData  = Array(rows) {IntArray(columns)}
        for (i in 0 until rows) {
            for (j in 0 until columns) {
                resultData[i][j] = data[i][j] + other.data[i][j]
            }
        }
        return Matrix(resultData)
    }

    operator fun minus(other: Matrix): Matrix {
        val resultData  = Array(rows) {IntArray(columns)}
        for (i in 0 until rows) {
            for (j in 0 until columns) {
                resultData[i][j] = data[i][j] - other.data[i][j]
            }
        }
        return Matrix(resultData)
    }

    operator fun times(other: Matrix): Matrix {
        val resultData  = Array(rows) {IntArray(other.columns) {0} }
        for (i in 0 until rows) {
            for (j in 0 until other.columns) {
                for (k in 0 until columns)
                resultData[i][j] += data[i][k] * other.data[k][j]
            }
        }
        return Matrix(resultData)
    }

    override fun toString(): String {
        val sb = StringBuilder()
        for (i in 0 until rows) {
            for (j in 0 until columns) {
                sb.append("${data[i][j]}\t")
            }
            sb.append("\n")
        }
        return sb.toString()
    }
}
fun main() {
    val firstMatrix = Matrix(arrayOf(intArrayOf(3, -2, 5), intArrayOf(3, 0, 4)))
    val secondMatrix = Matrix(arrayOf(intArrayOf(2, 3), intArrayOf(-9, 0), intArrayOf(0, 4)))
    val secondMatrix1 = Matrix(arrayOf(intArrayOf(6, 3), intArrayOf(9, 0), intArrayOf(5, 4)))

    println("********** Addition **********")
    println("Matrix : 1 ")
    println(secondMatrix1)
    println("Matrix : 2 ")
    println(secondMatrix)
    val thirdMatrix = secondMatrix1 + secondMatrix
    println("Addition : ")
    println(thirdMatrix)

    println("********** Subtraction **********")
    println("Matrix : 1 ")
    println(secondMatrix1)
    println("Matrix : 2 ")
    println(secondMatrix)
    val subtractMatrix = secondMatrix1 - secondMatrix
    println("Subtraction : ")
    println(subtractMatrix)

    println("********** Multiplication **********")
    println("Matrix : 1 ")
    println(firstMatrix)
    println("Matrix : 2 ")
    println(secondMatrix)
    val multiplication = firstMatrix * secondMatrix
    println("Multiplication : ")
    println(multiplication)
}
```

Output :-



```
********** Addition **********
Matrix : 1
6   3
9   0
5   4

Matrix : 2
2   3
-9  0
0   4

Addition :
8   6
0   0
5   8

********** Subtraction **********
Matrix : 1
6   3
9   0
5   4

Matrix : 2
2   3
-9  0
0   4

Subtraction :
4   0
18  0
5   0

********** Multiplication **********
Matrix : 1
3   -2  5
3   0   4

Matrix : 2
2   3
-9  0
0   4

Multiplication :
24  29
6   25
```