[ 2CEIT5PE5: MOBILE APPLICATION DEVELOPMENT]

# Practical: 1

**AIM-** Instructions to Start working in IntelliJ IDEA in LAB PC

Submitted By: Pavan Patel
Enrollment number:21012021075

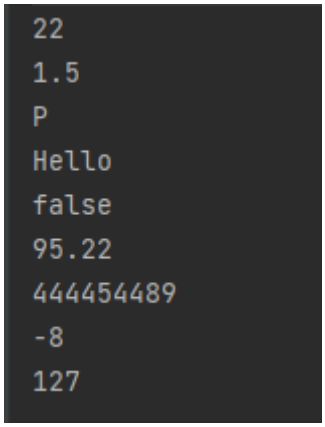

Department of Computer
Engineering/Information Technology

1.Store & Display values in different variable of different type (Integer, Double, Float, Long, Short, Byte, Char, Boolean, String)

- **CODE:**

```
fun main(){
var n1:Int=22
var n2:Float=1.5f
var n3:Char='P'
var n4:String="Hello"
var n5:Boolean=false
var n6:Double=95.22
var n7:Long=444454489
var n8:Short=-8
var n9:Byte=127
println(n1);
println(n2);
println(n3);
println(n4);
println(n5);
println(n6);
println(n7);
println(n8);
println(n9);
}
```

- **OUTPUT:**




2.Type conversion:

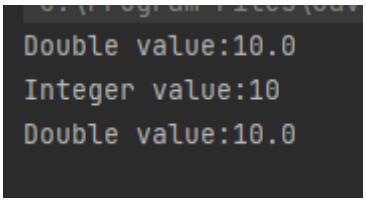
Integer to Double, String to Integer, String to Double.

- **CODE:**

```
fun main()
{
var a1=10
var a2=a1.toDouble()
println("Double value:"+a2)
```

```
var a3="10"
var a4=a3.toInt()
println("Integer value:"+a4)
var a5=a3.toDouble()
println("Double value:"+a5)
}
```

- **OUTPUT:**




3.Scan student’s information and display all the data.

- **CODE:**

```
fun main()
{
println("Enter student's Enroll:")
var enroll= readLine()
println("Enter students name:")
var name= readLine()
println("Enter Students Branch:")
var branch= readLine()
println("Enter students class:")
var Class = readLine()
println("Enter students Batch:")
var Batch= readLine()
println("Enter students college:")
var C= readLine()
println("Enter students University name:")
var n= readLine()
println("Enter Students Age:")
var age= readLine()
println("Enter student's Enroll:$enroll")
println("Enter students name:$name")
println("Enter Students Branch:$branch")
println("Enter students class:$Class")
println("Enter students Batch:$Batch")
println("Enter students college:$C")
println("Enter students University name:$n")
println("Enter Students Age:$age")
}
```

- **OUTPUT:**

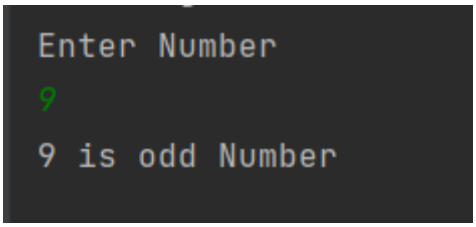
```
Enter student's Enroll:
21012021075
Enter students name:
Pavan Patel
Enter Students Branch:
IT
Enter students class:
CEIT-B
Enter students Batch:
5B6
Enter students college:
U.V.Patel College Of Engineering
Enter students University name:
Ganpat University
Enter Students Age:
20
Enter student's Enroll:21012021075
Enter students name:Pavan Patel
Enter Students Branch:IT
Enter students class:CEIT-B
Enter students Batch:5B6
Enter students college:U.V.Patel College Of Engineering
Enter students University name:Ganpat University
Enter Students Age:20

Process finished with exit code 0
```

4.Find the number is odd or even by using Control Flow inside println() method.

- **CODE:**

```
fun main() {
println("Enter Number ")
var i = readLine()!!.toInt()
if( i % 2 ==0 )
println("$i is even Number")
else
println("$i is odd Number")
}
```

- **OUTPUT:**

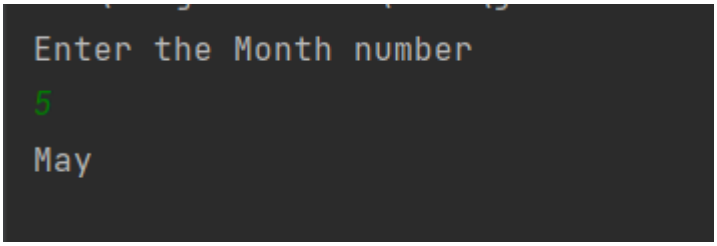


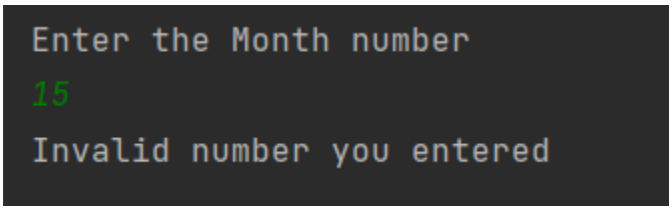

5.Display month name using When

- **CODE:**

```
fun main()
{
println("Enter the Month number")
var month = readLine()
when (month)
{
"1" -> println("January")
"2" -> println("February")
"3" -> println("March")
"4" -> println("April")
"5" -> println("May")
"6" -> println("June")
"7" -> println("July")
"8" -> println("August")
"9" -> println("September")
"10" -> println("October")
"11" -> println("November")
"12" -> println("December")
else -> println("Invalid number you entered")
}
}
```

- **OUTPUT:**



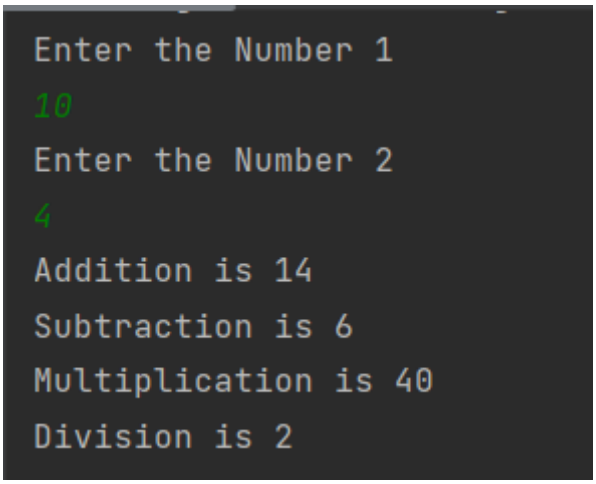

6. By using a user defined function perform all arithmetic operations.

- **CODE:**

```
import java.util.Scanner fun main(args: Array<String>)
{
val reader = Scanner(System.`in`)
println("Enter the Number 1")
val num1:Int = reader.nextInt()
val read = Scanner(System.`in`)
println("Enter the Number 2")
val num2:Int = read.nextInt()
val add = addition(num1,num2)
val sub = subtraction(num1,num2)
val mul = multiplication(num1,num2)
val div = division(num1,num2)
println("Addition is $add")
println("Subtraction is $sub")
println("Multiplication is $mul")
println("Division is $div")
}
fun addition(num1: Int,num2: Int): Int {
return num1 + num2
}
fun subtraction(num1: Int,num2: Int): Int
{
return num1 - num2
}
fun multiplication(num1: Int,num2: Int): Int
{
return num1 * num2
}
fun division(num1: Int,num2: Int): Int
{
```

```
return num1 / num2
}
```

- **OUTPUT:**

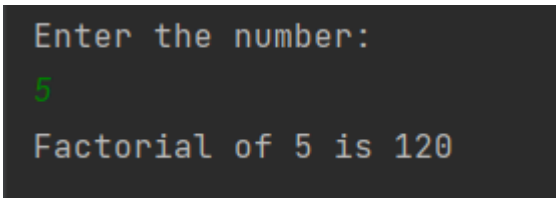



7.Find the factorial of number by recursion. Explain "tailrec" keyword.

- **CODE:**

```
tailrec fun fact(n:Int):Int
{
if(n==1)
{
return 1
}
return n*fact(n-1)
}
fun main()
{
println("Enter the number: ")
var num:Int = readLine()!!.toInt()
println("Factorial of $num is ${fact(num)}") }
```

- **OUTPUT:**



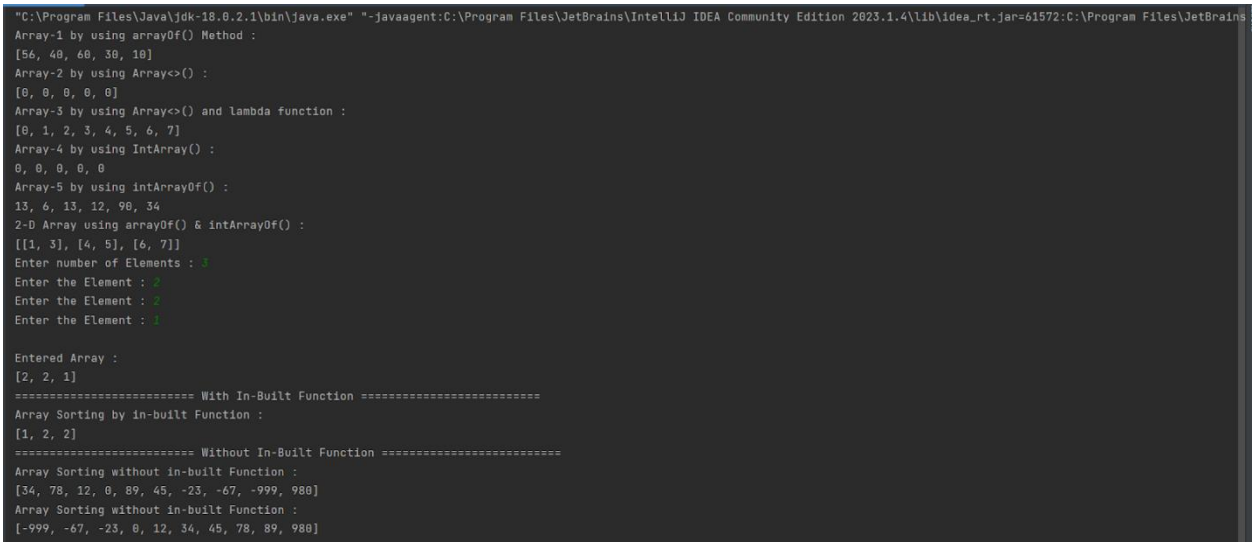

8.Create different types of Array as shown in image. Explore Arrays.deepToString(), contentDeepToString() methods, IntArray variable .joinToString() and use in program to print Array. Explore range, downTo, until etc. for loop and use in this program. Sort Array of Integer data type without using inbuilt function & with using inbuilt function

- **CODE:**

```
fun main() {
println("Array-1 by using arrayOf() Method : ")
val a1 = arrayOf(56, 40, 60, 30, 10)
println(a1.contentToString())
```

```
    println("Array-2 by using Array<>() : ")
    val a2 = Array(5) { 0 }
    println(a2.contentToString())
    println("Array-3 by using Array<>() and lambda function : ")
    val a3 = Array(8) { i -> i }
    println(a3.contentToString())
    println("Array-4 by using IntArray() : ")
    val a4 = IntArray(5)
    println(a4.joinToString(", "))
    println("Array-5 by using intArrayOf() : ")
    val a5 = intArrayOf(13, 6, 13, 12, 90, 34)
    println(a5.joinToString(", "))
    println("2-D Array using arrayOf() & intArrayOf() : ")
    val a6 = arrayOf(intArrayOf(1, 3), intArrayOf(4, 5), intArrayOf(6, 7))
    println(a6.contentDeepToString())
    print("Enter number of Elements : ")
    val size: Int = readLine()!!.toInt()
    val a7 = IntArray(size) { 0 }
    for (i in 0 until size) {
    print("Enter the Element : ")
    a7[i] = readLine()!!.toInt()
    }
    println("\nEntered Array : ")
    println(a7.contentToString())
    println("========================== With In-Built Function
==========================")
    println("Array Sorting by in-built Function : ")
    a7.sort()
    println(a7.contentToString())
    val a8 = intArrayOf(34, 78, 12, 0, 89, 45, -23, -67, -999, 980)
    println("========================== Without In-Built Function
==========================")
    println("Array Sorting without in-built Function : ")
    println(a8.contentToString())
    var temp: Int
    for (i in a8.indices) {
    for (j in a8.indices) {
    if (a8[j] > a8[i]) {
    temp = a8[j]
    a8[j] = a8[i]
    a8[i] = temp
    }
    } }
    println("Array Sorting without in-built Function : ")
    println(a8.contentToString()) }
```

- **OUTPUT:**

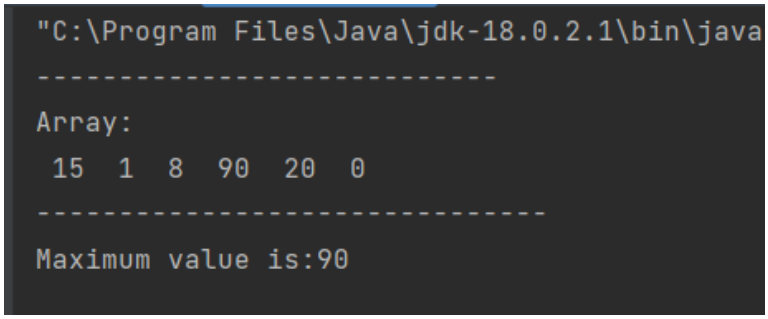



9.Find the max number from ArrayList.

- **CODE:**

```
import kotlin.math.max
fun main(args:Array<String>) {
println("----------------------------")
var ary: IntArray = intArrayOf(15, 1, 8, 90, 20, 0)
var max_val = ary[0]
for (i in ary) {
if (i > max_val) {
max_val = i
}
}
println("Array:")
for (i in ary) {
print(" " + i + " ")
}
println("\n-----------------------------")
println("Maximum value is:" + max_val)
}
```

- **OUTPUT:**



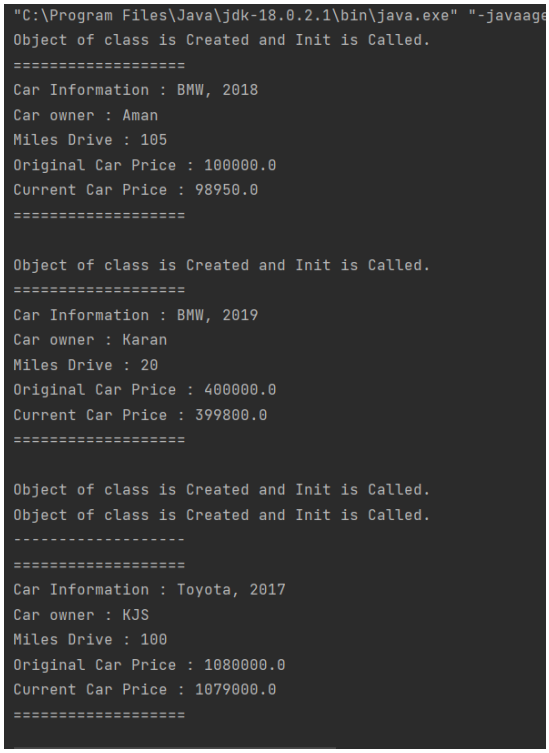
10. Write Different types of Class & Constructor. Create a class Car and set various members like type, model, price, owner, milesDrive. add the function getCarPrice in it. Create an object of Car class and access property of it. (getCarInformation(), getOriginalCarPrice(), getCurrentCarPrice(), displayCarInfo() etc.)

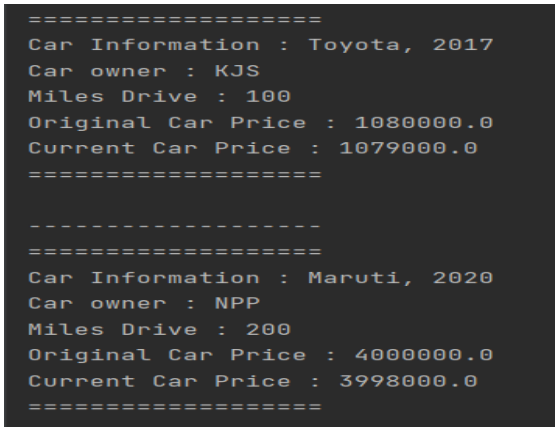
- **CODE:**

```
fun main() {
val car1 = Car("BMW, 2018", "Aman", 105, 100000.0, 98950.0)
car1.getCarFullDetails()

val car2 = Car("BMW, 2019", "Karan", 20, 400000.0, 399800.0)
car2.getCarFullDetails()

val Cars = ArrayList<Car> (2)
val car3 = Car("Toyota, 2017","KJS",100,1080000.0,1079000.0)
val car4 = Car("Maruti, 2020","NPP",200,4000000.0,3998000.0)
Cars.add(car3)
Cars.add(car4)

for (i in Cars){
println("--------------------")
i.getCarFullDetails()
}
}
class Car(private val model: String, private val owner: String, private val miles: Int, private val original: Double, private val current: Double) {
init {
println("Object of class is Created and Init is Called.")
}

private fun info(): String {
return model
}

private fun carowner(): String{
return owner
}

private fun milesDrive(): Int {
return miles
}

private fun orgprice(): Double {
return original
}

private fun currprice(): Double {
return current
```

```
}

fun getCarFullDetails() {
println("====================")
println("Car Information : ${info()}")
println("Car owner : ${carowner()}")
println("Miles Drive : ${milesDrive()}")
println("Original Car Price : ${orgprice()}")
println("Current Car Price : ${currprice()}")
println("====================\n")
}
}
```

- **OUTPUT:**



11. Write about Operator Overloading. Perform Matrix Addition, Subtraction & Multiplication using Class Matrix & operator overloading. Overload toString() function in Matrix class.

- **CODE:**

```
import kotlin.math.min
class Matrix(var data:Array<IntArray>, var rows:Int, var cols:Int)
{
operator fun plus(other: Matrix): Array<IntArray> {

val resultData = Array(rows) {IntArray(cols) }
for (i in 0 until rows) {
for (j in 0 until cols) {
resultData[i][j] = this.data[i][j] + other.data[i][j]
}
}
return resultData
}
operator fun minus(other: Matrix): Array<IntArray> {

val resultData = Array(rows) {IntArray(cols) }
for (i in 0 until rows) {
for (j in 0 until cols) {
resultData[i][j] = this.data[i][j] - other.data[i][j]
}
}
return resultData
}

operator fun times(other: Matrix): Array<IntArray> {

val resultData = Array(rows) {IntArray(cols) }
for (i in 0 until rows) {
for (j in 0 until other.cols) {
for (k in 0 until cols) {
resultData[i][j] += this.data[i][k] * other.data[k][j]
}
}
}
return resultData
}
}
fun print1(array:Array<IntArray>,row: Int,col: Int) {
println(" ( $row x $col Matrix):")
for (i in 0 until row)
{
for(j in 0 until col)
{
```

```
print("${array[i][j]}\t")
}
println()
}
}
fun main()
{
val a1=arrayOf(intArrayOf(3,-2,5), intArrayOf(3,0,4))
var a2=arrayOf(intArrayOf(2,3), intArrayOf(-9,0), intArrayOf(0,4))
var a3=arrayOf(intArrayOf(6,3), intArrayOf(9,0), intArrayOf(5,4))
var firstmatrix=Matrix(a1,2,3)
var secondmatrix=Matrix(a2,3,2)
var secondmatrix1=Matrix(a3,3,2)
println("**********Addition**********")
print("Matirx:1")
print1(a2,3,2)
print("Matrix:2")
print1(a3,3,2)
print("Addition:")
var plusmatrix=secondmatrix+secondmatrix1
print1(plusmatrix,3,2)

println("**********Subtraction**********")
print("Matirx:1")
print1(a3,3,2)
print("Matrix:2")
print1(a2,3,2)
print("Subtraction:")
var minusmatrix=secondmatrix1-secondmatrix
print1(minusmatrix,3,2)

println("**********Subtraction**********")
print("Matirx:1")
print1(a1,2,3)
print("Matrix:2")
print1(a2,3,2)
print("Multiplication:")
var multimatrix=firstmatrix*secondmatrix
print1(multimatrix,2,2)
}
```

- **OUTPUT:**

```
*****************Addition****************
Matirx:1 ( 3 x 2 Matrix):
2   3
-9  0
0   4
Matrix:2 ( 3 x 2 Matrix):
6   3
9   0
5   4
Addition: ( 3 x 2 Matrix):
8   6
0   0
5   8
```

```
*****************Subtraction****************
Matirx:1 ( 3 x 2 Matrix):
6   3
9   0
5   4
Matrix:2 ( 3 x 2 Matrix):
2   3
-9  0
0   4
Subtraction: ( 3 x 2 Matrix):
4   0
18  0
5   0
*****************Subtraction****************
Matirx:1 ( 2 x 3 Matrix):
3   -2  5
3   0   4
Matrix:2 ( 3 x 2 Matrix):
2   3
-9  0
0   4
Multiplication: ( 2 x 2 Matrix):
24  29
6   25
```

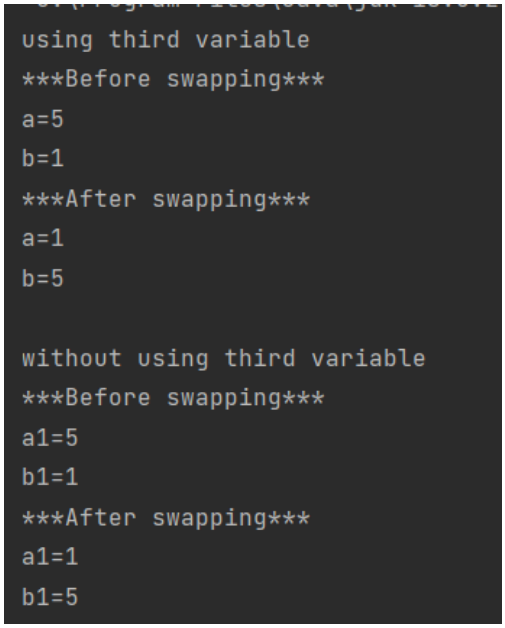
12.

- **CODE:**

```
fun main(){
   println("using third variable")
   var a=5
   var b=1
   println("***Before swapping***")
   println("a=$a")
```

```
        println("b=$b")
        println("***After swapping***")
        var c=a
        a=b
        b=c
        println("a=$a")
        println("b=$b")
        println("")
        println("without using third variable")
        var a1=5
        var b1=1
        println("***Before swapping***")
        println("a1=$a1")
        println("b1=$b1")
        println("***After swapping***")
        a1=a1-b1 //5-1=4
        b1=a1+b1 //4+1=5
        a1=b1-a1 //5-4=1
        println("a1=$a1")
        println("b1=$b1")
    }
```

- **OUTPUT:**



```
using third variable
***Before swapping***
a=5
b=1
***After swapping***
a=1
b=5

without using third variable
***Before swapping***
a1=5
b1=1
***After swapping***
a1=1
b1=5
```