[2CEIT5PE5: MOBILE APPLICATION DEVELOPMENT]

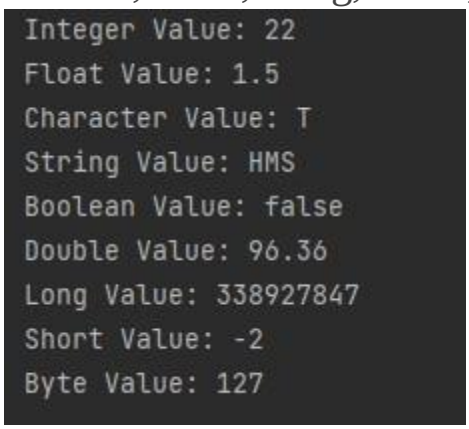
Practical:1

AIM:KOTLIN PROGRAMS

Submitted By:
Enrollment number: 20012011035

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()
{
println("20012011035_ISHWAR DESAI")
var i = 35
var s = "Ishwar"
var c = 'I'
var d = 12.4
var f = 32.0f
var by = 137
var bool = true
var l = 20012011035
println("Integer: $i")
println("String: $s")
println("Character: $c")
println("Double: $d")
println("Float: $f")
println("Byte: $by")
println("Boolean: $bool")
println("Long: $l")
}
```

Output:

```
20012011035_ISHWAR DESAI
Integer: 35
String: Ishwar
Character: I
Double: 12.4
Float: 32.0
Byte: 137
Boolean: true
Long: 20012011035
```

2.Type conversion:

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

```
Integer Value:10
Double Value (From Integer):10.0
String Value:10
Integer Value1 (From String):10
Integer Value2 (From String):10
Double Value (From String):11.12
```

Code:

```
fun main()
{
    println("20012011035_ISHWAR DESAI")
    //Integer  to Double
    var i = 10
    var db = i.toDouble()
    println("Integer $i to Double $db")
    //String to Integer
    var s = "10"
    var i2 = s.toInt()
    println("String $s to Integer $i2")

    //String to Double
    var s2 = "11"
    var d = s2.toDouble()
    println("String $s2 to Double $d")
}
```

Output:

```
20012011035_ISHWAR DESAI
Integer 10 to Double 10.0
String 10 to Integer 10
String 11 to Double 11.0
```

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

```
student Enrollment No.:19012011011
student Name:XYZ
student Branch:CE
student Class:CEIT-A
student Batch:AB1
student College Name:U V Patel College of Engineering
student University Name:Ganpat University
student Age:20

********************

Student's Data:
Enrollment No.:19012011011
Name:XYZ
Age:20
Branch:CE
Class:CEIT-A
Batch:AB1
College Name:U V Patel College of Engineering
University Name:Ganpat University
```

Code:

```
fun main()
{
    println("Enter Student's Enroll:")
    var enroll= readLine()
    println("Enter Student's Name:")
    var name= readLine()
    println("Enter Student's Branch:")
    var branch= readLine()
    println("Enter Student's Class:")
```

```
    var c= readLine()
    println("Enter Batch:")
    var  batch= readLine()
    println("Enter Student's Collage Name:")
    var  n = readLine()
    println("Enter Student's University Name:")
    var u= readLine()
    println("Enter Student's Age:")
    var age= readLine()

    println(" Student's Enroll:$enroll")
    println(" Student's Name:$name")
    println("Enter Student's Branch:$branch")
    println("Enter Batch:$batch")
    println("Enter Student's Collage Name:$n")
    println("Enter Student's University Name:$u")
    println("Enter Student's Age:$age")
}
```

Output:

```
Enter Student's Enroll:
20012011035
Enter Student's Name:
ISHWAR
Enter Student's Branch:
CE
Enter Student's Class:
C
Enter Batch:
AB11
Enter Student's Collage Name:
U.V.PATEL COLLAGE OF ENGINEERING
Enter Student's University Name:
GANPAT
Enter Student's Age:
19
 Student's Enroll:20012011035
 Student's Name:ISHWAR
Enter Student's Branch:CE
Enter Batch:AB11
Enter Student's Collage Name:U.V.PATEL COLLAGE OF ENGINEERING
Enter Student's University Name:GANPAT
Enter Student's Age:19
```

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

```
Enter Number:5
5 is odd
```

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:

```
Enter Number
5
5 is odd Number
```

5.Display month name using When

```
Enter Month Number:1
January

Enter Month Number:13
please enter proper number
```

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:

```
Enter the Month number
3
March
```

```
Enter the Month number
13
Invalid number you entered
```

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

```
Addition of 111, 2222, -222 is 2111
Subtraction of 111, 2222, -222 is -1889
Multiplication of 111, 2222, -222 is -54754524
Division of 2222, 111 is 20
```

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:

```
Enter the Number 1
10
Enter the Number 2
5
Addition is 15
Subtraction is 5
Multiplication is 50
Division is 2
```

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

```
Enter Number:5
Factorial of 5 = 120
By TailRec Keyword, Factorial of 5 = 120
```

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:

```
Enter the number:
5
Factorial of 5 is 120
```

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

```
Create Array-1 by using arrayOf() method:
[10, 90, 60, 80, 100]
Create Array-2 by using Array<>():
[0, 0, 0, 0, 0]
Create Array-3 by using Array<>() and lambda function:
[0, 1, 2, 3, 4, 5, 6, 7]
Create Array-4 by using IntArray():
0, 0, 0, 0, 0

Create Array-5 by using intArrayOf():
12, 10, 1, 5, 18, 19

Create 2D Array-6 by using arrayOf() and intArrayOf():
[[1, 3], [4, 5], [6, 7]]
Please enter Array Value:
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***************
Before Sorting:
56, 23, 49, 12, 2

After Sorting without built-in function:
2, 12, 23, 49, 56
```

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:

```
Array-1 by using arrayOf() Method :
[56, 40, 60, 30, 10]
Array-2 by using Array<>() :
[0, 0, 0, 0, 0]
Array-3 by using Array<>() and lambda function :
[0, 1, 2, 3, 4, 5, 6, 7]
Array-4 by using IntArray() :
0, 0, 0, 0, 0
Array-5 by using intArrayOf() :
13, 6, 13, 12, 90, 34
2-D Array using arrayOf() & intArrayOf() :
[[1, 3], [4, 5], [6, 7]]
Enter number of Elements : 2
Enter the Element : 1
Enter the Element : 4

Entered Array :
[1, 4]
========================= With In-Built Function ========================
Array Sorting by in-built Function :
[1, 4]
========================= Without In-Built Function =====================
Array Sorting without in-built Function :
[34, 78, 12, 0, 89, 45, -23, -67, -999, 980]
Array Sorting without in-built Function :
[-999, -67, -23, 0, 12, 34, 45, 78, 89, 980]
```

9.Find the max number from ArrayList.

```
a[0]=57
a[1]=90
a[2]=10
a[3]=13
a[4]=14
Largest element =90
```

Code:

```
fun main(args:Array<String>){
println("--------------------------------")
var ary:IntArray = intArrayOf(15, 7, 12, 27, 11, 4)
var max_num = ary[0]
```

```
    for (i in ary){
if (i > max_num){
max_num = i
        }
}
    println("Array:")
for(i in ary){
print(""+ i +" ")
    }
    println("\n---------------------------------
")     println("Maximum Number: "+max_num) }
```

Output:

```
---------------------------------
Array:
15 7 12 27 11 4
---------------------------------
Maximum Number: 27
```

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

```
Creating Car Class Object car1 in next line
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
----------

Creating Car Class Object car2 in next line
Object of class is created and Init is called.
----------
Car Information: BMW, 2019
Car Owner: Karan
Miles Drive: 20
Original Car Price: 400000.0
Current Car Price: 399800.0
----------

******* ArrayList of Car *************
Object of class is created and Init is called.
Object of class is created and Init is called.
----------
----------
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
----------
```

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:

```
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
==================

Object of class is Created and Init is Called.
==================
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.
------------------
==================
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. Write about Operator Overloading. Perform Matrix Addition, Subtraction & Multiplication using Class & operator overloading. Overload toString() function in Matrix class.

```kotlin
fun main() {
    val firstMatrix = Matrix(arrayOf(intArrayOf(3, -2, 5), intArrayOf(3, 0, 4)), noOfRow: 2, noOfCol: 3)
    val secondMatrix = Matrix(arrayOf(intArrayOf(2, 3), intArrayOf(-9, 0), intArrayOf(0, 4)), noOfRow: 3, noOfCol: 2
    val secondMatrix1 = Matrix(arrayOf(intArrayOf(6, 3), intArrayOf(9, 0), intArrayOf(5, 4)), noOfRow: 3, noOfCol: 2

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


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

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

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

***************Subtraction************
Matrix: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

***************Multiplication************
Matrix: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
```

Code:

```
class Matrix(private val matrix: Array<IntArray>, private val rows: Int,
private val cols: Int) {

    override fun toString(): String {
var res = "($rows x $cols Matrix): \n"
for (i in matrix) {             for (j in i)
{                   res += "$j\t"
            }
            res += "\n"
        }
        return res
    }
    operator fun plus(obj: Matrix): Matrix {
        val sum = Array(this.rows) { IntArray(this.cols) }

        for (i in 0 until this.rows) {
for (j in 0 until this.cols) {
                sum[i][j] = this.matrix[i][j] + obj.matrix[i][j]
            }
}
        return Matrix(sum, this.rows, this.cols)
    }
    operator fun minus(obj: Matrix): Matrix {
        val sub = Array(this.rows) { IntArray(this.cols) }

        for (i in 0 until this.rows) {
for (j in 0 until this.cols) {
                sub[i][j] = this.matrix[i][j] - obj.matrix[i][j]
            }
}
        return Matrix(sub, this.rows, this.cols)
    }
    operator fun times(obj: Matrix): Matrix {
        val mul = Array(this.rows) { IntArray(obj.cols) }

        for (i in 0 until this.rows) {
for (j in 0 until obj.cols) {
mul[i][j] = 0
                for (k in 0..obj.cols) {
                    mul[i][j] += this.matrix[i][k] * obj.matrix[k][j]
                }
            }
}
        return Matrix(mul, this.rows, obj.cols)
    }
}
fun main() {
    val firstMatrix = Matrix(arrayOf(intArrayOf(3, -2, 5), intArrayOf(3, 0,
4)), 2, 3)
    val secondMatrix1 = Matrix(arrayOf(intArrayOf(2, 3), intArrayOf(-9, 0),
intArrayOf(0, 4)), 3, 2)
    val secondMatrix2 = Matrix(arrayOf(intArrayOf(6, 3), intArrayOf(9,
0), intArrayOf(5, 4)), 3, 2)
    println("********************** Addition **********************")
println("Matrix 1 : ")
```

```
    println(secondMatrix2.toString())
println("Matrix 2 : ")
    println(secondMatrix1.toString())
    val addMatrix = secondMatrix2 + secondMatrix1
println("Addition : $addMatrix")

    println("********************** Subtraction **********************")
println("Matrix 1 : ")
    println(secondMatrix2.toString())
println("Matrix 2 : ")
    println(secondMatrix1.toString())
    val subMatrix = secondMatrix2 - secondMatrix1
println("Subtraction : $subMatrix")

    println("********************** Multiplication
**********************")
println("Matrix 1 : ")
println(firstMatrix.toString())
println("Matrix 2 : ")
    println(secondMatrix1.toString())     val
mulMatrix = firstMatrix * secondMatrix1
println("Multiplication : $mulMatrix") }
```

Output:

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

Matrix 2 :
(3 x 2 Matrix):
2   3
-9  0
0   4

Addition : (3 x 2 Matrix):
8   6
0   0
5   8

********** Subtraction **********
Matrix 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

********** Multiplication **********
Matrix 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
```

Exercises:

1. Swap Value of two variables without using third variable and with using third variable.

```
With using Third Variable:
Before Swapping:
The value of a is:10 and Value of B is:20
After swapping
The value of a is:20 and Value of B is:10
**********************
Without using Third Variable:
Before swapping
The value of a is:10 and b is 20
After swapping
The value of a is:20 and b is 10
```

Code:

```
fun main()
{
    println("======Using third variable=====")
var first = 10      var second = 20
    println("=======before swap=======")
println("First number is : $first")
println("Second number is : $second")
val temporary = first       first = second
second = temporary
    println("========After swap=========")
println("First number is : $first")      println("Second
number is : $second")

    println("===Without Using third variable===")
var one = 10      var two = 20
    println("========Before swap=========")
println("First number = $one")      println("Second
number = $two")

    one = one - two
two = one + two      one
= two - one

    println("========After swap========")
println("First number = $one")
println("Second number = $two") }
```

Output:

```
======Using third variable=====
=======before swap=======
First number is : 10
Second number is : 20
========After swap=========
First number is : 20
Second number is : 10
===Without Using third variable===
========Before swap=========
First number = 10
Second number = 20
========After swap========
First number = 20
Second number = 10
```

2. Create two class named as Product and Laptop. Inherit with this information: Product class should be parent and child class should be Laptop class.
   Add Product Name, Quantity, Amount per Quantity in Product class. In Laptop class add CPU name, RAM size, HDD Size, etc. of Laptop configuration.
   Create primary and secondary Constructor of both class.
   If Primary constructor is there then can we create secondary constructor in inheritance?
   If we can create secondary and primary constructor both in child class then what is restriction if parent have more than two different secondary constructor?
   Create List of 5 laptops in ArrayList and display all objects information.

3. Create two class named as Person and Student. Inherit with this information: Person class should be parent and child class should be Student class.

Add first name, last name, age in Person class. In Laptop class add enrollment no, branch, class, lab batch, etc.

Create primary and secondary Constructor of both class.

Create List of 5 students in ArrayList and display all objects information.