Kotlin Basic Practical Program

1. Write a program that prints your name and your college name.

```
fun main()
{
    val myName = "Jainish"
    val myCollegeName = "AtmiyaUniversity"
    println(myName)
    println(myCollegeName)
}
```

2. Write a program that prints your address with name.

```
fun main()
{
    val addrName = " Vruj "
    val myAddress = "Jilla Garden, flat no: A4-404 "
    println(addrName)
    println(myAddress)
}
```

3. Write a program that accept two numbers and perform all basic mathematicaloperation and print.

```
fun main()

{

   val a=100

   val b=60

   val add = a+b

   println("Add: $a + $b = $add")

   val sub = a-b

   println("Sub: $a - $b = $sub")

   val multiply = a*b

   println("Multiply: $a * $b = $multiply")

   val division = a/b

   println("Division: $a / $b = $division")

   val modulus = a%b

   println("Modulus: $a % $b = $modulus")

}
```

4. Write a program to calculate simple interest.

Kotlin Basic Practical Program

```
fun main() {
    val principal = 10000.0
    val rate = 5.5
    val time = 2.0
    val simpleInterest = (principal * rate * time) / 100

    println("Principal amount: \${principal}")
    println("Rate of interest: ${rate}% per annum")
    println("Time period: ${time} years")
    println("Simple Interest: \${simpleInterest}")
}
```

5. Write a program to calculate compound interest.

```
fun main() {
    val principalAmount = 10000
    println("Principal amount is defined as:
$principalAmount")
    val interestRate = 5
    println("The rate of interest is defined as:
$interestRate %")
```

```
    val timePeriod = 3

    println("The time period is defined as: $timePeriod
years")

    val compoundInterest = principalAmount.toDouble() *
Math.pow((1 +
interestRate.toDouble()/100.00),timePeriod.toDouble())-
principalAmount

    println(" Compound Interest is: $compoundInterest")

}
```

6. Write a program to calculate 10% bonus of salary.

```
fun main() {

    val salary = 50000.0

    val bonus = calculateBonus(salary)

    println("The 10% bonus for a salary of $$salary is
$$bonus")

}

fun calculateBonus(salary: Double): Double {

    return salary * 0.10

}
```

7. Write a program to convert KM into Meter.

```
fun main(){

   val kilometers = 5.0

   val meters = convertKmToMeters(kilometers)

   println("$kilometers kilometers is equal to
$meters meters")

}

fun convertKmToMeters(kilometers: Double):Double{

   return kilometers * 1000

}
```

8. The distance between two cities is input through keyboard. Write a program to convertand print

this distance in feet, meter, inch and centimeter.

```
fun main() {

   println("Enter the distance two cities in
kilometers: ")

   val distance = readLine()?.toDoubleOrNull()
```

```
    if (distance != null) {
        val feetConversion = distance * 3280.84
        val meterConversion = distance * 1000
        val inchConversion = distance * 39370.1
        val centimeterConversion = distance * 100000
        println("feet: $feetConversion feet")
        println("meters: $meterConversion meters")
        println("inches: $inchConversion inches")
        println("centimeters: $centimeterConversion
centimeters")
    } else {
        println("Invalid input.")
    }
}
```

9. Write a program to find volume of cylinder (v = 3.14r2h).

```
fun main() {
    val pi = 3.14
```

```
    val radius = 5

    val height = 10

    val volume = pi * radius * radius * height

    println("Volume of the cylinder is: $volume")

}
```

10. Write a program to calculate area of triangle (a = 1/2hb).

```
fun main() {

    val base = 5

    val height = 7

    val area = 0.5 * base * height

    println("$base")

    println("$height")

    println("$area")

}
```

11. Write a program to calculate area and perimeter of the rectangle.

```
fun main() {
```

```
    val length = 10
    val width = 5
    val area = length * width
    val perimeter = 2 * (length + width)
    println("area = $area")
    println("perimeter = $perimeter")
}
```

12. Write a program to calculate area of circle.

```
import kotlin.math.PI
fun main() {
    val radius = 8
    val area = PI * radius * radius
    println("$radius")
    println("$area")
}
```

13. Write a program to swap two values.

```
fun main() {
```

```
    var a = 40
    var b = 80
    println("Before swapping:")
    println("a = $a")
    println("b = $b")
    val temp = a
    a = b
    b = temp
    println("After swapping:")
    println("a = $a")
    println("b = $b")
}
```

14. Write a program to swap two values without using third variable.

```
fun main() {
    var a = 70
    var b = 30
    println("Before swapping:")
```

```
    println("a = $a")
    println("b = $b")
    a = a + b
    b = a - b
    a = a - b
    println("\nAfter swapping:")
    println("a = $a")
    println("b = $b")
}
```

15. Write a program to read the value of X and Y and print the result of following expression

(X+Y)/(X-Y)

```
import java.util.Scanner
fun main() {
    val scanner = Scanner(System.`in`)
    println("Enter the value of X:")
    val x = scanner.nextDouble()
    println("Enter the value of Y:")
```

```
    val y = scanner.nextDouble()

    val result = (x + y) / (x - y)

    println("Result of (X + Y) / (X - Y) = $result")

}
```

16. Write a program to read the value of X and Y and print the result of following

expression (X+Y)/2

```
fun main() {

    println("Enter the value of X:")

    val x = readLine()?.toInt() ?: 0

    println("Enter the value of Y:")

    val y = readLine()?.toInt() ?: 0

    val result = (x + y) / 2

    println("(X + Y) / 2 = $result")

}
```

17. Write a program to read the value of X and Y and print the result of following expression

(X+Y)*(X-Y).

```
fun main() {
    println("Enter the value of X:")
    val x = readLine()?.toInt() ?: 0
    println("Enter the value of Y:")
    val y = readLine()?.toInt() ?: 0
    val result = (x + y) * (x - y)
    println("($x + $y) * ($x - $y) = $result")
}
```

18. Write a program to read the value of X and Y and print the result of following expression

3X2+2XY+3Y2.

```
import java.util.Scanner
fun main() {
    val scanner = Scanner(System.`in`)
    println("Enter the value of X:")
    val X = scanner.nextDouble()
    println("Enter the value of Y:")
```

```
    val Y = scanner.nextDouble()

    val result = 3 * X * X + 2 * X * Y + 3 * Y * Y

    println("Result of the expression 3X^2 + 2XY + 3Y^2 is: $result")

}
```

19. Write a program to read the value of X and Y and print the result of following expression

(2X+3Y)/XY.

```
fun main() {

    println("Enter the value of X:")

    val x = readLine()!!.toDouble()

    println("Enter the value of Y:")

    val y = readLine()!!.toDouble()

    val result = (2 * x + 3 * y) / (x * y)

    println("Result = $result")

}
```

20. Write a program to convert negative to positive and positive to negative.

```
fun main() {
    println("Enter a number:")
    val number = readLine()!!.toInt()
    val result = if (number >= 0) {
        -number
    } else {
        kotlin.math.abs(number)
    }
    println("Converted result: $result")
}
```

21. Write a program that accept 5 numbers from user and find average of the value.

```
fun main() {
    var sum = 0.0
    println("Enter 5 numbers:")
    for (i in 1..5) {
        print("Enter number $i: ")
```

```
        val number = readLine()!!.toDouble()
        sum += number
    }
    val average = sum / 5
    println("Average = $average")
}
```

22. Write a program to find out the net salary of an employee and get the basic salary fromhim.

Applicable TA 4%, DA 30%, HRA 15% on basic salary.
Applicable 3% tax 12% PF on basic salary.

```
fun main() {
    println("Enter the Basic Salary:")
    val basicSalary = readLine()!!.toDouble()
    val ta = 0.04
    val da = 0.30
    val hra = 0.15
    val tax = 0.03
    val pf = 0.12
```

```
    val Ta = basicSalary * ta

    val Da = basicSalary * da

    val Hra = basicSalary * hra

    val totalAllowances = ta + da + hra

    val Tax = basicSalary * tax

    val Pf = basicSalary * pf

    val totalDeductions = tax + pf

    val netSalary = basicSalary + totalAllowances -
totalDeductions

    println("Net Salary: $netSalary")

}
```

23. Write a program to find maximum number from 2 numbers.

```
fun main() {

    val number1 = 88

    val number2 = 54

    val maxNumber = if (number1 > number2) {

        number1
```

```
    } else {
        number2
    }
    println("Maximum number $number1 and $number2
is: $maxNumber")
}
```

24. Write a program to find out minimum number from 2 numbers.

```
fun main() {
    val number1 = 90
    val number2 = 43
    val minNumber = if (number1 < number2) {
        number1
    } else {
        number2
    }
    println("Minimum number $number1 and $number2
is: $minNumber")
}
```

25. Write a program to find minimum and maximum no from 2 numbers.

```
fun main() {
    val number1 = 20
    val number2 = 9
    val minNumber = if (number1 < number2) {
        number1
    } else {
        number2
    }
    val maxNumber = if (number1 > number2) {
        number1
    } else {
        number2
    }
    println("numbers: $number1 and $number2")
    println("Minimum number: $minNumber")
    println("Maximum number: $maxNumber")
```

```
}
```

26. Write a program to check number is odd or even.

```
fun main() {
    println("Enter a number:")
    val number = readLine()!!.toInt()
    if (number % 2 == 0) {
        println("$number is even.")
    } else {
        println("$number is odd.")
    }
}
```

27. Write a program that accepts the year from user and check it leap year or not.

```
fun main() {
    println("Enter a year:")
    val year = readLine()!!.toInt()
    val isLeapYear = if (year % 4 == 0) {
```

```
        if (year % 100 == 0) {
            year % 400 == 0
        } else {
            true
        }
    } else {
        false
    }
    if (isLeapYear) {
        println("$year is a leap year.")
    } else {
        println("$year is not a leap year")
    }
}
```

28. Write a program to that accept the number from user and check it is divisible by 5 or not.

```
fun main() {
    println("Enter a number:")
```

```
    val number = readLine()!!.toInt()
    if (number % 5 == 0) {
        println("$number is divisible by 5.")
    } else {
        println("$number is not divisible by 5.")
    }
}
```

29. Write a program that accept the number from user and check its negative, positive or zero.

```
fun main() {
    print("Enter a number: ")
    val number = readLine()?.toDoubleOrNull()
    if (number != null) {
        when {
            number > 0 -> println("The number is positive.")
            number < 0 -> println("The number is negative.")
            else -> println("The number is zero.")
```

```
        }
    } else {
        println("Invalid input. Please enter a valid
number.")
    }
}
```

30. Write a program input one integer number. Check whether number is equal to 10 ornot.

```
fun main() {
    println("Please enter an integer number:")
    val number = readLine()?.toIntOrNull()
    if (number == null) {
        println("Invalid input. Please enter a valid
integer.")
    } else if (number == 10) {
        println("The number is equal to 10.")
    } else {
        println("The number is not equal to 10.")
    }
```

```
}
```

31. Write a program input one integer number check whether number is in between 0 to100 or not.

```
fun main() {
    print("Enter an integer number: ")
    val number = readLine()?.toIntOrNull()
    if (number != null) {
        if (number in 0..100) {
            println("$number is between 0 and 100.")
        } else {
            println("$number is not between 0 and 100.")
        }
    }
}
```

32. Write a program input one integer number check whether number is four digits or not.

```
fun main() {
```

```
    println("Enter an integer number:")
    val number = readLine()?.toIntOrNull()
    if (number != null && number in 1000..9999) {
        println("The number $number is a four-digit
number.")
    } else {
        println("The number is not a four-digit
number.")
    }
}
```

33. Write a program input one integer number check whether is greater than 50 and less then 200.

```
fun main() {
    print("Enter an integer number: ")
    val number = readLine()?.toIntOrNull()
    if (number != null) {
        if (number > 50 && number < 200) {
            println("$number is greater than 50 and
less than 200.")
```

```
        } else {
            println("$number is not in the range ")
        }
    } else {
        println("Invalid input. Please enter a valid
integer.")
    }
}
```

34. Write a program input one integer number display appropriate day of week.

```
fun main() {
    println("Enter an (1-7) to display the day of
the week:")
    val dayNumber = readLine()?.toIntOrNull()
    val day = when (dayNumber) {
        1 -> "Sunday"
        2 -> "Monday"
        3 -> "Tuesday"
        4 -> "Wednesday"
```

```
        5 -> "Thursday"
        6 -> "Friday"
        7 -> "Saturday"
        else -> "Invalid day number entered"
    }
    println("Day of the week: $day")
}
```

35. Write a program input one integer number display appropriate name of month.

```
fun main() {
    println("Enter the month number (1-12): ")
    val monthNumber = readLine()?.toIntOrNull()
    if (monthNumber != null && monthNumber in 1..12)
{
        val monthName = when (monthNumber) {
            1 -> "January"
            2 -> "February"
            3 -> "March"
```

```
            4 -> "April"
            5 -> "May"
            6 -> "June"
            7 -> "July"
            8 -> "August"
            9 -> "September"
            10 -> "October"
            11 -> "November"
            12 -> "December"
            else -> "Invalid month number"
        }
        println("Month name: $monthName")
    } else {
        println("Invalid input. Please enter a valid
month number")
    }
}
```

36. Write a program to find out maximum from three number.

```
fun main() {
    val num1 = 11
    val num2 = 46
    val num3 = 20
    val max = if (num1 >= num2 && num1 >= num3) {
        num1
    }
    else if (num2 >= num1 && num2 >= num3) {
        num2
    }
    else {
        num3
    }
    println("Maximum number is: $max")
}
```

37. Write a program to find out minimum from three number.

Kotlin Basic Practical Program

```
fun main() {
    val num1 = 30
    val num2 = 19
    val num3 = 55
    val min = if (num1 <= num2 && num1 <= num3) {
        num1
    }
    else if (num2 <= num1 && num2 <= num3) {
        num2
    }
    else {
        num3
    }
    println("Minimum number is: $min")
}
```

38. Enter age of person and display message as.

Kotlin Basic Practical Program

Up to 5 year	Kid
5- 12	Children
13 – 19	Teenager
20 – 30	Young
31 – 60	Mid age group
60 or above	Old

```
fun main() {
    println("Enter the age of the person:")
    val age = readLine()?.toIntOrNull()
    if (age != null) {
        val message = when {
                    age <= 5 -> "Kid"
            age in 6..12 -> "Children"
            age in 13..19 -> "Teenager"
            age in 20..30 -> "Young"
            age in 31..60 -> "Mid age group"
```

```
            else -> "Old"
        }
        println("Age $age belongs to: $message")
    } else {
        println("Invalid input. Please enter a valid
age.")
    }
}
```

39. Write a program input integer number and select number to perform following task

1 – Addition

2 – Subtraction

3 – Multiplication

4 – Division

5 – Exit

```
import java.util.Scanner
fun main() {
    val scanner = Scanner(System.`in`)
    while (true) {
```

```
println("\nMenu:")
println("1. Addition")
println("2. Subtraction")
println("3. Multiplication")
println("4. Division")
println("5. Exit")
print("Enter your choice: ")
val choice = scanner.nextInt()
if (choice == 5) {
    break
}
print("Enter the first number: ")
val num1 = scanner.nextInt()
print("Enter the second number: ")
val num2 = scanner.nextInt()
when (choice) {
    1 -> println("Result: ${num1 + num2}")
    2 -> println("Result: ${num1 - num2}")
    3 -> println("Result: ${num1 * num2}")
```

```
            4 -> {
                if (num2 != 0) {
                    println("Result:
${num1.toDouble() / num2}")
                } else {
                    println("Error: Division by
zero!")
                }
            }
            else -> println("Invalid choice. Please
enter a number from 1 to5.")
        }
    }
    println("Exiting the program.")
}
```

40. Write a program input 5 subject marks and find class and result.

```
fun main() {
    val subjects = 5
```

Kotlin Basic Practical Program

```
    val marks = IntArray(subjects)
    for (i in 0 until subjects) {
        print("Enter marks for Subject ${i + 1}: ")
        marks[i] = readLine()!!.toInt()
    }
    val total = marks.sum()
    val average = total / subjects.toDouble()
    val result = when {
        average >= 80 -> "Distinction"
        average >= 60 -> "First Class"
        average >= 50 -> "Second Class"
        average >= 40 -> "Pass Class"
        else -> "Fail"
    }
    println("\nTotal Marks: $total")
    println("Average Marks: %.2f".format(average))
    println("Result: $result")
}
```

41. Write a program to find number which number is divisible by 3 but not divisible by 7.

```
fun main() {

    val start = 1

    val end = 100

    println("Numbers divisible by 3 but not
divisible by 7:")

    for (num in start..end) {

        if (num % 3 == 0 && num % 7 != 0) {

            println(num)

        }

    }

}
```

42. Write a program that reads a sales bill amount calculate discount on bill amount as follows if bill amount less than 5000 calculate 7.5% discount on bill amount otherwise calculate11% discount on bill. Print bill amount discount amount and net payable bill.

```
fun main() {
```

Kotlin Basic Practical Program

```
    println("Enter the bill amount:")
    val billAmount = readLine()?.toDoubleOrNull()
    if (billAmount != null) {
        val discount = if (billAmount < 5000) {
            billAmount * 0.075
        } else {
            billAmount * 0.11
        }
        val netPayable = billAmount - discount
        println("Bill Amount: \u20B9$billAmount")
        println("Discount Amount: \u20B9$discount")
        println("Net Payable Bill:
\u20B9$netPayable")
    } else {
        println("Invalid input. Please enter a valid
numeric bill amount.")
    }
}
```

43. An electric power distribution company charger its domestic customers as follows Consumption unit rate of charge.

0 – 200	RS. 0.50 per unit
201 – 400	RS. 100 + RS. 0.65 per unit
401 – 600	RS. 230 + RS. 0.80 per unit
601 and above	RS. 390 + RS. 1.00 per unit

Read the customer number and power consumed and print the amount to be paid by the customer

```
fun main() {
    val customerNumber: Int
    val powerConsumed: Int
    val rate: Double
    val amount: Double
```

Kotlin Basic Practical Program

```
    println("Enter customer number:")
    customerNumber = readLine()!!.toInt()
    println("Enter power consumed (in units):")
    powerConsumed = readLine()!!.toInt()
    when {
        powerConsumed in 0..200 -> {
            rate = 0.50
            amount = powerConsumed * rate
        }
        powerConsumed in 201..400 -> {
            rate = 0.65
            amount = 100 + (powerConsumed - 200) *
rate
        }
        powerConsumed in 401..600 -> {
            rate = 0.80
            amount = 230 + (powerConsumed - 400) *
rate
        }
```

```
        powerConsumed >= 601 -> {
            rate = 1.00
            amount = 390 + (powerConsumed - 600) *
rate
        }
        else -> {
            println("Invalid power consumption
value.")
            return
        }
    }
    println("Customer $customerNumber needs to pay
Rs. $amount")
}
```

44. Get a string from user and display it in upper case.

```
import java.util.Scanner
fun main() {
    val scanner = Scanner(System.`in`)
```

```
    print("Enter a string: ")
    val inputString = scanner.nextLine()
    val upperCaseString = inputString.toUpperCase()
    println("String in uppercase: $upperCaseString")
}
```

45. Get a string from user and display it in lower case.

```
import java.util.Scanner
fun main() {
    val scanner = Scanner(System.`in`)
    print("Enter a string: ")
    val inputString = scanner.nextLine()
    val lowerCaseString = inputString.toLowerCase()
    println("String in uppercase: $lowerCaseString")
}
```

46. Get a string from user and check it is vowel or consonants.

```
fun main() {
    val input = readLine()?.trim()?.toLowerCase()
```

```
    if (input != null && input.length == 1) {
        if (input in "aeiou") {
            println("$input is a vowel.")
        } else if (input in
"abcdefghijklmnopqrstuvwxyz") {
            println("$input is a consonant.")
        } else {
            println("Invalid input. Please enter a
single alphabet character.")
        }
    } else {
        println("Invalid input. Please enter a
single alphabet character.")
    }
}
```

47. Write a program that accept character from keyboard and determine whether the character is a capital letter, small letter, digit or a special character.

```
fun main() {
    print("Enter a character: ")
```

```
    val ch = readLine()!!.first()

    when {

        ch in 'A'..'Z' -> println("$ch is a capital
letter.")

        ch in 'a'..'z' -> println("$ch is a small
letter.")

        ch in '0'..'9' -> println("$ch is a digit.")

        else -> println("$ch is a special
character.")

    }

}
```

48. Print 1 to 10.

```
fun main() {

    for (i in 1..10) {

        println(i)

    }

}
```

49. Print 2 4 6 8 10.

```
fun main() {
    for (i in 2..10 step 2) {
        print("$i ")
    }
}
```

50. Print 1 3 5 7 9.

```
fun main() {
    for (i in 1..9 step 2) {
        print("$i ")
    }
}
```

51. Print 1 2 4 8 16 32 64 128 256 512 1024.

```
fun main() {
    var number = 1
    repeat(11) {
        print("$number ")
        number *= 2
```

```
    }
}
```

52. Print 10 to 1.

```
fun main() {
    for (i in 10 downTo 1) {
        println(i)
    }
}
```

53. Print 1 10 2 9 3 8 4 7 5 6.

```
fun main() {
    var i = 1
    var j = 10
    while (i <= 5 && j >= 6) {
        print("$i $j ")
        i++
        j--
    }
```

```
}
```

54. Print 1 2 3 5 6 7 9 -_ _ _ n.

```
fun main() {
    val n = 10
    var count = 0
    var num = 1
    while (count < n) {
        if (num != 4 && num != 8) {
            print("$num ")
            count++
        }
        num++
    }
}
```

55. Input and number display table of that number.

```
import java.util.Scanner
fun main() {
```

```
    val scanner = Scanner(System.`in`)
    print("Enter a number: ")
    val number = scanner.nextInt()
    println("Multiplication table of $number:")
    for (i in 1..10) {
        println("$number * $i = ${number * i}")
    }
}
```

56. Print series of 1/1, 1/2, 1/3, _ _ _ 1/N.

```
fun main() {
    val N = 5
    for (i in 1..N) {
        println("1/$i")
    }
}
```

57. Print series of 1/2, 2/3, 3/4, 4/5, _ _ _ N/N+1.

```
fun main() {
```

```
    val N = 10
    for (i in 1..N) {
        println("$i/${i + 1}")
    }
}
```

58. Print series 1 + 4 – 9 + 16 – 25 + 36 + _ _ _ + N2.

```
fun main() {
    val N = 7
    var sign = 1
    var num = 1
    for (i in 1..N) {
        val term = sign * num * num
        print("$term ")
        sign *= -1
        num++
    }
}
```

59. Print 0 1 1 2 3 5 8 13 21 34 55 (Fibonacci).

```
fun main() {
    val n = 11
    var a = 0
    var b = 1
    print("Fibonacci series: ")
    repeat(n) {
        print("$a ")
        val sum = a + b
        a = b
        b = sum
    }
}
```

60. Print factorial value of given number.

```
fun main() {
    val number = 5
    var factorial = 1
    for (i in 1..number) {
```

```
        factorial *= i
    }
    println("Factorial of $number = $factorial")
}
```

61. Check whether the number is prime or not.

```
fun main() {
    val number = 102
    var isPrime = true
    if (number <= 1) {
        isPrime = false
    } else {
        for (i in 2 until number) {
            if (number % i == 0) {
                isPrime = false
                break
            }
        }
    }
```

```
    if (isPrime) {
        println("$number is a prime number.")
    } else {
        println("$number is not a prime number.")
    }
}
```

62. Print prime number between given range.

```
fun main() {
    val start = 1
    val end = 100
    println("Prime numbers between $start and $end are:")
    for (num in start..end) {
        if (isPrime(num)) {
            println(num)
        }
    }
}
```

```
fun isPrime(num: Int): Boolean {
    if (num <= 1) {
        return false
    }
    for (i in 2 until num) {
        if (num % i == 0) {
            return false
        }
    }
    return true
}
```

63. Write program that display square, cubes and factorials of all integer from 1 to 10.

```
fun main() {
    for (i in 1..10) {
        val square = i * i
        val cube = i * i * i
        val factorial = factorial(i)
```

```
        println("Number: $i")
        println("Square: $square")
        println("Cube: $cube")
        println("Factorial: $factorial")
        println()
    }
}
fun factorial(n: Int): Long {
    var result = 1L
    for (i in 1..n) {
        result *= i
    }
    return result
}
```

64. Display sum of digit.

```
fun main() {
    val number = 12345
    var sum = 0
```

```
    var n = number
    while (n != 0) {
        sum += n % 10
        n /= 10
    }
    println("Sum of digits of $number is: $sum")
}
```

65. Display reverse number.

```
fun main() {
    val number = 12345
    var reversed = 0
    var original = number
    while (original != 0) {
        val digit = original % 10
        reversed = reversed * 10 + digit
        original /= 10
    }
```

```
    println("Original number: $number")
    println("Reversed number: $reversed")
}
```

66. Check entered number is Armstrong or not.

```
fun main() {
    fun isArmstrong(number: Int): Boolean {
        val digits = number.toString().map {
it.toString().toInt() }
        val power = digits.size
        val sumOfPowers = digits.map {
Math.pow(it.toDouble(),
            power.toDouble()).toInt() }.sum()
        return sumOfPowers == number
    }
    val number = 600
    if (isArmstrong(number)) {
```

```
        println("$number is an Armstrong number.")
    } else {
        println("$number is not an Armstrong
number.")
    }
}
```

67. Check entered number is palindrome or not.

```
fun main() {
    println("Enter a number:")
    val input = readLine() ?: return
    val number = input.toIntOrNull() ?: return
    if (isPalindrome(number)) {
        println("$number is a palindrome.")
    } else {
        println("$number is not a palindrome.")
    }
}
```

```
fun isPalindrome(number: Int): Boolean {
    val original = number.toString()
    val reversed = original.reversed()
    return original == reversed
}
```

68. Count odd and even digits from given number.

```
fun main() {
    val number = 1234567
    var evenCount = 0
    var oddCount = 0
    var num = number
    while (num != 0) {
        val digit = num % 10
        if (digit % 2 == 0) {
            evenCount++
        } else {
            oddCount++
        }
```

```
        num /= 10
    }
    println("Number of even digits: $evenCount")
    println("Number of odd digits: $oddCount")
}
```

69. Accept number and find how many zeros are there.

```
fun main() {
    println("Enter a number: ")
    val number = readLine()?.toIntOrNull()
    if (number != null) {
        val countZeros = number.toString().count { it == '0' }
        println("Number of zeros in $number: $countZeros")
    } else {
        println("Invalid input. Please enter a valid number.")
    }
```

```
}
```

70. Count digits of given number.

```
fun countDigits(number: Int): Int {
    var count = 0
    var num = number
    if (num < 0) {
        num = -num
    }
    while (num > 0) {
        num /= 10
        count++
    }
    return count
}
fun main() {
    val number = 12345
    val digitCount = countDigits(number)
```

```
   println("Number of digits in $number:
$digitCount")

}
```

71.

```
1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5
```

```
fun main() {
    for (i in 1..5) {
        for (j in 1..5) {
            print("$j ")
        }
        println()
    }
}
```

72.

```
1 1 1 1 1
2 2 2 2 2
3 3 3 3 3
4 4 4 4 4
5 5 5 5 5
```

```
fun main() {
    for (i in 1..5) {
        for (j in 1..5) {
            print("$i ")
        }
        println()
    }
}
```

73.

```
1 2 3 4 5
2 4 6 8 10
3 6 9 12 15
```

```
4 8 12 16 20

5 10 15 20 25
```

```
fun main() {
    for (i in 1..5) {
        for (j in 1..5) {
            print("${i * j}\t")
        }
        println()
    }
}
```

74.

```
1 2 3 4 5

$ 2 3 4 5

$ $ 3 4 5

$ $ $ 4 5

$ $ $ $ 5
```

```
fun main() {
    for (i in 0 until 5) {
```

```
        for (j in 0 until 5) {
            if (j < i) {
                print("$ ")
            } else {
                print("${j + 1} ")
            }
        }
        println()
    }
}
```

75.

```
1 $ $ $ $
2 2 $ $ $
3 3 3 $ $
4 4 4 4 $
5 5 5 5 5
```

```
fun main() {
    for (i in 1..5) {
```

```
        for (j in 1..5) {
            if (j <= i) {
                print("$i ")
            } else {
                print("$ ")
            }
        }
        println()
    }
}
```

76.

```
X 0 0 0 0
X X 0 0 0
X X X 0 0
X X X X 0
X X X X X
fun main() {
    for (i in 0 until 5) {
```

```
        for (j in 0 until 5) {
            if (j <= i) {
                print("X ")
            } else {
                print("0 ")
            }
        }
        println()
    }
}
```

77.

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
fun main() {
    for (i in 1..5) {
```

```
        for (j in 1..i) {
            print("$i ")
        }
        println()
    }
}
```

78.

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
fun main() {
    for (i in 1..5) {
        for (j in 1..i) {
            print("$j ")
        }
        println()
```

```
    }
}
```

79.

```
5
5 4
5 4 3
5 4 3 2
5 4 3 2 1
fun main() {
    for (i in 5 downTo 1) {
        for (j in 5 downTo i) {
            print("$j ")
        }
        println()
    }
}
```

80.

```
1

2 3

4 5 6

7 8 9 10

11 12 13 14 15
```

```
fun main() {
    var number = 1
    for (i in 1..5) {
        for (j in 1..i) {
            print("$number ")
            number++
        }
        println()
    }
}
```

```
81.

1

0 1
```

```
0 1 0
1 0 1 0
1 0 1 0 1
fun main() {
    for (i in 0 until 5) {
        for (j in 0 until 5 - i - 1) {
            print(" ")
        }
        for (j in 0..i) {
            print("${if (j % 2 == 0) 1 else 0} ")
        }
        println()
    }
}
```

82.

```
1
1 0
1 0 1
```

```
1 0 1 0

1 0 1 0 1
```

```
fun main() {
    for (i in 1..5) {
        for (j in 1..i) {
            if (j % 2 == 0) {
                print("0 ")
            } else {
                print("1 ")
            }
        }
        println()
    }
}
```

83.

```
        *
      * *
    * * *
```

```
  * * * *
* * * * *
fun main() {
    for (i in 1..5) {
        for (j in 1..(5 - i)) {
            print(" ")
        }
        for (k in 1..i) {
            print("*")
        }
        println()
    }
}
```

84.

```
* * * * *
* * * *
* * *
* *
```

```
*
```

```
fun main() {
    for (i in 1 .. 10 ) {
        for (j in i..5) {
            print("* ")
        }
        println()
    }
}
```

85.

```
    *
   * *
  * * *
 * * * *
* * * * *
```

```
fun main() {
    for (i in 0 .. 4){
        for (j in 0 .. 4-i){
```

```
            print(" ")
        }
        for (k in 0 .. i){
            print("* ")
        }
        println(" ")
    }
}
```

86.

```
A A A A A
B B B B B
C C C C C
D D D D D
E E E E E
fun main() {
    val rows = 5
    val cols = 5
```

```
    for (i in 0 until rows) {
        for (j in 0 until cols) {
            print("${'A' + i} ")
        }
        println()
    }
}
```

87.

```
* * * * *
*       *
*       *
*       *
* * * * *
fun main() {
    val rows = 5
    val cols = 5

    for (i in 0 until rows) {
```

```
        for (j in 0 until cols) {
            if (i == 0 || i == rows - 1 || j == 0 ||
j == cols - 1) {
                print("* ")
            } else {
                print("  ")
            }
        }
        println()
    }
}
```

88.

```
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
fun main() {
```

```
    val rows = 5


    for (i in 1..rows) {
        for (j in i downTo 1) {
            print("$j ")
        }
        println()
    }
}
```

89.

```
A
B C
D E F
G H I J
K L M N O
fun main() {
    val rows = 5
    var currentChar = 'A'
```

```
    for (i in 1..rows) {
        for (j in 1..i) {
            print("$currentChar ")
            currentChar++
        }
        println()
    }
}
```