**TASK 4**

1. **SWAP-CASE PROGRAM**

**def** swap\_case(s):

    a=''

**for** i **in** s:

**if**(i.isupper()==**True**):

            a+=(i.lower())

**elif**(i.islower()==**True**):

            a+=(i.upper())

**else**:

            a+=i

**return** a

**if** \_\_name\_\_ == '\_\_main\_\_':

    s = **input**()

    result = swap\_case(s)

**print**(result)

1. **STRING SPLIT AND JOIN PROGRAM**

**def** split\_and\_join(line):

    line=line.split(" ")

    line="-".join(line)

**return** line

**if** \_\_name\_\_ == '\_\_main\_\_':

    line = **input**()

    result = split\_and\_join(line)

**print**(result)

1. **PRINTING NAME PROGRAM**

**def** print\_full\_name(first, last):

**print**("Hello {} {}! You just delved into python.".**format**(first,last))

**if** \_\_name\_\_ == '\_\_main\_\_':

    first\_name = **input**()

    last\_name = **input**()

    print\_full\_name(first\_name, last\_name)

1. **MUTATIONS PROGRAM**

**def** mutate\_string(string, position, character):

    a=**list**(string)

    a[position]=character

    string=''.join(a)

**return** string

**if** \_\_name\_\_ == '\_\_main\_\_':

    s = **input**()

    i, c = **input**().split()

    s\_new = mutate\_string(s, **int**(i), c)

**print**(s\_new)

1. **FINDING A STRING PROGRAM**

**def** count\_substring(string, sub\_string):

    count=0

**for** i **in** **range**(**len**(string)-**len**(sub\_string)+1):

**if**(string[i:i+**len**(sub\_string)]==sub\_string):

            count+=1

**return** count

**if** \_\_name\_\_ == '\_\_main\_\_':

    string = **input**().strip()

    sub\_string = **input**().strip()

    count = count\_substring(string, sub\_string)

**print**(count)

1. **STRING VALIDATORS PROGRAM**

**if** \_\_name\_\_ == '\_\_main\_\_':

    s = **input**()

**print**(**any**(i.isalnum() **for** i **in** s))

**print**(**any**(i.isalpha() **for** i **in** s))

**print**(**any**(i.isdigit() **for** i **in** s))

**print**(**any**(i.islower() **for** i **in** s))

**print**(**any**(i.isupper() **for** i **in** s))

1. **TEXT ALIGNMENT PROGRAM**

thickness = **int**(**input**())

c = 'H'

*#Top Cone*

**for** i **in** **range**(thickness):

**print**((c\*i).rjust(thickness-1)+c+(c\*i).ljust(thickness-1))

*#Top Pillars*

**for** i **in** **range**(thickness+1):

**print**((c\*thickness).center(thickness\*2)+(c\*thickness).center(thickness\*6))

*#Middle Belt*

**for** i **in** **range**((thickness+1)//2):

**print**((c\*thickness\*5).center(thickness\*6))

*#Bottom Pillars*

**for** i **in** **range**(thickness+1):

**print**((c\*thickness).center(thickness\*2)+(c\*thickness).center(thickness\*6))

*#Bottom Cone*

**for** i **in** **range**(thickness):

**print**(((c\*(thickness-i-1)).rjust(thickness)+c+(c\*(thickness-i-1)).ljust(thickness)).rjust(thickness\*6))

1. **TEXT WRAP PROGRAM**

**import** textwrap

**def** wrap(string, max\_width):

**return** textwrap.fill(string,max\_width)

**if** \_\_name\_\_ == '\_\_main\_\_':

    string, max\_width = **input**(), **int**(**input**())

    result = wrap(string, max\_width)

**print**(result)

1. **DESIGNER DOOR MAT PROGRAM**

k,l=**map**(**int**,**input**().split())

**for** i **in** **range**(1,k,2):

**print**(**str**('.|.'\*i).center(l,'-'))

**print**("WELCOME".center(l,'-'))

**for** i **in** **range**(k-2,-1,-2):

**print**(**str**('.|.'\*i).center(l,'-'))

1. **STRING FORMATTING PROGRAM**

**def** print\_formatted(number):

    l=**len**(**bin**(number)[2:])

**for** i **in** **range**(1,number+1):

**print**(**str**(i).rjust(l,' '),end=" ")

**print**(**oct**(i)[2:].rjust(l,' '),end=" ")

**print**(((**hex**(i)[2:]).upper()).rjust(l,' '),end=" ")

**print**(**bin**(i)[2:].rjust(l,' '),end=" ")

**print**(" ")

**if** \_\_name\_\_ == '\_\_main\_\_':

    n = **int**(**input**())

    print\_formatted(n)

1. **ALPHABET RANGOLI**

**def** print\_rangoli(size):

    o="abcdefghijklmnopqrstuvwxyz"

    data=[o[i] **for** i **in** **range**(n)]

    it=**list**(**range**(n))

    it=it[:-1]+it[::-1]

**for** i **in** it:

        tem=data[-(i+1):]

        ro=tem[::-1]+tem[1:]

**print**("-".join(ro).center(n\*4-3,"-"))

**if** \_\_name\_\_ == '\_\_main\_\_':

    n = **int**(**input**())

    print\_rangoli(n

1. **CAPITALIZE PROGRAM**

**def** solve(s):

**for** i **in** s[:].split():

        s=s.replace(i,i.capitalize())

**return** s

**if** \_\_name\_\_ == '\_\_main\_\_':

    fptr = **open**(os.environ['OUTPUT\_PATH'], 'w')

    s = **input**()

    result = solve(s)

    fptr.write(result + '\n')

    fptr.close()

1. **MINION GAME PROGRAM**

**def** minion\_game(string):

    p1=0

    p2=0

    length=**len**(string)

**for** i **in** **range**(length):

**if**(s[i] **in** "AEIOU"):

            p1+=(length)-i

**else**:

            p2+=(length)-i

**if** p1>p2:

**print**("Kevin",p1)

**elif** p1<p2:

**print**("Stuart",p2)

**elif** p1==p2:

**print**("Draw")

**else**:

**print**("Draw")

**if** \_\_name\_\_ == '\_\_main\_\_':

    s = **input**()

    minion\_game(s)

1. **MERGE THE TOOLS PROGRAM**

**def** merge\_the\_tools(string, k):

**for** part **in** **zip**(\*[**iter**(string)]\*k):

        d=**dict**()

**print**(''.join([d.setdefault(c,c) **for** c **in** part **if** c **not** **in** d]))

**if** \_\_name\_\_ == '\_\_main\_\_':

    string, k = **input**(), **int**(**input**())

    merge\_the\_tools(string, k)