ACKNOWLEDGEMENT

PRACTICAL EXERCISE-1A

Q: Write a program to find the factorial

PROGRAM: -

school project > practicals > prog1.py > ...

```
1  number = int(input("Enter a number:"))
2  inp_num = number
3  factorial = 1
4  while number > 0:
5      factorial = factorial*number
6      number -= 1
7
8  print(f"The factorial of {inp_num} is {factorial}")
9
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog1.py"
Enter a number:64
The factorial of 64 is 126886932185884164103433389335161480802865516174545192198801894375214704230400000000000000
PS C:\Users\maste\OneDrive\Desktop\phyton programming>
```

PRACTICAL EXERCISE-1B

Q: Write a program to check if a number is prime or not prime

PROGRAM: -

school project > practicals > prog2.py > ...

```
number = int(input("Enter a number: "))

if number < 2:
    print(number, "is not a prime number")
else:
    is_prime = True

    for i in range(2, int(number**0.5) + 1):
        if number % i == 0:
            is_prime = False
            break

    if is_prime:
        print(number, "is a prime number")
    else:
        print(number, "is not a prime number")

```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog2.py"
Enter a number: 997
997 is a prime number
PS C:\Users\maste\OneDrive\Desktop\phyton programming>
```

PRACTICAL EXERCISE-2

Q: Write a program to find the sum of a list recursively

PROGRAM: -

school project > practicals > prog3.py > summation

```
1   inpt_list = eval(input("Enter a list: "))
2
3   def summation(lst):
4       if len(lst) == 0:
5           return 0
6       else:
7           return lst[0] + summation(lst[1:])
8
9   result = summation(inpt_list)
10  print(f"The sum of the list: {inpt_list} is {result}")
11
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog3.py"
Enter a list: [131,24,46]
The sum of the list: [131, 24, 46] is 201
PS C:\Users\maste\OneDrive\Desktop\phyton programming>
```

PRACTICAL EXERCISE-3

Q: Write a program to calculate the n^{th} term of Fibonacci series.

PROGRAM: -

school project > practicals > prog3.py > ...

```python
def fib(n):
    if n < 2:
        return n
    return fib(n-1) + fib(n-2)

def get_fibonacci_term(n):
    print(f"Calculating the {n}th Fibonacci term...")
    result = fib(n)
    print(f"The {n}th Fibonacci term is {result}")

n = int(input("Enter the nth term to calculate: "))
get_fibonacci_term(n)
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog3.py"
Enter the nth term to calculate: 23
Calculating the 23th Fibonacci term...
The 23th Fibonacci term is 28657
PS C:\Users\maste\OneDrive\Desktop\phyton programming>
```

PRACTICAL EXERCISE-4

Q: Write a program to search any word in given string or sentence

PROGRAM: -

school project > practicals > prog4.py > ...

```
1  text = "This is a sample text. Here is another line."
2
3  query = input("Enter word to search: ")
4
5  lines = text.split('. ')
6  line_num = 1
7
8  for line in lines:
9    words = line.split()
10
11   word_num = 1
12
13   for word in words:
14     if word == query:
15       print(f"Found {query} at line {line_num}, word {word_num}")
16
17     word_num += 1
18
19   line_num += 1
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog4.py"
Enter word to search: sample
Found sample at line 1, word 4
```

PRACTICAL EXERCISE-5

Q: Write a program to read and display file content line by line with each word separated by #

PROGRAM: -

school project > practicals > prog5.py > ...

```
filename = r'C:\Users\maste\OneDrive\Desktop\phyton programming\school
project\practicals\data.txt'

with open(filename) as file:
  for line in file:
    words = line.strip().split()

    print('#'.join(words))

```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog5.py"
Eu#do#ad#sunt#mollit#ex#ex#quis#et#ut#est#velit.#Fugiat#ad#eu#nulla#voluptate.#Lorem#exercitation#Lorem#voluptate#exercitation#proident#aliquip#nisi.

Nulla#laboris#ut#duis#eu#pariatur#tempor.#Sunt#velit#reprehenderit#aliqua#dolor#culpa#proident#adipisicing#labore#aliqua#non#enim#elit.#Id#consectetur#ex#non#esse.#Eiusmod#ea#irure#incididunt#eiusmod.#Ex#qui#amet#dolore#Lorem#exercitation#consectetur#mollit#officia#velit#pariatur#sit#ullamco.

Sit#aliquip#consectetur#velit#aute.#Est#laboris#aliqua#ea#ut#sint.#Adipisicing#adipisicing#cupidatat#ea#nulla#cillum#enim#ex#est#officia#reprehenderit.#Est#adipisicing#proident#cupidatat#dolor#duis#adipisicing#proident.
PS C:\Users\maste\OneDrive\Desktop\phyton programming>
```

PRACTICAL EXERCISE-6

Q: Write a program to read the content of a file and display the total number of consonants, uppercase, vowels and lowercase characters.

PROGRAM: -

school project > practicals > prog6.py > ...

```
consonants_upper = 0
vowels = 0
lower_chars = 0

VOWELS = 'aeiou'

with open(r'C:\Users\maste\OneDrive\Desktop\phyton programming\school
project\practicals\data.txt','r') as f:

  for line in f:
    for char in line:
      if char.isupper() and char not in VOWELS:
        consonants_upper += 1
      elif char in VOWELS:
        vowels += 1
      elif char.islower():
        lower_chars += 1

print("Consonant upper case letters:", consonants_upper)
print("Vowels:", vowels)
print("Lower case characters:", lower_chars)
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog6.py"
Consonant upper case letters: 14
Vowels: 248
Lower case characters: 292
PS C:\Users\maste\OneDrive\Desktop\phyton programming>
```

PRACTICAL EXERCISE-7

Q: Create a binary list to store roll number and name and create a search function for searching name using role number.

PROGRAM: -

school project > practicals > prog7.py > ...

```python
student_records = []

def add_student():
    roll = int(input("Enter student roll number: "))
    name = input("Enter student name: ")
    student_records.append((roll, name))
    student_records.sort()
    print("Student added successfully!")

def search_by_roll():
    roll = int(input("Enter roll number to search: "))
    i = 0
    j = len(student_records) - 1
    while i <= j:
        mid = (i + j) // 2
        if student_records[mid][0] == roll:
            print(f"Name for roll {roll} is {student_records[mid][1]}")
            return
        elif roll < student_records[mid][0]:
            j = mid - 1
        else:
            i = mid + 1
    print("Roll number not found")

while True:
    print("1. Add Student")
    print("2. Search by Roll Number")
    print("3. Exit")

    choice = int(input("Enter choice: "))
    if choice==1:
        add_student()
    elif choice==2:
        search_by_roll()
    elif choice==3:
        break
    else:
        print("Invalid choice")
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog7.py"
1. Add Student
2. Search by Roll Number
3. Exit
Enter choice: 1
Enter student roll number: 23
Enter student name: xyz
Student added successfully!
1. Add Student
2. Search by Roll Number
3. Exit
Enter choice: 2
Enter roll number to search: 23
Name for roll 23 is xyz
1. Add Student
2. Search by Roll Number
3. Exit
Enter choice:
```

PRACTICAL EXERCISE-8

Q Write a program to create a binary file to store roll number marks and name and write a function to update marks.

PROGRAM: -

school project > practicals > prog8.py > ...

```python
import pickle
import os

records = []

def show_menu():
    print("\nStudent Records")
    print("1. Add New Student")
    print("2. Update Marks")
    print("3. View Records")
    print("4. Exit")

    choice = int(input("Enter choice: "))
    perform_action(choice)

def perform_action(choice):
    if choice == 1:
        add_student()
    elif choice == 2:
        update_marks()
    elif choice == 3:
        view_records()
    elif choice == 4:
        exit()
    else:
        print("Invalid choice")

def add_student():
    roll = int(input("Enter roll number: "))
    name = input("Enter name: ")
    marks = int(input("Enter marks: "))

    records.append({'roll':roll, 'name':name, 'marks':marks})
    print("Record added successfully!")

def update_marks():
    roll = int(input("Enter roll number to update marks: "))
    new_marks = int(input("Enter new marks: "))
    update_record(roll, new_marks)

def update_record(roll, new_marks):
    for record in records:
        if record['roll'] == roll:
            record['marks'] = new_marks
            print("Marks updated successfully!")
            return
    print("Roll number not found")

def view_records():
    for record in records:
        print(record['roll'], record['name'], record['marks'])

if os.path.exists('student_data.pkl'):
  with open('student_data.pkl', 'rb') as f:
    records = pickle.load(f)

while True:
  show_menu()

  with open('student_data.pkl', 'wb') as f:
    pickle.dump(records, f)
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog8.py"

Student Records
1. Add New Student
2. Update Marks
3. View Records
4. Exit
Enter choice: 1
Enter roll number: 23234
Enter name: ddwdd
Enter marks: 225
Record added successfully!

Student Records
1. Add New Student
2. Update Marks
3. View Records
4. Exit
Enter choice: 3
5 dt 76
23234 ddwdd 225

Student Records
1. Add New Student
2. Update Marks
3. View Records
4. Exit
Enter choice:
```

PRACTICAL EXERCISE-9

Q: Write a program to create a csv file, storing employee number and salary, and also write a search function.

PROGRAM: -

school project > practicals > prog9.py > ...

```
import csv
records = []
def show_menu():
  print()
  print("Employee Records")
  print("1. Search Records")
  print("2. View Records")
  print("3. Exit")
  choice = int(input("Enter choice: "))
  perform_action(choice)
def perform_action(choice):
  if choice == 1:
    search_records()
  elif choice == 2:
    view_records()
  elif choice == 3:
    exit()
  else:
    print("Invalid option")
def search_records():
  emp_no = int(input("Enter employee number to search: "))
  search_record(emp_no)
def search_record(emp_no):
  for record in records:
    if record['emp_no'] == emp_no:
      print(record['emp_no'], record['salary'])
      return
  print("Employee number not found")
def view_records():
  print("Emp No.\tSalary")
  for record in records:
    print(record['emp_no'], '\t', record['salary'])
with open('employees.csv', 'r') as file:
  reader = csv.reader(file)
  next(reader) # skip header row
  for row in reader:
    records.append({'emp_no': int(row[0]), 'salary': float(row[1])})
while True:
  show_menu()
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python/Python311/pyth
/maste/OneDrive/Desktop/phyton programming/school project/practicals/prog9.py"

Employee Records
1. Search Records
2. View Records
3. Exit
Enter choice: 1
Enter employee number to search: 23
Employee number not found

Employee Records
1. Search Records
2. View Records
3. Exit
Enter choice: 3
```

PRACTICAL EXERCISE-10

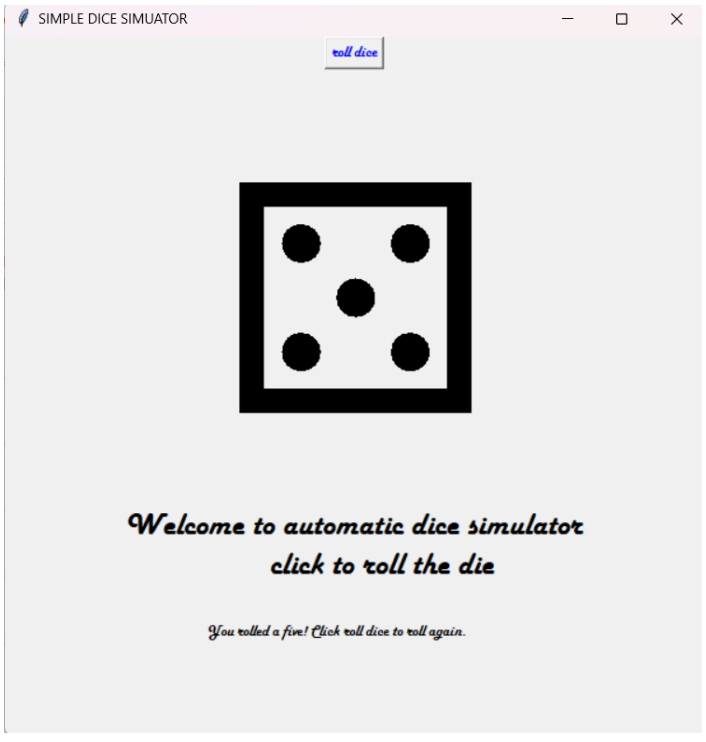
Q: Write a program to generate random number one to 6 and simulator dice.

PROGRAM: -

school project > practicals > prog10.py > ...

```
import random
import tkinter

root = tkinter.Tk()
root.title('SIMPLE DICE SIMUATOR')
root.geometry('600x600')

label = tkinter.Label(root, text='', font=('Helvetica', 260))

label2 = tkinter.Label(root, text='Welcome to automatic dice simulator\n      click to roll the die', font=('Harlow Solid Italic',20))
label2.place(x=100,y=400)

def roll_dice():
    # just some unicode values for the die faces nothing else
    value = ['\u2680', '\u2681', '\u2682', '\u2683', '\u2684', '\u2685']
    result=random.choice(value)
    label.configure(text=result)
    label.pack()
    if(result=='\u2680'):
        label3=tkinter.Label(root,text='You rolled a one! Click roll dice to roll again.',font=('Harlow Solid Italic',10))
        label3.place(x=170,y=500)
    elif(result=='\u2681'):
        label3=tkinter.Label(root,text='You rolled a two! Click roll dice to roll again.',font=('Harlow Solid Italic',10))
        label3.place(x=170,y=500)
    elif(result=='\u2682'):
        label3=tkinter.Label(root,text='You rolled a three! Click roll dice to roll again.',font=('Harlow Solid Italic',10))
        label3.place(x=170,y=500)
    elif(result=='\u2683'):
        label3=tkinter.Label(root,text='You rolled a four! Click roll dice to roll again.',font=('Harlow Solid Italic',10))
        label3.place(x=170,y=500)
    elif(result=='\u2684'):
        label3=tkinter.Label(root,text='You rolled a five! Click roll dice to roll again.',font=('Harlow Solid Italic',10))
        label3.place(x=170,y=500)
    elif(result=='\u2685'):
        label3=tkinter.Label(root,text='You rolled a six! Click roll dice to roll again.',font=('Harlow Solid Italic',10))
        label3.place(x=170,y=500)
button = tkinter.Button(root, text='roll dice',font=('Harlow Solid Italic',10), foreground='blue', command=roll_dice)
button.pack()
root.mainloop()
```

OUTPUT: -



PRACTICAL EXERCISE-11

Q: Write a program for currency conversion from dollar to rupee.

PROGRAM: -

school project > practicals > prog11.py > ...

```python
exchange_rate = 83.50
print("Dollar to Rupee Converter")
while True:

  print("1. Dollar to Rupees")
  print("2. Rupees to Dollars")
  print("3. Exit")

  choice = int(input("Enter choice: "))

  if choice == 1:
    dollars = float(input("Enter amount in dollars: "))
    rupees = round(dollars * exchange_rate, 2)
    print(f"{dollars} dollars = {rupees} rupees")

  elif choice == 2:
    rupees = float(input("Enter amount in rupees: "))
    dollars = round(rupees / exchange_rate, 2)
    print(f"{rupees} rupees = {dollars} dollars")

  elif choice == 3:
    break

  else:
    print("Invalid choice")

print("Program ended")
```

OUTPUT: -

```
PS C:\Users\maste\OneDrive\Desktop\phyton programming> & C:/Users/maste/AppData/Local/Programs/Python
/Python311/python.exe "c:/Users/maste/OneDrive/Desktop/phyton programming/school project/practicals/p
rog11.py"
Dollar to Rupee Converter
1. Dollar to Rupees
2. Rupees to Dollars
3. Exit
Enter choice: 1
Enter amount in dollars: 232332
232332.0 dollars = 19399722.0 rupees
1. Dollar to Rupees
2. Rupees to Dollars
3. Exit
Enter choice:
```