Python Practical Solutions
Class XII (2025–26)

Pratyush Ranjan Maharana
Class- XII B

तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

Submitted to - Mrs.Kavita

1. Write a program to check a number whether it is prime or not.

```python
def is_prime(n):
    if n <= 1:
        return False
    if n <= 3:
        return True
    if n % 2 == 0 or n % 3 == 0:
        return False
    i = 5
    while i * i <= n:
        if n % i == 0 or n % (i + 2) == 0:
            return False
        i += 6
    return True

num = int(input("Enter a number: "))
print(f"{num} is {'prime' if is_prime(num) else 'not prime'}")
```

Output:

```
Enter a number: 29
29 is prime
```

2.Write a program to check whether it is plaindrome or not

```python
def is_palindrome(s):
    return str(s) == str(s)[::-1]

val = input("Enter a number/string: ")
print(f"'{val}' is {'a palindrome' if is_palindrome(val)
   ↪ else 'not a palindrome'}")
```

Output:

```
Enter a number/string: madam
'madam' is a palindrome
```

3. Write a program to display ASCII code of a charcater and vice versa.

```python
def ascii():
    c = input("Enter character: ")
    if len(c) == 1:
        print("ASCII ->", ord(c))
    else:
        print("Only 1 character allowed.")
def char():
    n = int(input("Enter ASCII code: "))
    if 0 <= n <= 127:
        print("Character",chr(n))
    else:
        print("Invalid ASCII code.")
ascii()
char()
```

Output:

```
Enter a character: Z
ASCII value: 90
Enter ASCII code (0-127): 97
Character: a
```

4. Write a program using function that reads a line and prints its statistics like

number of uppercase: number of lowercase: number of digits :

```
def string_stats(text):
    return {
        ’Uppercase’: sum(1 for c in text if c.isupper()),
        ’Lowercase’: sum(1 for c in text if c.islower()),
        ’Digits’: sum(1 for c in text if c.isdigit())
    }
text = input("Enter a line of text: ")
stats = string_stats(text)
print("\nStatistics:")
for k, v in stats.items():
    print(f"{k}: {v}")
```

Output:

```
Enter character: p
ASCII -> 112
Enter ASCII code: 113
Character     q
```

5. Write a program to compute GCD and LCM of two numbers .

```python
def gcd(a, b):
    while b != 0:
        a, b = b, a % b
    return a
def lcm(a, b):
    return abs(a * b) // gcd(a, b)

num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))

g = gcd(num1, num2)
l = lcm(num1, num2)

print(f"GCD of {num1} and {num2} is: {g}")
print(f"LCM of {num1} and {num2} is: {l}")
```

Output:

```
Enter first number: 2
Enter second number: 3
GCD of 2 and 3 is: 1
LCM of 2 and 3 is: 6
```

6. Write a program using the FACT() function to calculate the factorial of an integer.

```python
def FACT(n):
    if n < 0:
        return "Factorial not defined for negative numbers."
    elif n == 0 or n == 1:
        return 1
    else:
        fact = 1
        for i in range(2, n + 1):
            fact *= i
        return fact
num = int(input("Enter an integer: "))
result = FACT(num)
print(f"The factorial of {num} is: {result}")
```

Output:

```
Enter a number: 5
Factorial of 5: 120
```

7. Write a program to generate random numbers between 1 to 6 and check whether a user won a Lottery or not.

```python
import random
def roll_dice():
    return random.randint(1, 6)
user_guess = int(input("Guess a number between 1 and 6: "))
lottery_number = roll_dice()
if user_guess == lottery_number:
    print("you won")
else:
    print(f"you loser. The winning number was {lottery_number}. Try again!")
```

Output:

```
Guess a number between 1 and 6: 2
Sorry, you lost. The winning number was 4. Try again!
```

8. Vowel Counter in Text File

```python
def count_vowels(filename):
    vowels = 'aeiouAEIOU'
    with open(filename, 'r') as f:
        return sum(1 for char in f.read() if char in vowels)
with open('sample.txt', 'w') as f:
    f.write("MY NAME IS
    PRATYUSH.\n")
print("Vowel count:", count_vowels('sample.txt'))
```

Output:

```
Vowel count: 13
```

9. Filter lines containing ’p’

```python
with open('input.txt', 'w') as f:
    f.write("This is a text with some p's.\n")
    f.write("Python is powerful.\n")
    f.write("This line does not contain the letter p.\n")
    f.write("Pineapple is a fruit.\n")
with open('input.txt', 'r') as infile, open('output.txt', 'w') as outfile:
    for line in infile:
        if 'p' in line.lower():
            outfile.write(line)

print("Lines with 'p' copied to output.txt")
```

Output:

```
Lines with 'p' copied to output.txt
```

10. Write a program to count number of words in text file.

```

with open(’sample.txt’, ’w’) as f:
    f.write("TEXT IN WORDS.\n")
    f.write(" number of words present in this file \n")

def count_words(filename):
    with open(filename, ’r’) as f:
        return len(f.read().split())

print("Total words:", count_words(’sample.txt’))
```

Output:

```
Total words: 10
```

11. Write a program to append one more record in the existing text file name as student.txt

```python
def append_student_record():
    filename = "student.txt"
    print("Enter student details to append:")
    name = input("Name: ")
    roll_no = input("Roll Number: ")
    grade = input("Grade: ")

    with open(filename, ’a’) as file:
        file.write(f"{name}, {roll_no}, {grade}\n")
    print("Record appended successfully!")

append_student_record()
```

Output:

```
Enter student details to append:
Name: pratyush
Roll Number: 27
Grade: A+
Record appended successfully!
```

12. Write a python program to write student data in binary file

```python
import pickle
student = {
    "roll_no": 27,
    "name": "pratyush",
    "class": "12th",
    "marks": 99
}
with open("student.dat", "wb") as file:
    pickle.dump(student, file)
print("data saved in binary file :)")
```

Output:

```
data saved in binary file :)
```

13. Write a python program to read student data from a binary file.

```python
import pickle
students = [
    {'roll': 1, 'name': 'pratyush', 'marks': 99},
    {'roll': 2, 'name': 'rohan', 'marks': 92},
    {'roll': 3, 'name': 'rahul', 'marks': 79}
]

with open('students.dat', 'wb') as file:
    pickle.dump(students, file)
try:
    with open('students.dat', 'rb') as file:
        loaded_students = pickle.load(file)
        print("Student Records:")
        for student in loaded_students:
            print(f"Roll No: {student['roll']}, Name: {student['name']}, Marks: {student['marks']}")
except FileNotFoundError:
    print("The file was not found.")
except EOFError:
    print("The file is empty.")
except Exception as e:
    print("An error occurred:", e)
```

Output:

```
Student Records:
Roll No: 1, Name: pratyush, Marks: 99
Roll No: 2, Name: rohan, Marks: 92
Roll No: 3, Name: rahul, Marks: 79
```

14. Write a python program to modify/update student data in a binary file.

```python
import pickle
try:
    with open('students.dat', 'rb') as f:
        data = pickle.load(f)
except:
    data = []
print("\nBefore:", *data, sep="\n")
roll = int(input("\nRoll to update: "))
for s in data:
    if s['roll'] == roll:
        s['name'] = input("New name: ")
        s['marks'] = int(input("New marks: "))
        break
else:
    print("Not found."); exit()
with open('students.dat', 'wb') as f:
    pickle.dump(data, f)
print("\nAfter:", *data, sep="\n")
```

Output:

```
 Student Data Before Update
{'roll': 1, 'name': 'pratyush', 'marks': 99}
{'roll': 2, 'name': 'paaji', 'marks': 0}
{'roll': 3, 'name': 'rahul', 'marks': 79}
Enter roll number to update: 2
Enter new name: rohan
Enter new marks: 99
Student Data After Update
{'roll': 1, 'name': 'pratyush', 'marks': 99}
{'roll': 2, 'name': 'rohan', 'marks': 99}
{'roll': 3, 'name': 'rahul', 'marks': 79}
```

15. Write a python program to delete student data from a binary file.

```python
import pickle
try:
    with open('students.dat', 'rb') as f:
        data = pickle.load(f)
except:
    data = []
print("\nBefore:", *data, sep="\n")
roll = int(input("\nRoll to delete: "))
deleted = [s for s in data if s['roll'] == roll]
if deleted:
    data = [s for s in data if s['roll'] != roll]
    with open('students.dat', 'wb') as f:
        pickle.dump(data, f)
    print(f"\nDeleted: {deleted[0]}")
else:
    print("\nNot found.")
print("\nAfter:", *data, sep="\n")
```

Output:

```

Before:
{'roll': 1, 'name': 'ok', 'marks': 2}
{'roll': 2, 'name': 'pratyush', 'marks': 99}
{'roll': 3, 'name': 'rahul', 'marks': 79}
Roll to delete: 1
Deleted: {'roll': 1, 'name': 'ok', 'marks': 2}

After:
{'roll': 2, 'name': 'pratyush', 'marks': 99}
{'roll': 3, 'name': 'rahul', 'marks': 79}
```

16. Write a python program to search a student record in a binary file.

```python
import pickle
try:
    with open('students.dat', 'rb') as f:
        students = pickle.load(f)
except:
    students = []
roll = int(input("Enter Roll Number to Search: "))
record = next((s for s in students if s['roll'] == roll), None)

if record:
    print("\nStudent Found:")
    print(f"Roll: {record['roll']}\nName: {record['name'].title()}\nMarks: {record['marks']}")
else:
    print("\nNo student found with that roll number.")
```

Output:

```
Enter Roll Number to Search: 2
Student Found:
Roll: 2
Name: Pratyush
Marks: 99
```

17. Create a binary file with name and roll number.Search for a given roll number and display the name,if not found display appropriate message.

```python
import pickle
def file(filename):
    with open(filename, 'wb') as f:
        n = int(input("Enter number of students: "))
        for _ in range(n):
            name = input("Enter name: ")
            roll = int(input("Enter roll number: "))
            data = {'name': name, 'roll': roll}
            pickle.dump(data, f)
    print("File created successfully.\n")
def search_roll(filename, roll_to_search):
    found = False
    try:
        with open(filename, 'rb') as f:
            while True:
                try:
                    data = pickle.load(f)
                    if data['roll'] == roll_to_search:
                        print(f"Roll Number: {roll_to_search}, Name: {data['name']}")
                        found = True
                        break
                except EOFError:
                    break
        if not found:
            print("Roll number not found.")
    except FileNotFoundError:
        print("File does not exist.")

filename = "students.dat"
file(filename)
roll = int(input("Enter roll number to search: "))
search_roll(filename, roll)
```

Output:

```
Enter number of students: 1
Enter name: pratyush
Enter roll number: 27
File created successfully.
Enter roll number to search: 27
Roll Number: 27, Name: pratyush
#if rollno doesn't exist in file then:-
Enter roll number to search: 1
Roll number not found.
```

18. Create a binary file with roll number,name,and marks. Input a roll number and update the marks

```python
import pickle

def create_file(fname):
    with open(fname, 'wb') as f:
        for _ in range(int(input("Number of students: "))):
            data = {'roll': int(input("Roll: ")), 'name': input("Name: "), 'marks': float(input("Marks: "))}
            pickle.dump(data, f)
def update_marks(fname, roll):
    students, found = [], False
    with open(fname, 'rb') as f:
        try:
            while True: students.append(pickle.load(f))
        except EOFError: pass
    for s in students:
        if s['roll'] == roll:
            s['marks'] = float(input(f"Current marks: {s['marks']}. Enter new marks: "))
            found = True
            break
    if found:
        with open(fname, 'wb') as f:
            for s in students: pickle.dump(s, f)
        print("Marks updated successfully.")
    else:
        print("Roll number not found.")
file = 'students.dat'
create_file(file)
update_marks(file, int(input("Enter roll to update marks: ")))
```

Output:

```
Number of students: 1
Roll: 27
Name: pratyush
Marks: 99
Enter roll to update marks: 2
Roll number not found.
```

19. Write a program to perform read and write operation with csv.file

```
import csv
def write():
  file_name="meow.csv"
  data=[
      ["roll","name","marks"],  #data is written to meow.csv
      [1 ,'pratyush', 99],
      [2 ,'rohan', 98],
      [3 ,'piyush', 97],
  ]
  with open(file_name,mode='w',newline='') as f:
    writer=csv.writer(f)
    writer.writerows(data)
  print("data is written to",file_name)

def read():
    file_name='meow.csv'
    print("\nReading data from",file_name)
    with open(file_name,mode='r') as f:
      reader=csv.reader(f)
      for row in reader:
        print(row)
write()
read()
```

Output:

```
data is written to meow.csv

Reading data from meow.csv
['roll', 'name', 'marks']
['1', 'pratyush', '99']
['2', 'rohan', '98']
['3', 'piyush', '97']
```