- Question 1. Python program to check whether a given number is even or odd using recursion

Answer . def is_even(n):

```
    if n == 0:  # base case
        return True
    elif n == 1:  # base case
        return False
    else:
        return is_even(n - 2)  # recursive call


# example usage
print(is_even(4))  # True
print(is_even(5))  # False
```

- question 2. Python program to check whether a number is positive or negative

Answer. num = float(input("Enter a number: "))

```
if num > 0:
    print("The number is positive.")
elif num == 0:
    print("The number is zero.")
else:
    print("The number is negative.")
```

- question 3. Python program to check if a number is a palindrome

answer . num = input("Enter a number: ")

```
if num == num[::-1]:
    print("The number is a palindrome.")
else:
    print("The number is not a palindrome.")
```

- question 4. Python program to reverse a number

answer. num = int(input("Enter a number: "))

```
rev = 0


while num > 0:
    digit = num % 10
    rev = rev * 10 + digit
    num //= 10


print("The reversed number is:", rev)
```

- question5 . PYTHON PROGRAM TO PRINT ALL INTEGERS THAT AREN'T DIVISIBLE BY EITHER 2 OR 3

answer.
```
for i in range(1, 101):
    if i % 2 != 0 and i % 3 != 0:
        print(i)
```

- question6. PYTHON PROGRAM TO FIND SUM OF DIGIT OF A NUMBER WITHOUT RECURSION

answer.
```
num = int(input("Enter a number: "))
sum_of_digits = 0


while num > 0:
    digit = num % 10
    sum_of_digits += digit
    num //= 10


print("The sum of digits is:", sum_of_digits)
```

- question7. PYTHON PROGRAM TO PRINT TABLE OF A GIVEN NUMBER

answer.
```
num = int(input("Enter a number: "))


for i in range(1, 11):
    print(num, "x", i, "=", num * i)
```

- question8. PYTHON PROGRAM TO READ A NUMBER N AND COMPUTE N+NN+NNN

Answer. n = int(input("Enter a number: "))

nn = n * 11

nnn = n * 111

result = n + nn + nnn

print("The result is:", result)

- question9. PYTHON PROGRAM TO CHECK IF A NUMBER IS A STRONG NUMBER

answer. num = int(input("Enter a number: "))

Find the factorial of each digit and sum the results

sum_of_factorials = 0

for digit in str(num):

  factorial = 1

  for i in range(1, int(digit) + 1):

    factorial *= i

  sum_of_factorials += factorial

Check if the sum of factorials equals the input number

if sum_of_factorials == num:

  print(num, "is a strong number")

else:

  print(num, "is not a strong number")

- question 10. PYTHON PROGRAM TO PRINT NUMBERS IN A RANGE WITHOUT USING LOOPS

answer. start = int(input("Enter the starting number: "))

end = int(input("Enter the ending number: "))

numbers = list(range(start, end + 1))

```
print(*numbers, sep='\n')
```

- question 11. PYTHON PROGRAM TO CHECK IF TWO NUMBERS ARE AMICABLE NUMBERS OR NOT

answer.
```
def sum_of_divisors(n):
    return sum(i for i in range(1, n) if n % i == 0)

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

sum1 = sum_of_divisors(num1)
sum2 = sum_of_divisors(num2)

if sum1 == num2 and sum2 == num1:
    print(num1, "and", num2, "are amicable numbers")
else:
    print(num1, "and", num2, "are not amicable numbers")
```

- question 12. PYTHON PROGRAM TO FIND WHETHER A NUMBER IS A POWER OF TWO

answer.
```
def is_power_of_two(n):
    if n <= 0:
        return False
    return (n & (n - 1)) == 0

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

if is_power_of_two(num):
    print(num, "is a power of two")
else:
    print(num, "is not a power of two")
```

- question 13. PYTHON PROGRAM TO FIND PRODUCT OF TWO NUMBERS USING RECURSION

answer.
```
def multiply(x, y):
    if y == 0:
        return 0
    elif y > 0:
        return x + multiply(x, y-1)
    else:
        return -multiply(x, -y)


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


result = multiply(num1, num2)


print("The product of", num1, "and", num2, "is", result)
```

- question . 14 PYTHON PROGRAM TO FIND ALL PERFECT SQUARES IN THE GIVEN RANGE

answer.
```
import math


def find_perfect_squares(start, end):
    perfect_squares = []
    for i in range(start, end+1):
        sqrt = math.sqrt(i)
        if sqrt == int(sqrt):
            perfect_squares.append(i)
    return perfect_squares


start = int(input("Enter the starting number: "))
end = int(input("Enter the ending number: "))
```

```
perfect_squares = find_perfect_squares(start, end)


print("The perfect squares between", start, "and", end, "are:", perfect_squares)
```

- question. 15 PYTHON PROGRAM TO PRINT ALL POSSIBLE COMBINATIONS OF THREE DIGITS

answer.
```
for i in range(1, 10):
    for j in range(0, 10):
        for k in range(0, 10):
            if i != j and j != k and i != k:
                print(i, j, k)
```

- question 16. PYTHON PROGRAM TO FIND FIBONACCI NUMBERS USING RECURSION

answer.
```
def fibonacci(n):
    if n <= 1:
        return n
    else:
        return fibonacci(n-1) + fibonacci(n-2)


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


if n <= 0:
    print("Please enter a positive integer.")
else:
    print("The", n, "th Fibonacci number is:", fibonacci(n))
```

- question 17. PYTHON PROGRAM TO FIND THE FIBONACCI SERIES WITHOUT USING RECURSION

answer.
```
n = int(input("Enter a number: "))


# initialize the first two terms of the series
a, b = 0, 1
```

```
if n <= 0:
   print("Please enter a positive integer.")
elif n == 1:
   print("Fibonacci sequence up to", n, ":")
   print(a)
else:
   print("Fibonacci sequence up to", n, ":")
   print(a)
   print(b)
   for i in range(2, n):
      c = a + b
      a, b = b, c
      if c <= n:
         print(c)
      else:
         break
```

- question 19. Python program to find the factorial of a number using recursion
- Answer. def factorial(n):
- if n == 1:
- return 1
- else:
- return n * factorial(n-1)
-
- num = int(input("Enter a number: "))
-
- if num < 0:
- print("Factorial does not exist for negative numbers.")
- elif num == 0:
- print("Factorial of 0 is 1.")
- else:
- print("Factorial of", num, "is", factorial(num))

- Question 20. Python program to find the factorial of a number without recursion

Answer, num = int(input("Enter a number: "))

```
factorial = 1

if num < 0:
   print("Factorial does not exist for negative numbers.")
elif num == 0:
   print("Factorial of 0 is 1.")
else:
   for i in range(1, num+1):
      factorial *= i
   print("Factorial of", num, "is", factorial)
```

- Question 21. Python program to check if a string is a pangram or not

Answer. import string

```
def is_pangram(str):
   alphabet = set(string.ascii_lowercase)
   return alphabet <= set(str.lower())

# example usage
input_str = "The quick brown fox jumps over the lazy dog"
if is_pangram(input_str):
   print("The string is a pangram")
else:
   print("The string is not a pangram")
```

- Question 22. Python program to remove odd indexed characters in a string

Answer. def remove_odd_index_chars(input_str):

```
   new_str = ""
   for i in range(len(input_str)):
      if i % 2 == 0:
         new_str += input_str[i]
   return new_str


input_str = "Hello, World!"
new_str = remove_odd_index_chars(input_str)
```

```
print(new_str)
```

- Question 23. Python program to remove the nth index character from a non-empty string

Answer.
```
def remove_nth_char(input_str, n):
    return input_str[:n] + input_str[n+1:]


input_str = "Hello, World!"
n = 7
new_str = remove_nth_char(input_str, n)
print(new_str)
```