	SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGG.	**LABORATORY MANUAL**
	PRACTICAL EXPERIMENT INSTRUCTION SHEET	
	EXPERIMENT TITLE: Write a Python script that prints prime numbers less than 20.	
EXPERIMENT NO. : **SSGMCE/WI/IT/01/3IT09/01**	ISSUE NO. : 00	ISSUE DATE : 30.07.2023
REV. DATE :	REV. NO. :	DEPTT. : INFORMATION TECHNOLOGY
LABORATORY : 3IT09 COMPUTER SKILL LAB – I	SEMESTER : III	PAGE: 1 OF 2

1.0) AIM: Write a Python script that prints prime numbers less than 20..

2.0) SCOPE: The scope of this Python script is to demonstrate the concept of prime numbers and the use of functions and loops to find and print prime numbers less than 20. This script is intended for educational purposes to help students understand the implementation of prime number detection.

3.0) FACILITIES/ APPARATUS:

1. Python development environment (e.g., IDLE)
2. Input mechanism (keyboard)
3. Computer with Python installed

4.0) THEORY:

Program Description:

Program Description:

1. Function for Prime Check (is_prime): The script begins by defining a function named is_prime(num) that checks whether a given number num is prime or not. A prime number is a positive integer greater than 1 that is divisible only by 1 and itself. The function returns True for prime numbers and False for non-prime numbers.

2. Prime Number Detection: The is_prime function employs several techniques to efficiently determine whether a number is prime or not:

    - It handles special cases: numbers less than or equal to 1 are not prime, and 2 is the only even prime number.
    - For odd numbers greater than 2, it uses a loop to check if the number is divisible by any odd integer from 3 up to the square root of the number. If any divisor is found, the function returns False. Otherwise, it returns True.

3. Iterating Through Numbers: The script utilizes a for loop to iterate through numbers from 2 to 19 (less than 20), as the objective is to find prime numbers less than 20.

4. Printing Prime Numbers: Within the loop, each number is passed to the is_prime() function to check if it is prime. If the number is prime, it is printed to the console.

Example:

Suppose we have the following code:

```
print("Prime numbers less than 20:")
for number in range(2, 20):
    if is_prime(number):
        print(number, end=' ')
```

Output

```
Prime numbers less than 20:
2 3 5 7 11 13 17 19
```

Program

```python
# Function to check if a number is prime
def is_prime(num):
    if num <= 1:
        return False
    elif num == 2:
        return True
    elif num % 2 == 0:
        return False
    else:
        for i in range(3, int(num**0.5) + 1, 2):
            if num % i == 0:
                return False
        return True

# Print prime numbers less than 20
print("Prime numbers less than 20:")
for number in range(2, 20):
    if is_prime(number):
        print(number, end=' ')
```

Output

```
Prime numbers less than 20:
2 3 5 7 11 13 17 19
```

In this script:

We define a function is_prime(num) that checks whether a given number num is prime or not. It returns True for prime numbers and False otherwise.

Inside the function:

- We handle special cases: numbers less than or equal to 1 are not prime, and 2 is the only even prime number.
- For odd numbers greater than 2, we use a loop to check if the number is divisible by any odd integer from 3 up to the square root of the number. If it is divisible, we return False. Otherwise, we return True.
- We then use a for loop to iterate through numbers from 2 to 19 (less than 20) and check if each number is prime using the is_prime() function.
- If a number is prime, it is printed to the console.
- This script efficiently prints prime numbers less than 20, demonstrating the use of functions and loops in Python.

4.2) Program Execution:

Program Execution Line by Line:

- Lines 4-20: We define a function named is_prime(num) that checks whether a given number num is prime using the logic described above. This function can be reused throughout the script.
- Lines 23-24: We print a header message indicating that we are going to display prime numbers less than 20.
- Lines 25-27: We use a for loop to iterate through numbers from 2 to 19 (less than 20).
- Line 26: For each number in the specified range, we check if it is prime by calling the is_prime() function.
- Line 27: If the number is prime, it is printed to the console.

	SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGG.	**LABORATORY MANUAL**
	PRACTICAL EXPERIMENT INSTRUCTION SHEET	
	EXPERIMENT TITLE: Write a Python script that prints prime numbers less than 20.	
EXPERIMENT NO. : **SSGMCE/WI/IT/01/3IT09/01**	ISSUE NO. : 00	ISSUE DATE : 30.07.2023
REV. DATE :	REV. NO. :	DEPTT. : INFORMATION TECHNOLOGY
LABORATORY : 3IT09 COMPUTER SKILL LAB – I	SEMESTER : III	PAGE: 5 OF 2

5.0) Conclusion:

In conclusion, this Python script provides an educational example of how to detect and print prime numbers less than 20 using a function and a **for** loop. Students can learn about the concept of prime numbers, the use of functions to encapsulate logic, and the implementation of loops for repetitive tasks. This exercise promotes understanding of both mathematical concepts and programming techniques.