**Program:1**

```
For num in range(1500, 2700):
   If num % 7 == 0 and num % 5 == 0:
      Print(num)
```

**Program:2**

```
Def temperature_converter():
   Print("1.  From Celsius to Fahrenheit")
   Print("2.  From Fahrenheit to Celsius")
   Option = int(input("Enter your choice: "))
   If option in [1, 2]:
      Temp = float(input("Enter temperature: "))
      If option == 1:
         F_temp = (temp * 9) / 5 + 32
         Print("Temperature in Fahrenheit is: ", f_temp)
      Else:
         C_temp = (temp – 32) * 5 / 9
         Print("Temperature in Celsius is: ", c_temp)
   Else:
      Print("Invalid choice. Try Again!")
      Temperature_converter()
Temperature_converter()
```

**Program:3**

```
Import random
Number_to_guess = random.randint(1, 10)
Guess = int(input("Guess a Number b/w 1 & 10: "))
While guess != number_to_guess:
   Print("Wrong guess. Try Again!")
```

```
   Guess = int(input("Guess a Number b/w 1 & 10: "))
Print("Correct guess! The number is ", number_to_guess, ".")
```

**Program:4**

```
For I in range(1, 6):
   For j in range(1, I + 1):
      Print("*", end=' ')
   Print()
For I in range(4, 0, -1):
   For j in range(1, I + 1):
      Print("*", end=' ')
   Print()
```

**Program:5**

```
Input_word = input("Enter a Word: ")
Reversed_word = ""
For char in input_word:
   Reversed_word = char + reversed_word
Print(reversed_word)
```

**Program:6**

```
Print("Series of Numbers: (1,2,3,4,5,6,7,8,9)")
Even_count = 0
Odd_count = 0
For num in range(1, 10):
   If num % 2 == 0:
      Even_count += 1
   Else:
      Odd_count += 1
```

```
Print("Number of Even Numbers: ", even_count)
Print("Number of Odd Numbers: ", odd_count)
```

**Program:7**

```
Print("Data_list is a given list in which:")
Data_list = [1452, 11.23, 1 + 2j, True, 'w3resource', (0, -1), [5, 12], {"class": 'V', "section": 'A'}]
For item in data_list:
   Print(item, " is an item and its type is ", type(item))
```

**Program:8**

```
Print("Numbers from 0 to 6 except 3 and 6 are:")
For num in range(0, 7):
   If num == 3 or num == 6:
      Continue
   Print(num)
```

**Program:9**

```
A = 0
B = 1
C = a + b

Print(a, end=',')
Print(b, end=',')
Print(c, end=',')

Next_val = 0
While next_val < 51:
   Next_val = b + c
   B = c
```

```
    C = next_val
    Print(next_val, end=',')
    A = a + 1
```

**Program:10**

```
For num in range(1, 51):
    If num % 3 == 0 and num % 5 == 0:
        Print("FizzBuzz")
    Elif num % 3 == 0:
        Print("Fizz")
    Elif num % 5 == 0:
        Print("Buzz")
    Else:
        Print(num)
```

**Program:11**

```
Rows = int(input("Enter no. of rows:"))
Columns = int(input("Enter no. of columns:"))

I = 0
While I < rows:
    J = 0
    While j < columns:
        Print(I * j, end=',')
        J = j + 1
    Print()
    I = I + 1
```

**Program:12**

```
Binary_input = input("Enter 4 digit binary numbers separated by comma:")
Binary_numbers = binary_input.split(',')
Divisible_by_5 = []

For binary in binary_numbers:
   Decimal_value = int(binary, base=2)
   If decimal_value % 5 == 0:
      Divisible_by_5.append(binary)

Print(','.join(divisible_by_5))
```

**Program:13**

```
Input_string = input("Enter a string:")

Num_count = 0
Letter_count = 0

For char in input_string:
   If char.isdigit():
      Num_count = num_count + 1
   Elif char.isalpha():
      Letter_count = letter_count + 1

Print("Number of Letters:", letter_count, "and Number of Digits:", num_count)
```

**Program:14**

```
Is_valid = False
While not is_valid:
```

```
Password = input(“Enter Password:”)
If len(password) < 6:
   Print(“Password must contain at least 6 characters.”)
Elif len(password) > 16:
   Print(“Password must contain less than 16 characters.”)
Elif not any(char.isdigit() for char in password):
   Print(“There must be at least one digit in the password.”)
Elif not any(char.islower() for char in password):
   Print(“There must be at least one lower-case letter in the password.”)
Elif not any(char.isupper() for char in password):
   Print(“There must be at least one upper-case letter in the password.”)
Elif not any(char == ‘@’ for char in password):
   Print(“There must be at least one special character i.e [ @ or _ ] in the password.”)
Else:
   Print(“Password is valid.”)
   Is_valid = True
```