An electric power distribution company charges its domestic consumers as follows:
0-100 Rs. 1 per unit
101-300 Rs. 100 plus Rs. 1.25 per unit in excess of 100
301-500 Rs. 350 plus Rs. 1.50 per unit in excess of 300
500 and above Rs. 650 plus 1.75 per unit in excess of 500

Write a program that read the customer number & power consumed and prints the amount to be paid by the customer. Note that output should be well formatted.

***Title**: Generate an Electricity for a customer based on units consumed.*

***Objective**: To calculate and display the electricity bill for a customer based on the power consumed*

***Task**: Write a program that reads the customer number and power consumption, then computes and prints the amount to be paid by the customer, ensuring the output is well-formatted.*

Code:

```
#Input Data
cno=int(input("Enter Cusumer Number:"))
pc=int(input("Enter power consumed:"))
#Computing bill amount based on power consumed
if pc>0 and pc<=100:
   bill_amt=pc*1
elif pc>100 and pc<=300:
   bill_amt=100+(pc-100)*1.25
elif pc>300 and pc<500:
   bill_amt=350+(pc-300)*1.50
elif pc>500:
   bill_amt=650+(pc-500)*1.75
else:
   print("Invalid Power Consumed Units")
#Printing the bill in proper format
print("~"*60)
print("\t\tABC Power Company Ltd.")
print("~"*60)
print("Consumer Number:",cno)
print("Consumed Units:",pc)
print("------------------------------")
print("Bill Amount:",bill_amt)
```

Write a program to check whether given number is Armstrong or not

***Objective**: "To determine whether a given number is an Armstrong number, based on its digits and their powers."*

***Task**: "Write a program that checks if a given number is an Armstrong number, and prints the result."*

***Title**: "Check if a Given Number is an Armstrong Number"*

Code:

```
# Take input from the user
number = int(input("Enter a number to check: "))

# Step 1: Find the number of digits in the number
# Convert the number to a string to count the digits
num_str = str(number)
num_digits = len(num_str)

# Step 2: Calculate the sum of each digit raised to the power of num_digits
sum_of_powers = 0
for digit in num_str:
    sum_of_powers += int(digit) ** num_digits

# Step 3: Check if the sum is equal to the original number
if sum_of_powers == number:
    print(f"{number} is an Armstrong number")
else:
    print(f"{number} is not an Armstrong number")
```

Write a program to print a multiplication table of a given number

***Objective**: "To generate and display the multiplication table of a given number."*

***Task**: "Write a program that takes a number as input and prints its multiplication table."*

***Title**: "Print Multiplication Table of a Given Number"*

Code:

```
#Take input to accept a number for printing Multiplication table
```

```
n=int(input("Enter number to print multiplication table:"))
#Take for loop for multiple
i=1
while (i<=10):
   print(n," x ", i, " = ", n*i )
   i=i+1
```

Write a program to generate the following pattern:

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15

***Objective**: "To generate and display a number pattern where each row contains consecutive integers starting from 1."*

***Task**: "Write a program that prints the following number pattern, with each row having one more number than the previous one."*

***Title**: "Generate a Number Pattern with Consecutive Integers"*

Code:

```
#Take input for n lines
n=int(input("Enter n:"))
#Generating Pattern
k=1
for i in range(1,n+1):
   for j in range(1,i+1):
      print(k,end=" ")
      k=k+1
   print()
```

Write a program to create a list of students' marks with user-defined values and find the maximum

***Objective**: "To create a list of students' marks based on user input and find the maximum mark."*

***Task**: "Write a program that allows the user to input a list of students' marks and then displays the highest mark."*

***Title**: "Find the Maximum Mark from a List of Students' Marks"*

Code:

```
#Take input for n lines
n=int(input("Enter no. of subjects:"))
#Creating empty list
l=[]
#Accepting marks and appending marks into the list
for i in range(n):
   m=int(input("Enter marks:"))
   l.append(m)
print("Maximum marks scored:",max(l))
```

Python program that checks if a given string is a palindrome

***Objective**: "To check whether a given string is a palindrome, i.e., it reads the same forwards and backwards."*

***Task**: "Write a program that takes a string as input and checks if it is a palindrome, then displays the result."*

***Title**: "Check if a Given String is a Palindrome"*

***Code**:*

```
# Take input from the user
string = input("Enter a string to check: ")

# Step 1: Remove spaces and convert the string to lowercase to handle case insensitivity
string = string.replace(" ", "").lower()

# Step 2: Reverse the string
reversed_string = string[::-1]

# Step 3: Check if the string is equal to its reverse
if string == reversed_string:
   #print(f"'{string}' is a palindrome.")
   print(string, "- is a palindrome.")
else:
   #print(f"'{string}' is not a palindrome.")
   print(string, "- is a pliandrom.")
```

Write a program to count the frequency of every element in a given list

***Objective**: "To count and display the frequency of each element in a given list."*

***Task**: "Write a program that takes a list as input and calculates the frequency of every element, then prints the result."*

***Title**: "Count the Frequency of Elements in a Given List"*

***Code**:*

```
#Creating empty list
l = []
#Take input for n no. of elements
n=int(input("Enter the no. of elements:"))
#Append the values into the list
for i in range(n):
    val=int(input("Enter value "+str(i+1)+":"))
    l.append(val)
#Decalring a dictionary object to store the data
f = {}
for i in l:
    if (i in f):
        f[i] += 1
    else:
        f[i] = 1
#Displaying the data
for i, j in f.items():
    print(i, "->", j)
```

Check whether given number is Prime or not

***Objective**: "To determine whether a given number is a prime number or not."*

***Task**: "Write a program that checks if a given number is prime and displays the result."*

***Title**: "Check Whether a Given Number is Prime or Not"*

***Code**:*

```
# Program to check if a number is prime or not
num = int(input("Enter Number:"))
```

```
# To take input from the user
#num = int(input("Enter a number: "))

# define a flag variable
flag = False

if num == 0 or num == 1:
   print(num, "is not a prime number")
elif num > 1:
   # check for factors
   for i in range(2, num):
      if (num % i) == 0:
         # if factor is found, set flag to True
         flag = True
         # break out of loop
         break
   # check if flag is True
   if flag:
      print(num, "is not a prime number")
   else:
      print(num, "is a prime number")
```