



## ANSWERS

### PYTHON LOGICAL ASSIGNMENT – 1

#### 1) A Python Program to print half pyramid using \*

**Solution:**

```
# Way 1:

n = int(input("Enter the number of rows"))      # gives row from user
for i in range(1, n+1):                          # i represent the number of column
    for j in range(1, i+1):                      # j represent the number of row
        print(" * ", end=" ")                  # print *
    print()

# Way 2:

n = int(input("Enter the number of rows"))
for i in range(1, n+1):                          #using single loop gives range start from 1
                                                # and end from end+1 // range gives the
    print(" * " * i)
```

#### 2) A Python Program to Swap two Variables:.

**Solution:**

```
# Way 1:

no1 = int(input("Enter the First number"))
no2 = int(input("Enter the Second number"))

print("Before Swapping Number: ", no1, no2)

no1, no2 = no2, no1

print("After Swapping Number: ", no1, no2)
```



```
# Way 2:

no1 = int(input("Enter the First number"))
no2 = int(input("Enter the Second number"))

print("Before Swapping Number: ", no1, no2)

# create temporary variable and swap the values
temp = no1
no1 = no2
no2 = temp

print("After Swapping Number: ", no1, no2)
```

### 3) A Python Program to Check Whether Number is Prime or Not. Solution:

```
# Way 1:

num = int(input("Enter the number.."))
# prime number are greater than 1
if num > 1:
    # check for factors
    for i in range(2, num):
        if (num % i) == 0:
            print(num, "is not a prime number")
            break
    else:
        print(num, "is a prime number...")

# if number is less than
# or equal to 1, it is not prime
else:
    print(num, "is not a prime number")

# Way 2:

num = int(input("Enter the number.."))
# prime number are greater than 1
if num > 1:
```



```
# Iterate from 2 to n/2
for i in range(2, num // 2):

    # if number is divisible by a any number between
    # 2 and n/2, it is not prime number

    if (num % i) == 0:
        print(num, "is not a prime number")
        break
    else:
        print(num, "is a prime number...")

# if number is less than
# or equal to 1, it is not prime
else:
    print(num, "is not a prime number")

# Way 3:

k = 0
num = int(input("Enter the number.."))
for i in range(1, num+1):
    if num % i == 0:
        k += 1
if k == 2:
    print(num, "is a prime number")
else:
    print(num, "is not a prime number")
```

#### 4) A Python Program to Print the Fibonacci Sequence.

##### Solution:

```
# Way 1:

num = int(input("How many Range?"))
n1 = 0
n2 = 1
count = 0

if num <= 0:
```



```
    print("Print Positive Number")
elif num == 1:
    print("Print Fibonacci sequence upto", num, ":")
    print(n1)
else:
    print("Fibonacci Sequence: ")
    while count < num:
        print(n1)
        temp = n1 + n2
        n1 = n2
        n2 = temp
        count += 1

# Way 2:

num = int(input("How many Range of Number?"))
i = 0
n1 = 0
n2 = 1
while i < num:
    if i <= 1:
        Next = i
    else:
        Next = n1 + n2
        n1 = n2
        n2 = Next
    print(Next)
    i += 1

# Way 3:

num = int(input("How many Range of Number?"))
x, y = 0, 1
while y < num:
    print(y)
    x, y = y, x+y
```

**5) A Python Program to Check Whether Number is Armstrong or Not.**  
**Solution:**

```
num = int(input("Enter the number.."))
```



```
# Initialize Sum
sum = 0

temp = num
while temp > 0:
    digit = temp % 10
    sum += digit ** 3
    temp //= 10

if num == sum:
    print(num, "is an Armstrong number")
else:
    print(num, "is not an Armstrong number")

# Way 2:

from math import*
num = int(input("Enter the number.."))
# Initialize Sum
result = 0
n = 0
temp = num
while temp != 0:
    temp = int(temp/10)
    n += 1

temp = num
while temp != 0:
    remainder = temp % 10
    result = result + pow(remainder, n)
    temp = int(temp/10)

if result == num:
    print(num, "is an Armstrong number")
else:
    print(num, "is not an Armstrong number")
```

## 6) A Python Program to the Factors of a Number.

**Solution:**



```
num = int(input("Enter a Number..."))
factors = []
for i in range(1, num+1):
    if num % i == 0:
        factors.append(i)
print("Factors of {} = {}".format(num, factors))
```

### 7) A Python Program Check Whether Number is Palindrome or Not.

**Solution:**

```
num = int(input("Enter the Number..."))
temp = num
rev = 0
while num > 0:
    dig = num % 10
    rev = rev * 10 + dig
    num = num // 10
if temp == rev:
    print(temp, "is an Palindrome!!")
else:
    print(temp, "is not an Palindrome!!")
```

### 8) A Python Program to Check Whether Number is Happy or Not.

**Solution:**

```
num = int(input("Enter a number: "))
result = num
while (result != 1 and result != 4):
    digit = sum = 0
    while (result > 0):
        digit = result % 10
        sum = sum + (digit * digit)
        result = result // 10
    result = sum
```



```
if (result == 1):  
    print(num, " is a Happy Number!!!")  
else:  
    print(num, " is an Unhappy Number!!!")
```

### 9) A Python Program To Print String in Reverse.

#### Solution:

```
text = input("Enter the string...!!!")  
reverse = text[::-1]  
print("Reverse String is: ", reverse)
```