Question 1:
Write a python program to generate a Pascal Triangle using numpy.

Program:

```python
def pascal(m,n): return 1 if (n == 0) or (n == m) else pascal(m-1,n-1) + pascal(m-1, n)
N = int(input("Enter the number of lines: "))
for m in range(N):
    triangle = []
    for n in range(m+1): triangle.append(pascal(m,n))
    print(triangle)
```

Output:

```
Enter the number of lines: 5
[1]
[1, 1]
[1, 2, 1]
[1, 3, 3, 1]
[1, 4, 6, 4, 1]
```

Question 2:
Write a python program to generate a dataframe with the values given by the user in column "a", column "b" having the squares of column "a" and column "c" having the cubes of column "a".

Program:

```python
import pandas as pd
N = input("Enter numbers: ").split()
data = {'a': [int(n) for n in N], 'b': [int(n)**2 for n in N], 'c': [int(n)**3 for n in N]}
print("DataFrame: \n{}".format(pd.DataFrame(data)))
```

Output:

```
Enter numbers: 1 2 3 4
DataFrame:
   a   b   c
0  1   1   1
1  2   4   8
2  3   9  27
3  4  16  64
```

Question 3:
Write a python program to solve a row of a sudoku puzzle with fail safe mechanism like duplicate entries, values other than [0-9].

Program:

```python
print("Enter 9 elements (1-9) \nFor missing value Enter 0 \n")
count = 0; values = []; mValue = []

while (count < 9):
    data = input("Element {}/9 -> Enter numbers: ".format(count))
    for n in data.split():
        if int(n) not in values and int(n) <= 9 or int(n) == 0: values.append(int(n)); count += 1
    print("Current Data: {}".format(values))

print("\nGiven Data: {}".format(values))

for i in range(1,10):
    if i not in values: mValue.append(i)
print("Missing Value: {}".format(mValue))

for m in mValue: nval = [m if n == 0 else n for n in values]
print("After filling the Missing value: {}".format(nval))
```

Output:

```
Enter 9 elements (1-9)
For missing value Enter 0

Element 0/9 -> Enter numbers: 9 8 7 3 2 1 4 0 6
Current Data: [9, 8, 7, 3, 2, 1, 4, 0, 6]

Given Data: [9, 8, 7, 3, 2, 1, 4, 0, 6]
Missing Value: [5]
After filling the Missing value: [9, 8, 7, 3, 2, 1, 4, 5, 6]
```