14. ILLUSTRATION OF STACK PROGRAMMING USING LIST OF INTEGERS – I

Develop a program to implement the following stack operation in python using list of integers according to the user's choice

1. Push an integer to the stack
2. Pop integer from the stack
3. Display the stack
4. Exit

Source Code

```
stack = [] # stack is a global variable

def push(num):
    stack.append(num)
    print(num, "was pushed to the stack")

def pop():
    if stack != []:
        num = stack.pop()
        print(num, "was popped from the stack")
    else:
        print("Stack Underflow")
        print("There are no items in the stack")

def showStack():
    print("Stack:")
    print("    ", stack)


while True:
    print("============================================")
    print("What would you like to do?")
    print("""
    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit
    """)
```

```
    ch = input("Enter your choice[1/2/3/4]: ")

    if ch == "1":
        inp = int(input("Enter number to push to stack: "))
        push(inp)

    elif ch == "2":
        pop()

    elif ch == "3":
        showStack()

    elif ch == "4":
        print("[ Exiting ]") # Break from the loop to exit
        break

    else:
        print("[ Invalid Choice ]") # In case user inputs a choice that was n
```

OUTPUT

```
==============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 1
Enter number to push to stack: 23
23 was pushed to the stack
==============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 1
Enter number to push to stack: 32
32 was pushed to the stack
==============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
```

```
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 1
Enter number to push to stack: 43
43 was pushed to the stack
============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 1
Enter number to push to stack: 55
55 was pushed to the stack
============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 3
Stack:
     [23, 32, 43, 55]
============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 2
55 was popped from the stack
============================================
What would you like to do?

    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 3
Stack:
     [23, 32, 43]
============================================
What would you like to do?
```

```
    [1] Push an integer to the stack
    [2] Pop integer from the stack
    [3] Display the stack
    [4] Exit

Enter your choice[1/2/3/4]: 4
[ Exiting ]
```