

Assignment -2
Python Programming

Assignment Date	26 September 2022
Student Name	S.Suriya Jothi
Student Roll Number	912619104029
Maximum Marks	2 Marks

Question-1:

Consider a list (list = []). You can perform the following commands:

insert i e: Insert integer at position .

print: Print the list.

remove e: Delete the first occurrence of integer .

append e: Insert integer at the end of the list.

sort: Sort the list.

pop: Pop the last element from the list.

reverse: Reverse the list.

Initialize your list and read in the value of followed by lines of commands where each command will be of the types listed above. Iterate through each command in order and perform the corresponding operation on your list.

Solution:

```
import sys
if __name__ == '__main__':
    N = int(input())
    my_list = []

    inputs = []
    for line in sys.stdin:
        inputs.append(line)
    for item in inputs:
        if item[0:5] == 'print':
            print(my_list)
        elif item[0:2] == 'in':
            inserts = [s for s in item.split()][1:3]
            inserts = list(map(int, inserts))
            my_list.insert(inserts[0], inserts[1])
        elif item[0:3] == 'rem':
            inserts = list(map(int, [s for s in
            item.split()][1]))
            my_list.remove(inserts[0])
        elif item[0:2] == 'ap':
            inserts = list(map(int, [s for s in
            item.split()][1]))
            my_list.append(inserts[0])
```

```
elif item[0:4] == 'sort':
    my_list.sort()
elif item[0:3] == 'pop':
    my_list.pop()
elif item[0:7] == 'reverse':
    my_list.reverse()
```

Sample Input 0

```
12
insert 0 5
insert 1 10
insert 0 6
print
remove 6
append 9
append 1
sort
print
pop
reverse
print
Sample Output 0
```

```
[6, 5, 10]
[1, 5, 9, 10]
[9, 5, 1]
```

Question-2:

Write a Calculator program in Python?

Solution:

```
def add(x, y):
    return x + y
def subtract(x, y):
    return x - y
def multiply(x, y):
    return x * y
def divide(x, y):
    return x / y
print("Select operation.")
print("1.Add")
print("2.Subtract")
```

```

print("3.Multiply")
print("4.Divide")
while True:
    choice = input("Enter choice(1/2/3/4): ")
    if choice in ('1', '2', '3', '4'):
        num1 = float(input("Enter first number: "))
        num2 = float(input("Enter second number: "))
        if choice == '1':
            print(num1, "+", num2, "=", add(num1, num2))
        elif choice == '2':
            print(num1, "-", num2, "=", subtract(num1, num2))
        elif choice == '3':
            print(num1, "*", num2, "=", multiply(num1, num2))
        elif choice == '4':
            print(num1, "/", num2, "=", divide(num1, num2))
        next_calculation = input("Let's do next calculation? (yes/no): ")
        if next_calculation == "no":
            Break
        else:
            print("Invalid Input")

```

```

Select operation.
1.Add
2.Subtract
3.Multiply
4.Divide
Enter choice(1/2/3/4): 3
Enter first number: 15
Enter second number: 14
15.0 * 14.0 = 210.0
Let's do next calculation? (yes/no): no

```

Question-3:

Write a program to concatenate, reverse and slice a string?

Solution:

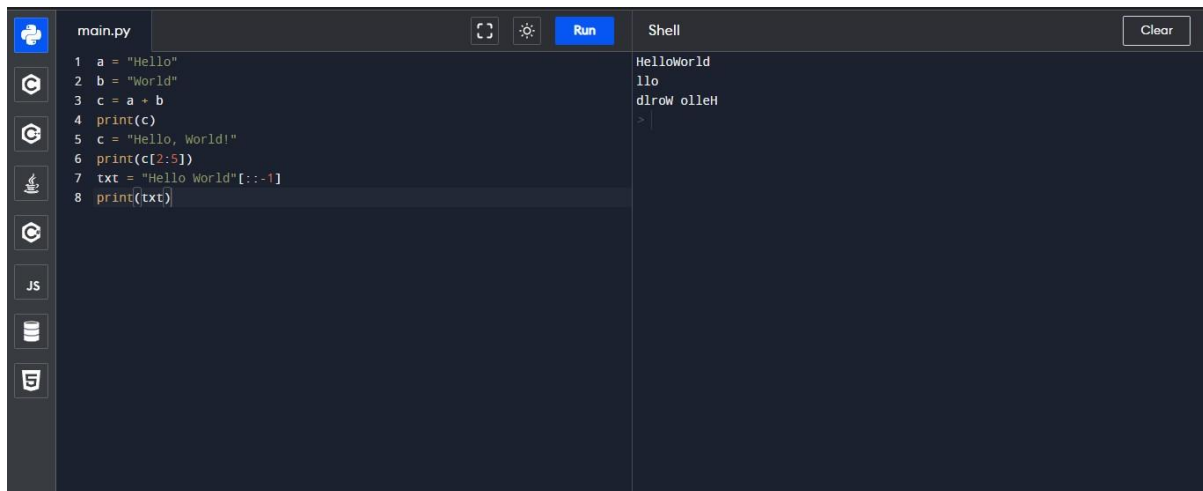
```

a="Hello"
b="World!"
c=a+b
Print(c)
c="Hello World!"

print(c[2:5])
txt = "Hello World"[::-1]

```

`print(txt)`



The screenshot shows a Python IDE interface. On the left, a file named 'main.py' contains the following code:

```
1 a = "Hello"
2 b = "World"
3 c = a + b
4 print(c)
5 c = "Hello, World!"
6 print(c[2:5])
7 txt = "Hello World"[::-1]
8 print(txt)
```

On the right, the 'Shell' window displays the output of the script:

```
HelloWorld
llo
dlroW olleH
> |
```

Question-4:

Why is Python a popular programming language?

Python is easy to learn

It uses a simplified syntax with an emphasis on natural language, for a much easier learning curve for beginners. And, because Python is free to use and is supported by an extremely large ecosystem of libraries and packages, it's often the first-choice language for new developer.

Question-5:

What are the other Frameworks that can be used with python?

There are various frameworks of python like:

- Bottle
- Flask
- Django
- Web2py
- AIOHTTP
- CherryPy
- Dash
- Falcon
- Growler
- UvLoop
- Pyramid
- Sanic
- CubicWeb
- TurboGears
- Hug

- MorePath

These are some of the frameworks used in python.

Question-6:

Full form of WSGI?

Web Server Gateway Interface