

Question 1:

Python 2 VS Python 3

Python 2:

- Many older libraries built for Python 2 are not forwards compatible
- Strings are stored as ASCII
- It rounds the calculations down to the nearest whole number
- `print "hello"`


Python 3:

- Most of today's libraries are strictly for use with python
- Text strings are unicode by default
- The expression `5/2` will return the expected result
- `print("hello")`

Images of code and process

–Input the string “Python” as a list of characters from console, delete at least 2 characters, reverse
input: •python •Sample output: •ntyp

```
 print("test")
```

```
 test
```

Creating variables and asking user for input

```
[2] def strManipulation(myString):  
    if len(myString) == 0:  
        return myString  
    else:  
        rev_str = myString[:2] + myString[4:]  
        return rev_str[::-1]
```

```
[3] usrString = input("Enter python for testing purpose: ")  
print(strManipulation(usrString))
```

```
Enter python for testing purpose: Python  
noyP
```

- Take two numbers from user and perform arithmetic operations on them.


```
[4] def add(num1, num2):  
    total = num1 + num2  
    return total
```

```
[5] print(" This will add two numbers ")  
print()  
  
num1 = float (input("Enter your first number: "))  
num2 = float (input("Enter your second number: "))  
  
print(add(num1,num2))
```

This will add two numbers

Enter your first number: 2
Enter your second number: 3
5.0

```
[6] def replaceOccur(sent1):  
    sent2 = sent1.replace("python","Pythons")  
    return sent2
```

```
 sent = input("Enter the sentence: I love with playing python: ")  
print(replaceOccur(sent))
```

Enter the sentence: I love playing python: I love playing with python
I love playing with Pythons