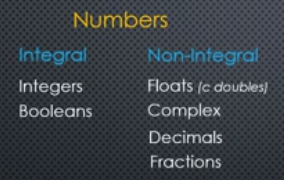
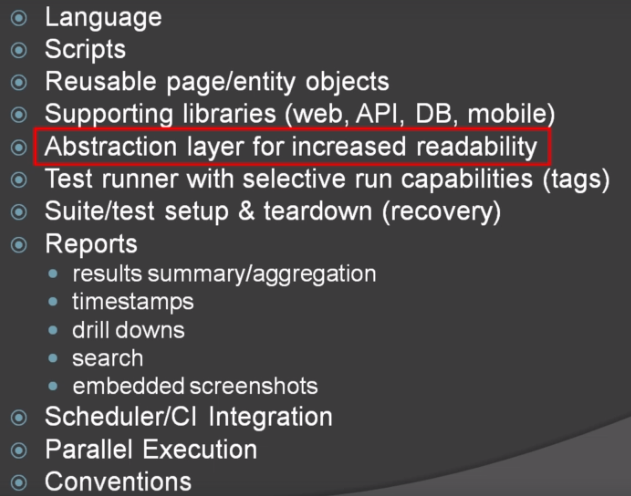
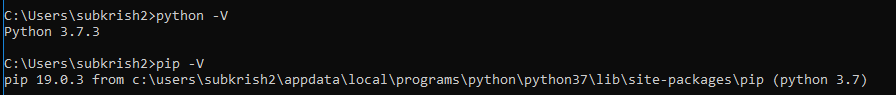
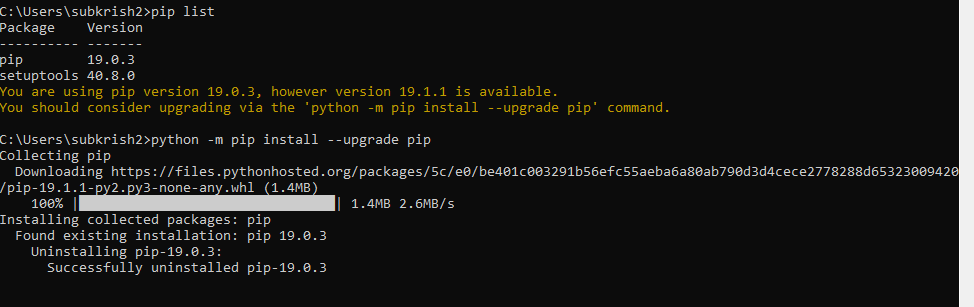
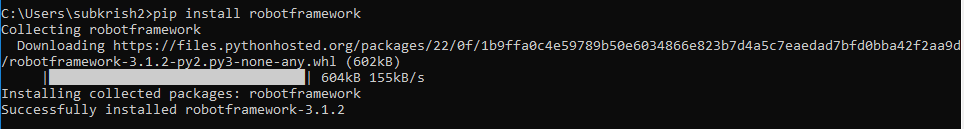
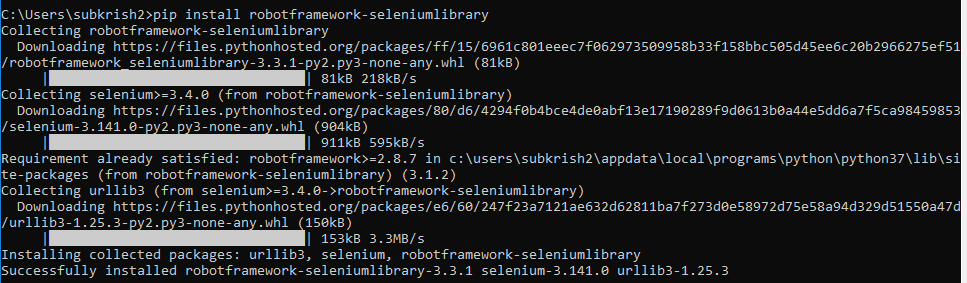
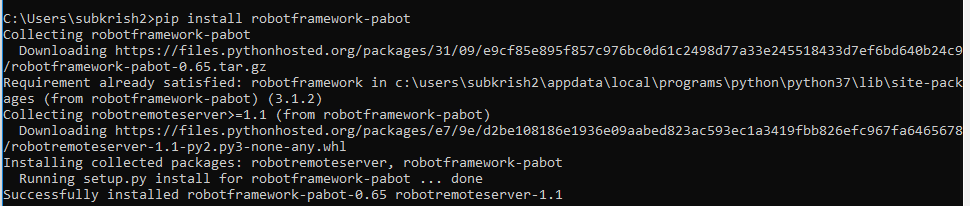
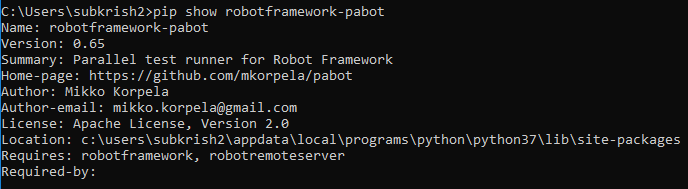
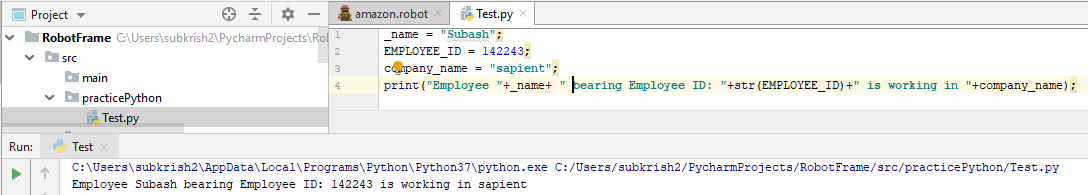
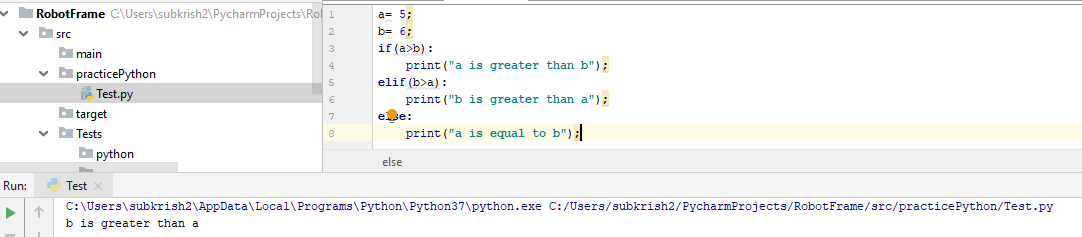
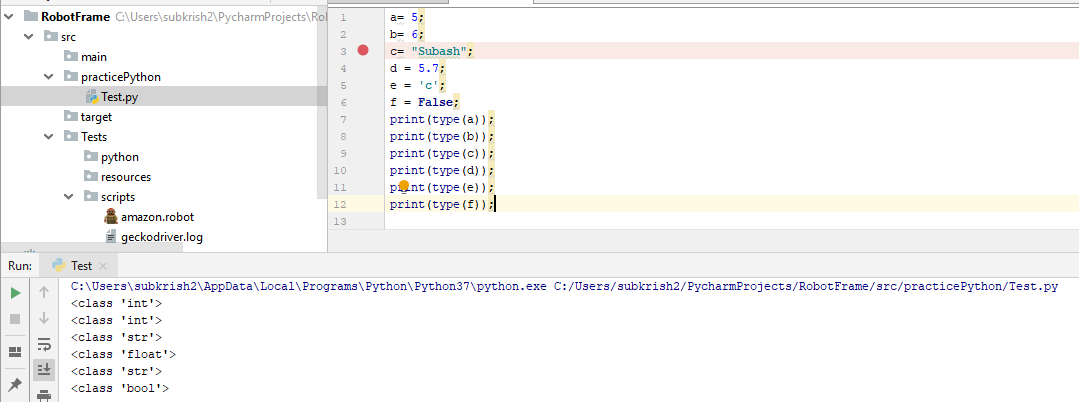
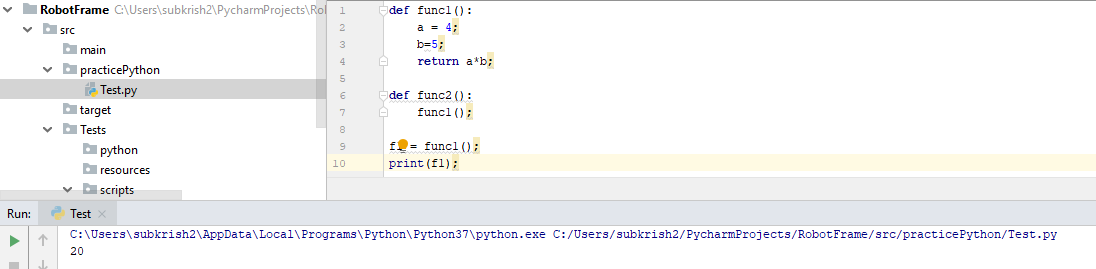
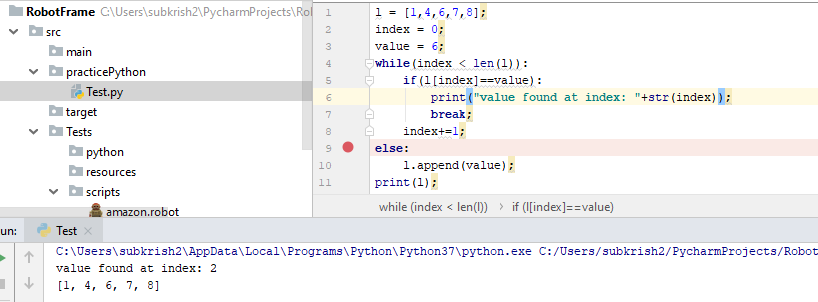
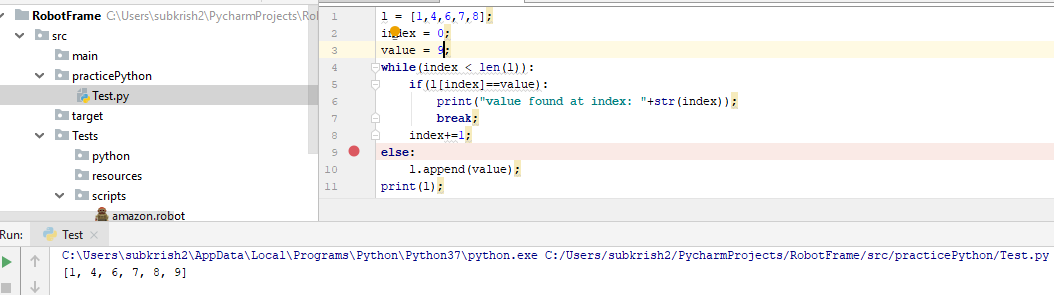
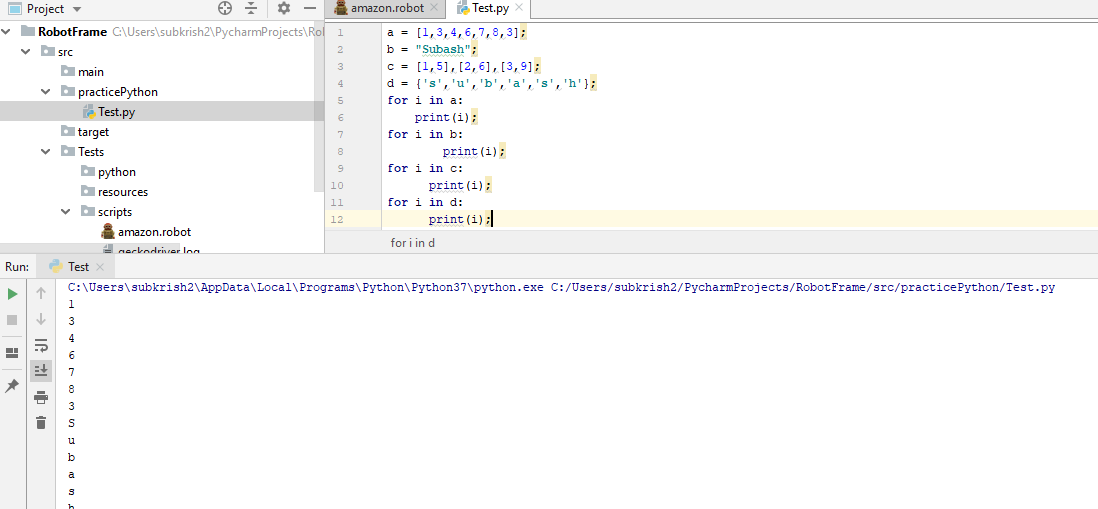
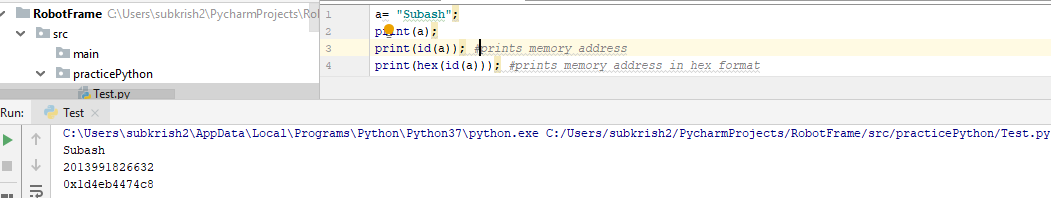
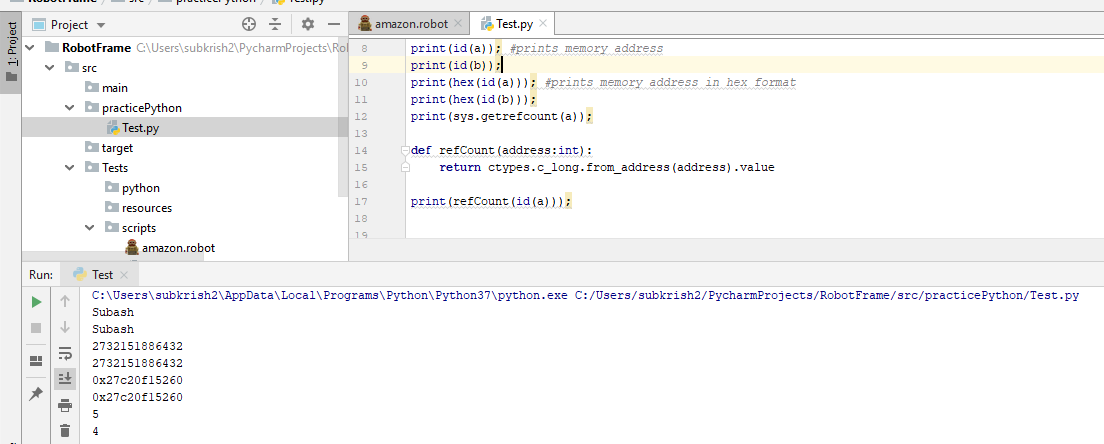
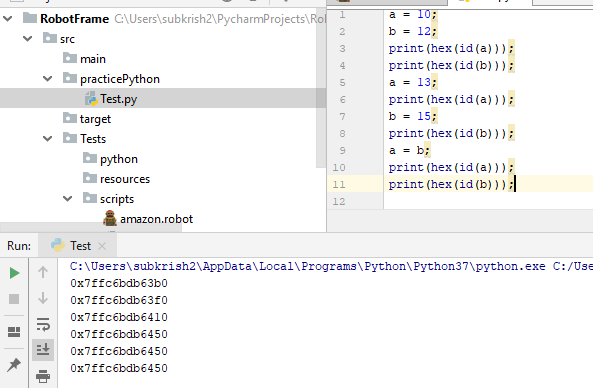
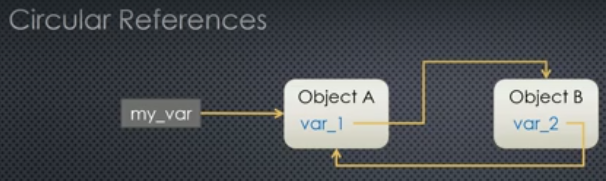
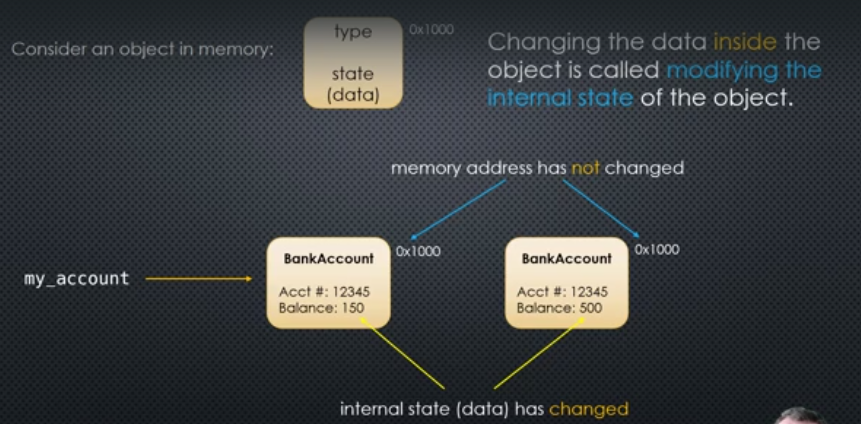
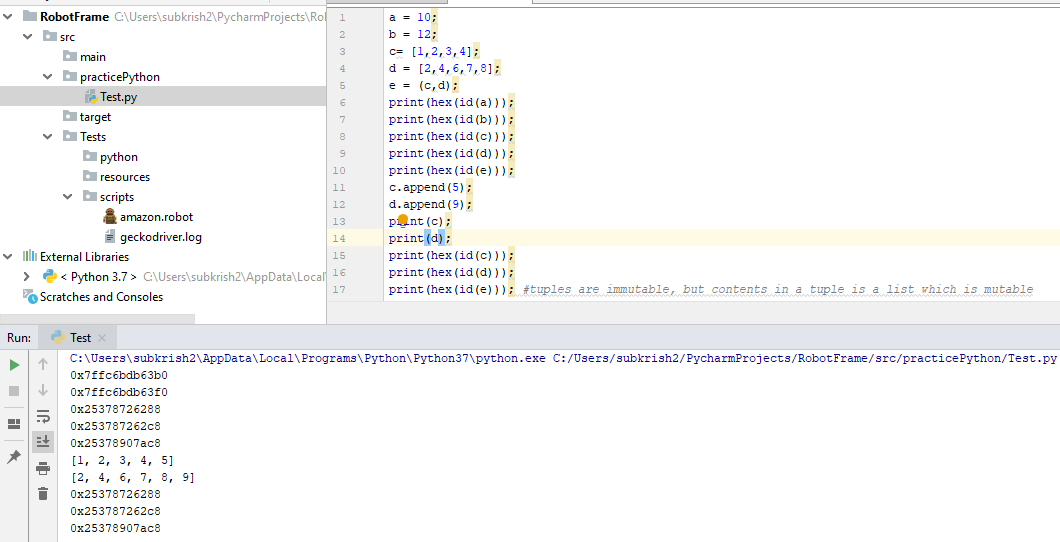
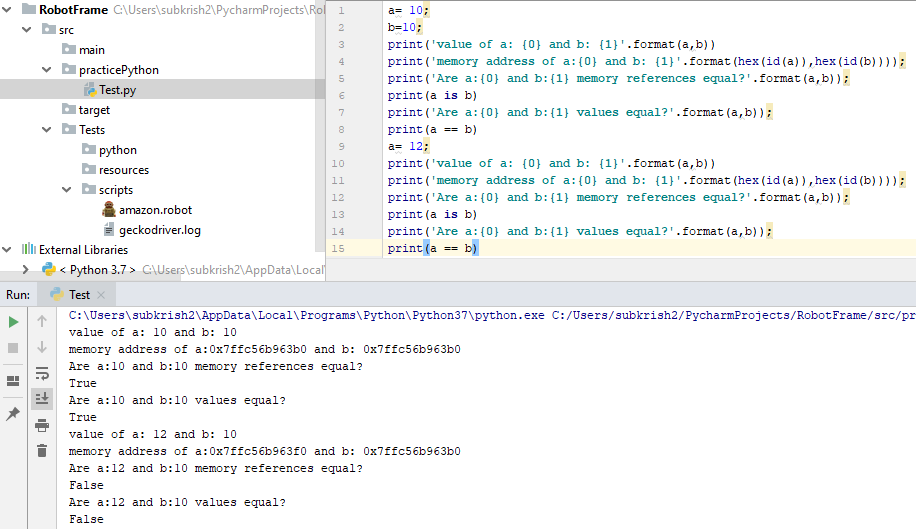
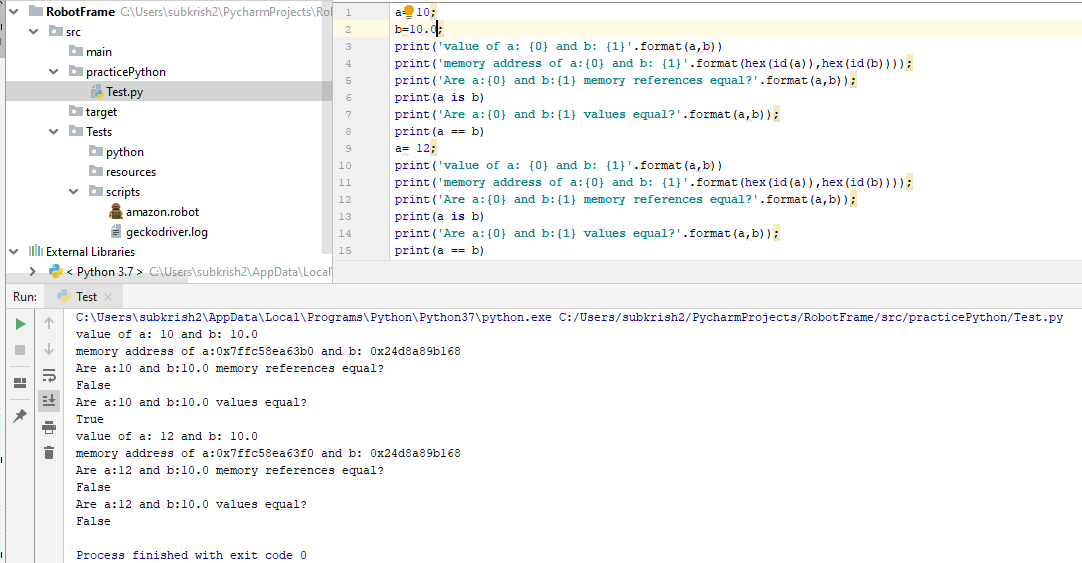
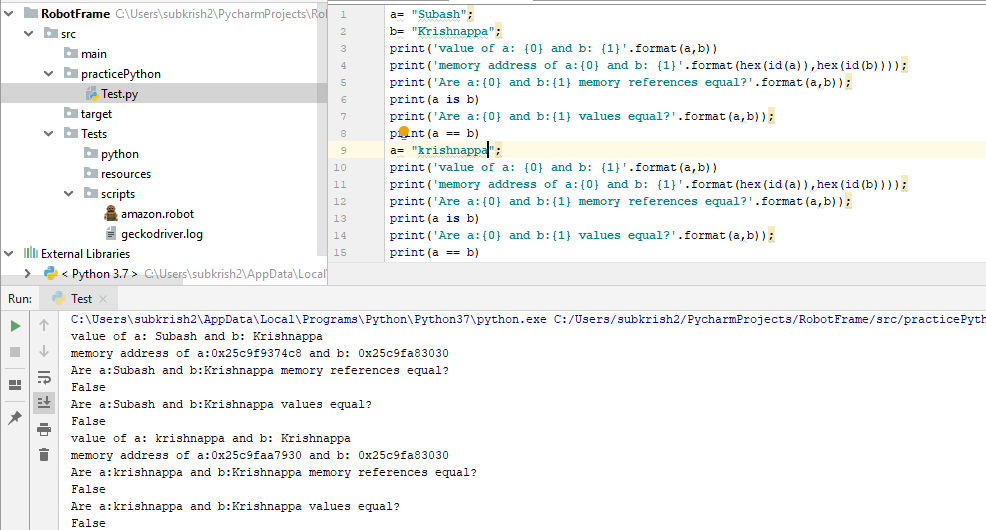
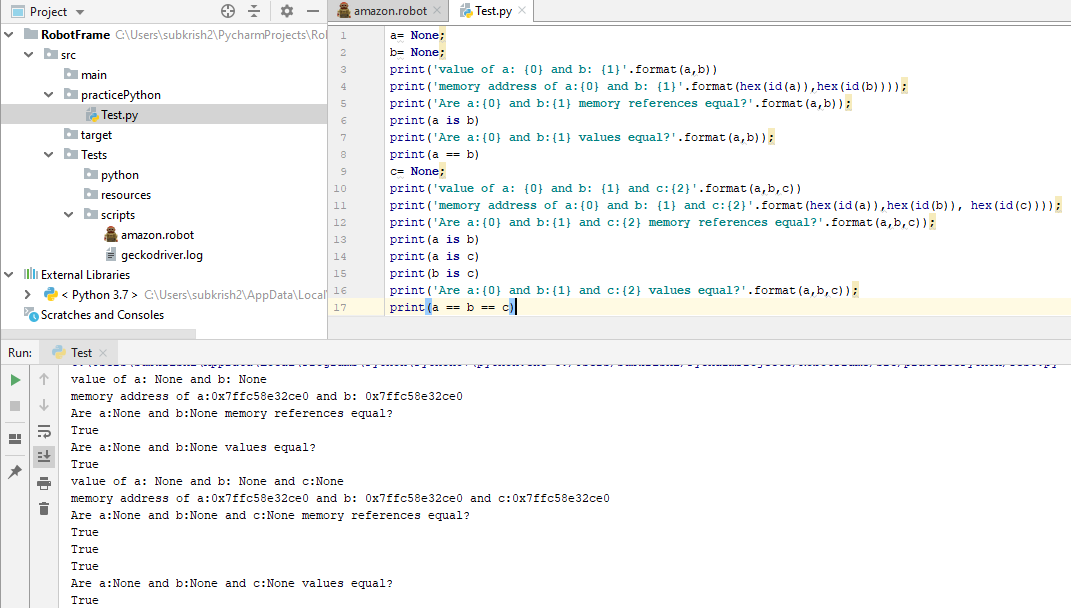
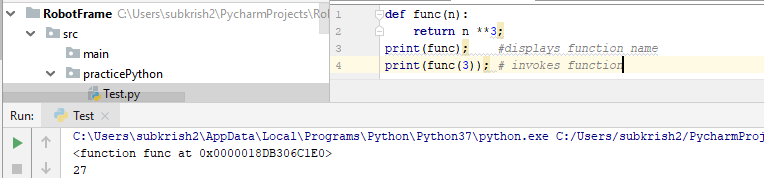
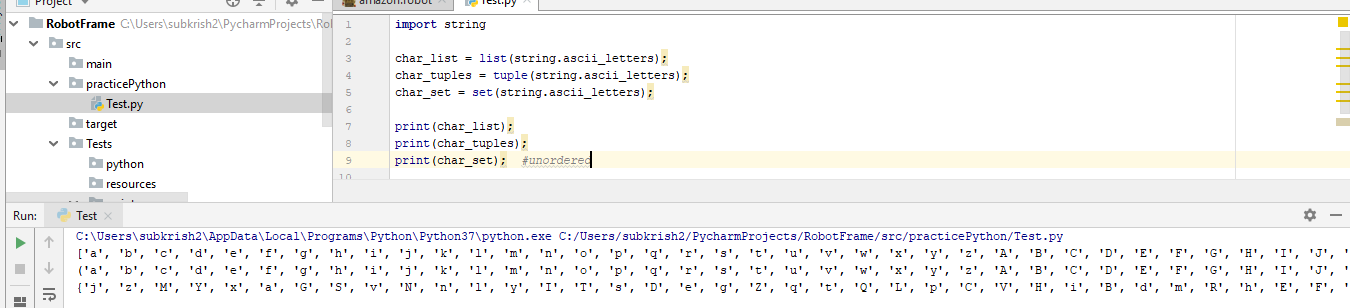
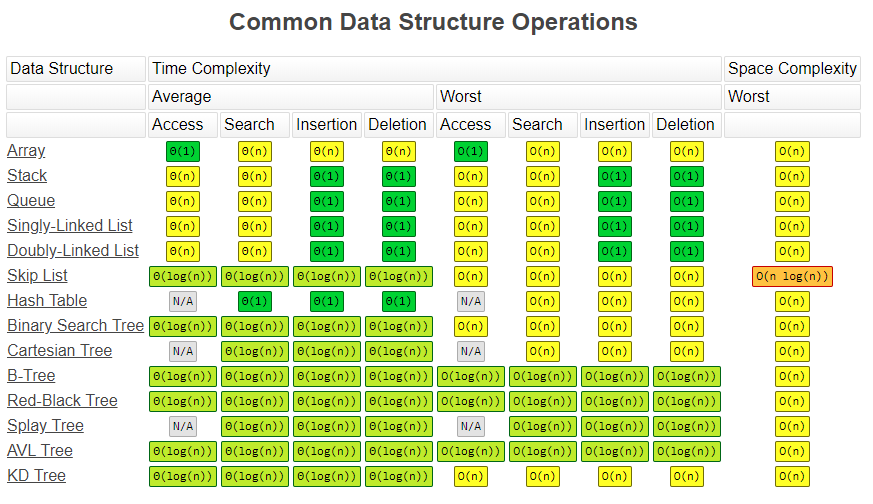
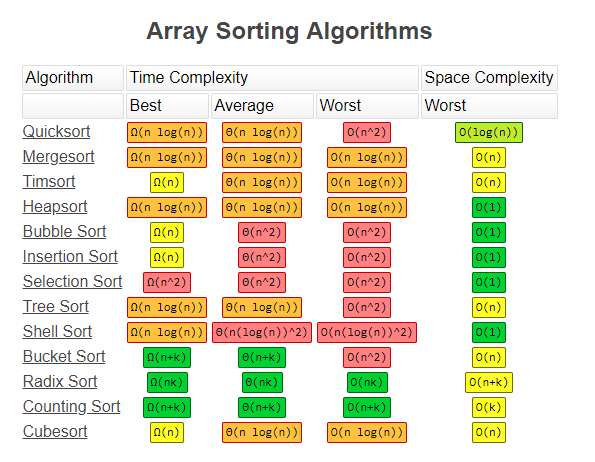
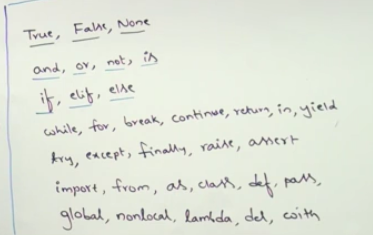
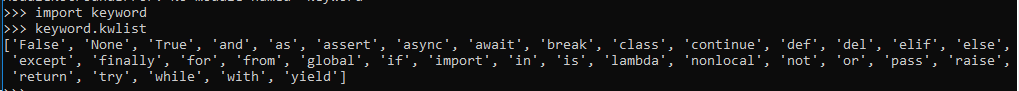
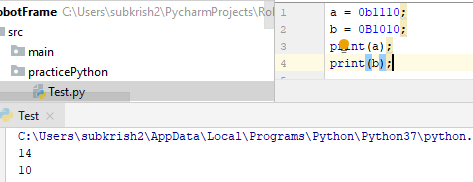
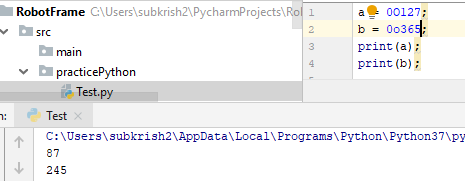
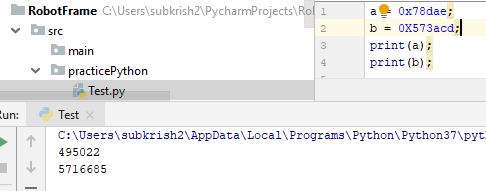
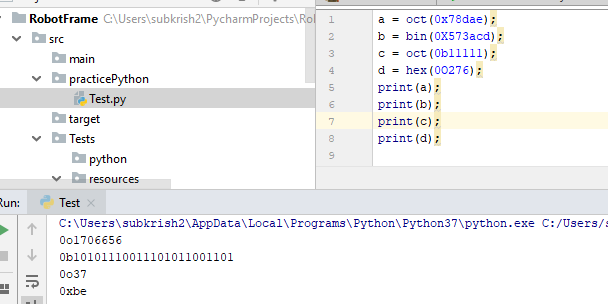
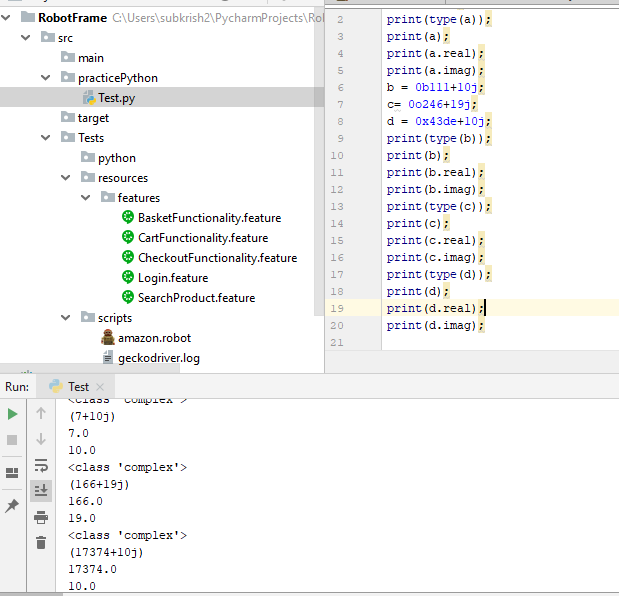
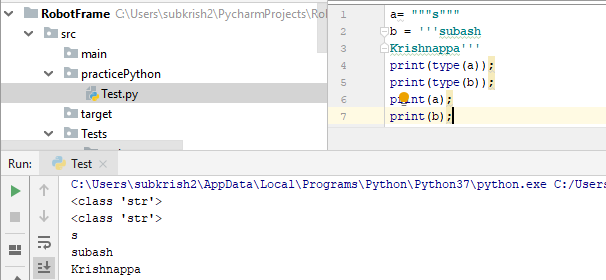
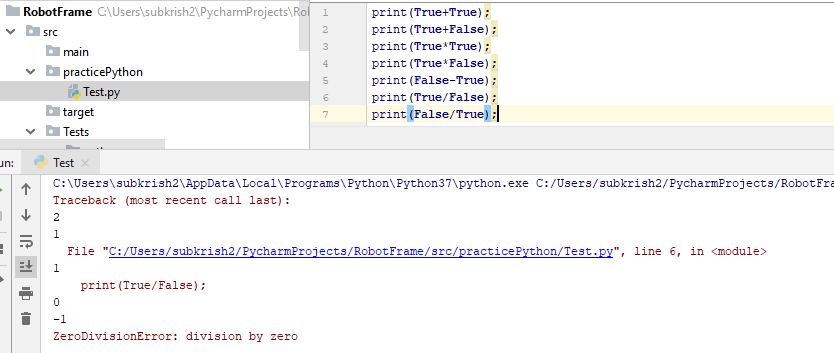
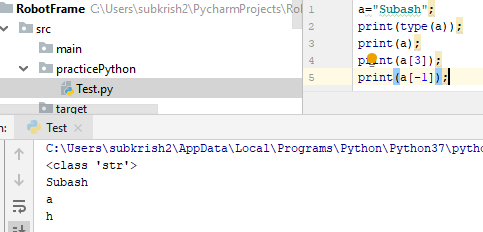
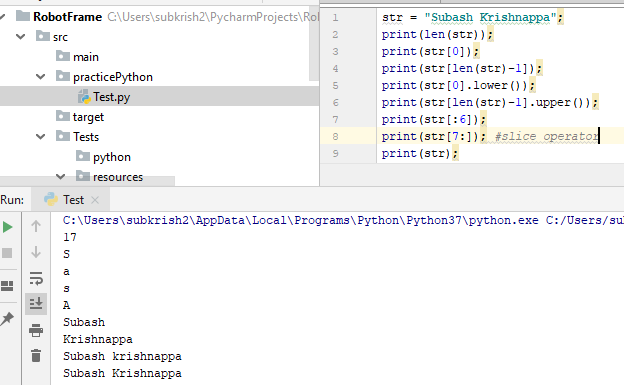
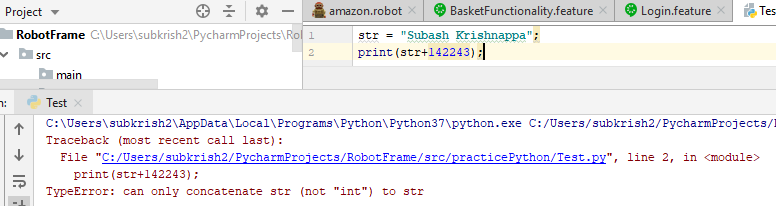
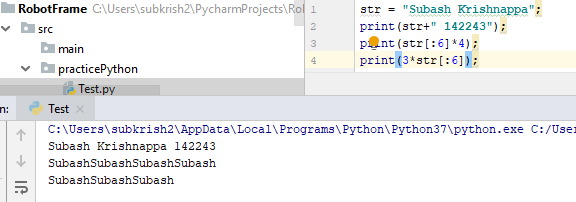
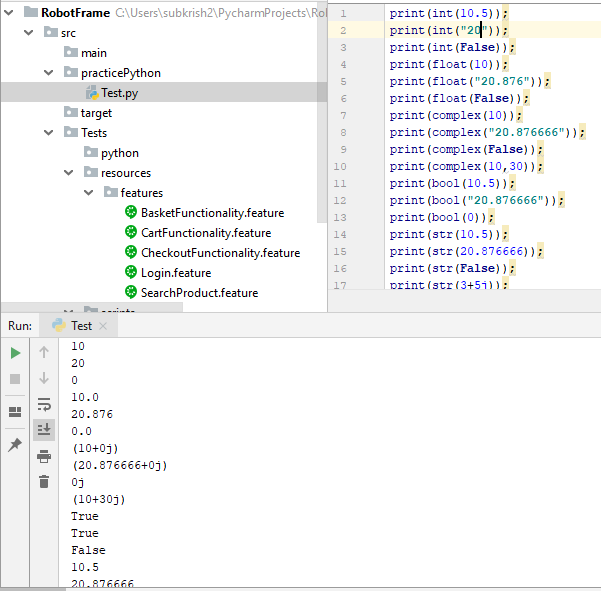
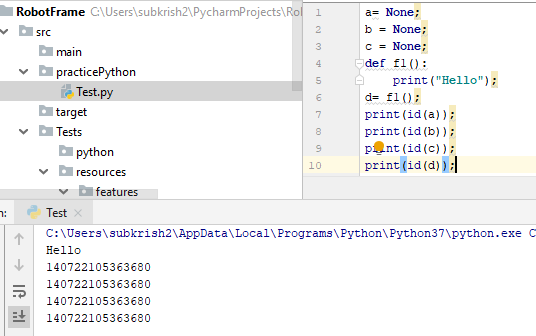
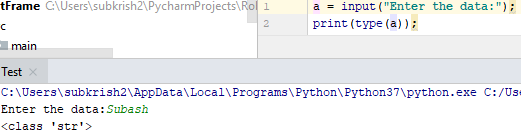
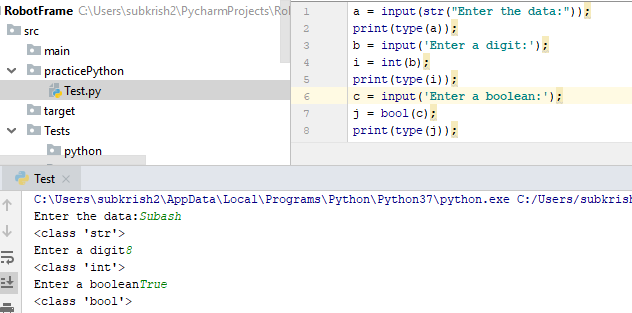
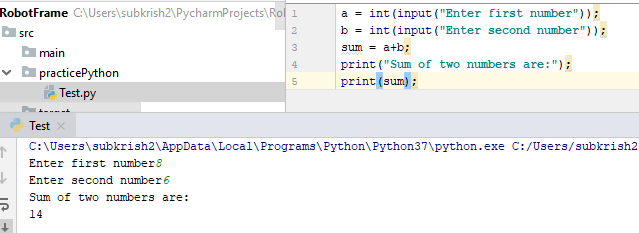
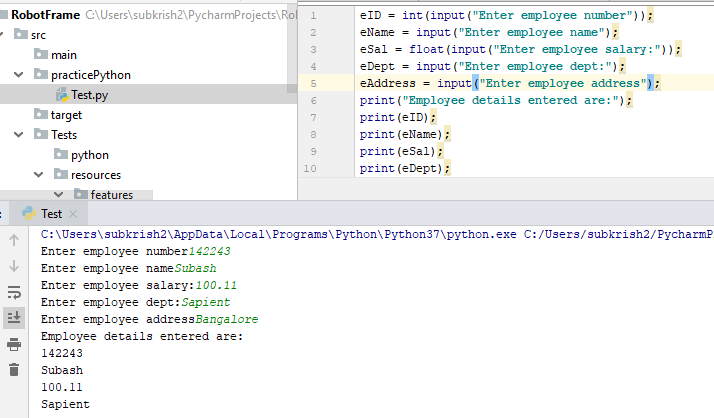
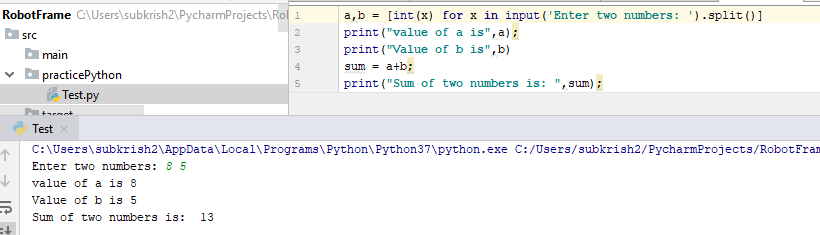
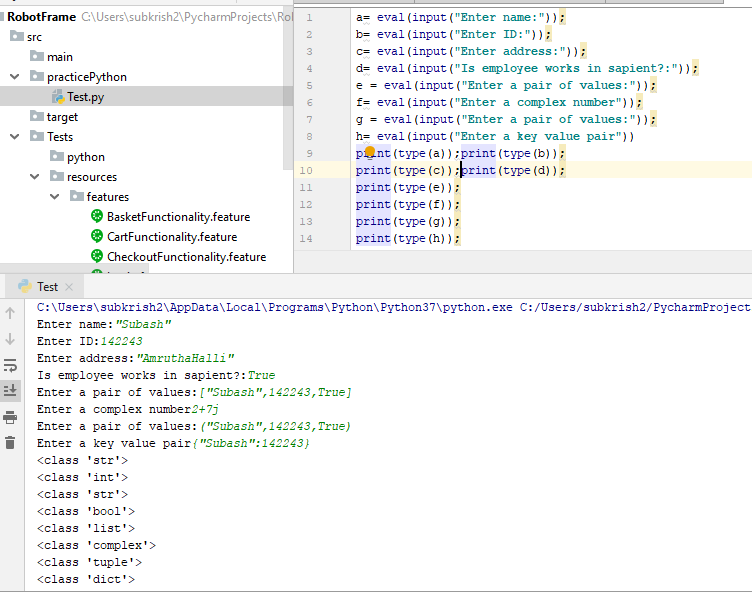
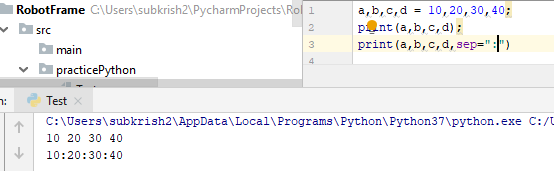
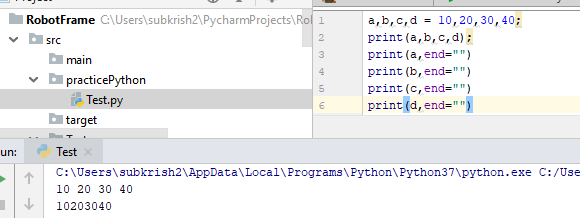
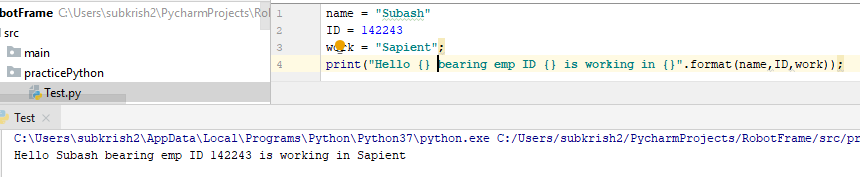
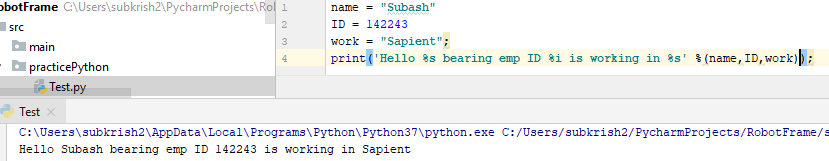
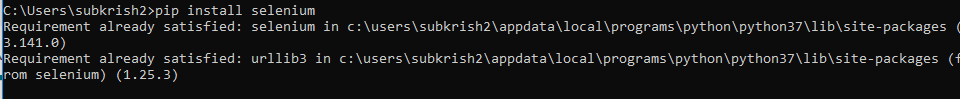
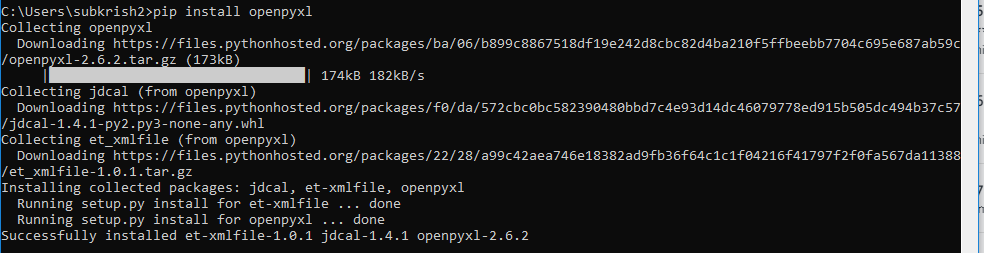
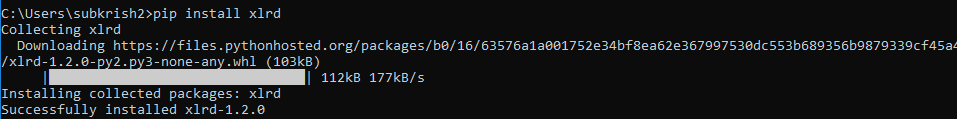
1. 
2. 
3. 
4. Robot Framework:  
   
5. 
6. 
7. Install robot framework  
   
8. Install robot-seleniumlibrary  
   
9. For parallel execution install pabot  
   
10. Verify installation  
    
11. 
12. If , elif and else clauses:  
    
13. Get type of variables  
    
14. Functions:  
    
15. While loop:  
      
    
16. For loop  
    
17. Variables  
    
18. Reference counting done by python memory manager and is done automatically
19. Finding reference count for a variable  
    
20.   
    memory address assigned to variables on reassignment
21. Garbage Collection  
      
    By default garbage collector is turned on, but can be turned off, must need to take care of circular reference
22. Object mutability and immutability  
    Changing data inside the object is called modifying the internal state of the object  
      
    Object whose internal state can be changed is mutable(Lists,sets,dictionaries)  
    Object whose internal state cannot be changed is immutable(int, float, Boolean, complex numbers, strings,tuples)  
    
23. Variable Equality:  
      
      
      
    
24. 
25. Interning: Reusing objects on demand  
    at startup, CPython loads(caches) a global list of integers in the range -5 till 256, any time an integer is referenced in that range, python will use cached version of the object -> these are singleton objects
26. PeepHole Optimization  
    
27. Numeric Types:  
    int, float.Float, decimal.Decimal, Boolean, complex data types
28. Data types:  
    int, float, str,complex, bool,list,tuple,dict,set,frozenset, byte, bytearray,
29. Specialized data types:  
    None,range,bin
30. Integers:

Python part 2:

1. 
2.   
   python maintained by Python Software Foundation(PSF)
3. Variable -> starting with underscore like \_x is a protected variable  
   \_\_x is a private variable  
   \_\_x\_\_ is a magic variable
4. Reserved Keywords:  
   
5. Display keywords in python  
   
6. Everything in python is an object
7.  memory address of a
8. Long data type is present in python 2, but in python 3, even huge values are represented using python 3
9. Display binary numbers  
   
10. Display octal numbers  
    
11. Display hexadecimal numbers  
    
12. Base conversion functions:  
    
13. Complex numbers  
    
14. String data type -> Multi line string literal, triple quotes are used for enclosing single and double quotes in a string as is
15. 
16. Bool operator mathematical operation  
    
17. Python supports both positive and negative indexes  
    
18. 
19. Concatenation Operator:  
    concatenation operator must be applied only if both he operands are strings, else, type exception error is displayed  
    
20. String repetitive operator:  
    
21. Typecasting/type-coersion:  
    
22. Immutability concepts are not applicable to complex numbers, but is applicable for other data types
23. None objects:  
    
24. Comments: single line comment using #  
    Multi line comment is not available in python
25. List Data type:
26. Tuple data type:
27. Difference between append and add
28. Dict data type:
29. Set data type:
30. Range data type:
31. Bytes and ByteArray data type:
32. Difference between == and is operator:  
    == is meant for content comparision  
    is operator is address(reference) comparision
33. Raw\_input() -> any data entered the type is “str”, available in 2.x but not in 3.x
34. Input() -> data type is as per the data entered in 2.x and in 3.x data type is always a “str”  
      
      
    
35. Typecasting:  
    
36. Python program to display sum of two numbers:  
    
37. Program to read employee data and display the data in the console:  
    
38. Read two numbers from keyboard in a single line:  
    
39. Eval function: eval()  
    
40. All command line arguments always are available in string format
41. Sep()-> separator function  
    
42. End operator  
    
43. Replacement operator  
    
44. Print with formatted string:  
    
45. Python does not support increment and decrement operators
46. Selenium Python:  
      
      
      
    To verify the packages installed  
    