Sequential Algo.

1. Calculate the sum of 2 numbers.
2. Calculate the Area of Rectangle (formula : Length * Breadth).
3. Write an algorithm/ program to find the area of a triangle.
4. Write an algorithm/ program to find the circumference of a circle (formula is 2* pi * r). Take r value from the user.
5. Write a program to take the kilometer value as an input and convert it into a meter.
6. Calculate the area of square (formula : side * side)
7. Take input from the user and find the square.
8. Calculate the square root of the user given number.
9. Calculate the cube of the user given number (take input from user).
10. Take 3 numbers from the user and calculate the average of that.
11. Find the 3% of 120.
12. Take marks from the user out of 120 and find what percentage the user gets.
13. Take a number from the user and Find the remainder if it is divided by 3.
14. Take input from the user and divide it by 3 such that output should be whole number not decimal(float).
15. Take a number from the user and show the last digit of that number. Hint: use number 10 for divide
16. Remove the last digit of any number taken by the user. Hint : use number 10
17. Add 6 in the 241 such that output would be 2416. Note : 6 and 241 both are numbers.
18. Swap the two number. Write a program to take 2 number and swap them.

Conditional Algorithm.

1. Take a number from user and check its even or odd.
2. Take 2 number from user and check which is greater.
3. Write a program to take 3 number as input and check which is greatest one.
4. Write a program to take 3 number as input and check which one is smallest among all 3.

5. Write a program to take a number from user and check it is divisible by 5 or not.
6. Write a program to check the last digit of a number is divisible of 2 or not.
7. Write a program to check the number is divisible of 3 and 9 or not.
8. Write a program to the character is vowel or not.
9. Write a program to take a string and check the string have vowel or not.
10. Write a program to check whether a person is eligible for voting or not (voting age is >= 18).
11. Write a program to check whether a person is senior citizen or not. If the age is > 60 then the person should be consider as senior citizen.
12. Write a program to check the number is positive or negative.
13. Write a program to whether a number is divisible by 2 and 3 both.
14. Accept the age of 4 people and display the youngest one.
15. Write a program to accept a number from 1 to 7 and display the name of the day like 1 for Sunday, 2 for Monday and so on.
16. Write a program to accept a number from 1 to 12 and display the name of the month like 1 for January, 2 for Feb. and so on.
17. Write a program to accept two numbers and mathematical operators and perform operation accordingly.

    Like : Enter First Number : 7
    Enter Second Number : 9
    Enter Operator : +
    Your Answer is : 16

18. Accept the marks of Eng, Math, Science and Social Science subject and display the stream allotted according to following: ---

    All subject total more than 80 marks ---- science stream
    Total < 80 and above 50 ---- Commerce Stream
    Total < 50 ---- Humanities.

19. Accept the electric unit from user and calculate the bill according to following rates.

    First 100 Units : Free
    Next 200 Units : Rs 2 per day.

Above 300 Units : Rs 5 per day.

If number of unit is 500 then total bill = 0 + 400 + 1000 = 1400

20. Accept the age and the gender (“Male”, “Female”), Number of days and display the wages(salary) accordingly.

Age	Gender	Wage/day
Age >= 18 and Age<30	Gender == “M”	700
	Gender == “F”	750
Age >= 30 and Age<40	Gender == “M”	800
	Gender == “F”	850

Iterative Algorithm.

1. Write an algorithm/program to print the number from 1 to 10.
2. Write an algorithm/ program to print the number from 20 to 31.
3. Write an algorithm/program to print “Hello” 5 times.
4. Write an algorithm/ program to print “Hello, World This is Python” 10 times.
5. Write an algorithm/ program to print the numbers from 5 to 1.
6. Write an algorithm/ program to print name of user and print he/her name 7 times.
7. Write an algorithm/ program to enter the number till the user enter ZERO.
8. Write an algorithm/ program to print the first 10 even number.
9. Write an algorithm/ program to print the first 10 odd number.
10. Write an algorithm/ program to iterate over a list.
11. Write an algorithm/ program to iterate over a string “Hello”.
12. Write an algorithm/ program to check the string have vowel or not.
13. Write an algorithm/ program to print the table of 2.
14. Write an algorithm/ program to print the table of 12.

15. Write an algorithm/ program to print the sum of all even number between 1 to 10.
16. Write an algorithm/ program to print the sum of all odd number between 1 to 10.
17. Write an algorithm/ program to print the factorial of number accepted from user.
18. Write an algorithm/ program to reverse the number.
19. Write an algorithm/ program to check the number is palindrome or not.
20. Write an algorithm/ program to count the digit of number accepted from the user.
21. Write an algorithm/ program to sum of the digit of number accepted from the user.
22. Write an algorithm/ program to product of the digit of number accepted from the user.
23. Write an algorithm/ program to check whether a number is Armstrong or not.
24. Write an algorithm/ program to check the string is palindrome or not.
25. Write an algorithm/ program to check the number is prime or not.
26. Write an algorithm/ program to enter the number till the user enter ZERO and at the end it should display the count of positive and negative numbers entered.
27. Write an algorithm/ program to find the HCF (Highest Common Factor) of two numbers entered from the user.
28. Write an algorithm/ program to convert Decimal to Binary and Binary to Decimal.
29. Write an algorithm/ program to accept any number and print the ascii value of each digit of number.
30. Write an algorithm/ program to display the name of the digit of number accepted from the user. Like for 231 output Two Three One

Datatype

String operations.

1. Write a program to join the below string.
   ---- a = "Hello" and b="World"

2. Write a program to join below 5 for words.
   ---- a = "H", b="e", c="l", d="l" and e="o"

3. Write a program to replicate "king" 5 times.

4. Write a program to replicate "Hello" 3 times such each Hello will have sapces.
    ---- eg. output ==> Hello Hello Hello Hello Hello

5. Write a program for following output.
    ---- a = "Hello", b="world", c = "Python"
    ---- output ==> Hello Hello World Hello Hello Hello Python Hello

6. Write a program for the following output :
    ---- a = "The Jungle King", b="Tiger", c = "Jumble"
    ---- output : "Tiger Jumble Tiger Tiger Tiger The Jungle King"

7. Write a program to extract the last character of.
    ---- a = "Hello"

8. Write a program to extract the letter "e" from "Hello".
9. Write a program to extract the letter "o" from "Hello".
10. Write a program to extract the letter "He" from "Hello".
11. Write a program to extract the letter "elo" from "Hello" without slicing.
12. Write a program to extract the letter from "The Jungle King" such that output will be "JngKing" without slicing.

13. Write a program to for the following output without slicing.
    ---- text = "The Jungle is a Green Place"
    ---- output : "The is"

14. Write a program to for the following output without slicing.
    ---- text = "The Jungle is a Green Place"
    ---- output : "TheThe is is"

15. Write a program to for the following output without slicing.
    ---- text = "The Jungle is a Green Place"
    ---- output : "TheTheTheTheTheTheis is is is is Greeeeeeeeeeen"

16. Write a program to reverse the following string.
    ---- text = "The Jungle book"

17. Write a program to reverse the following string.
    ---- text = "Python developer"

18. Write a program for the following output.
    ---- text = "Python developer"
    ---- output : "repoleved nohtyP"

19. Write a program for the following output using slicing.

---- text = "Python developer"
---- output : "developer Python"

20. Write a program for the following output using slicing.
---- text = "Python developer"
---- output : nohtyP repoleved"

21. Write a program for the following output using slicing.
---- text = "Python Developer"
---- output : "thon loper"

22. Write a program for the following output using slicing.
---- text = "Python Developer"
---- output : "thon loper loper loper loper"

23. Write a program for the following output using slicing.
---- text = "Python Developer"
---- output : "thon thon thon loperloperloper"

24. Write a program for the following output using slicing.
---- text = "Python Developer"
---- output : "Python loper Devel Dev Devel"

25. Write a program for the following output using slicing.
---- text = "Python Developer"
---- output : "Ph Dlr Developer"

26. Write a program for the following output using slicing.
---- text = "Python Developer"
---- output : "thon loper yoDe"

27. consider a text = "hELLo WorlD tHiS iS pYthON"

28. Write a program for the following output.
--- output : 'Hello world this is python'

29. Write a program for the following output.
--- output : 'Hello World This Is Python'

30. Write a program for the following output.
--- Output : 'HELLO WORLD THIS IS PYTHON'

31. Write a program for the following output.
--- Output : 'hello world this is python'

32. Write a program for the following output.
    --- Output : "   hELLo WorlD tHiS iS pYthON   "

33. Write a program for the following output.
    --- Output : '----------hELLo WorlD tHiS iS pYthON----------'

34. Write a program for the following output.
    --- Output : '----------hELLo WorlD tHiS iS pYthON----------'

35. Write a program to count the letter "o" in the following string.
    --- text = "Hello world im Python"

36. Write a program to count the letter "h" in the following string.
    --- text = "The world have may famours poets in this place.

37. Write a program to count the letter "is" in the following string.
    --- text = "His first love is Dog and last love is Dog and as she is dog Lover."

Tuple operations.

a = (1, 3, 2, 5, 7, 9)

b = ("Hello", "World", 1, 5, 2)

c = ("My", "Name", "is", "Python")

d = ("Hello", "Hello", "Hello", "World", 1, 2, 1, "Hello", 5, 2)

1. Write a program to get the below output.
   --- (1, 3, 2, 5, 7, 9, "Hello", "Wrold", 1, 5, 2)

2. Write a program to get the below output.
   --- ("Hello", "Wrold", 1, 5, 2, 1, 3, 2, 5, 7, 9)

3. Write a program to get the below output.
   --- ("Hello", "Wrold", 1, 5, 2, "Hello", "Wrold", 1, 5, 2, "Hello", "Wrold", 1, 5, 2)

4. Write a program to get the below output.
   --- (1, 3, 2, 5, 7, 9, 1, 3, 2, 5, 7, 9)

5. Write a program to get the below output.
   --- ("My", "Name", "is", "Python", "My", "Name", "is", "Python", "My", "Name", "is", "Python")

6. Write a program to get the below output.
   --- 2

7. Write a program to get the below output.
   --- "Hello"

8. Write a program to get the below output.
   --- "Name"

9. Write a program to get the below output.
   --- "World"

10. Write a program to get the below output.
    --- 9

11. Write a program to get the below output.
    --- "My"

12. Write a program to get the below output.
    --- ("My", "Name")

13. Write a program to get the below output.
    --- ("is", "Python")

14. Write a program to get the below output.
    --- ("World", 1, 5)

15. Write a program to get the below output.
    --- (3, 2, 5, 7)

16. Write a program to get the below output.
    --- (3, 2, 5, 7, "is", "Python")

17. Write a program to get the below output.
    --- ("is", "Python", 3, 2, 5, 7)

18. Write a program to get the below output.
    --- (3, 2, 5, 7, "is", "Python", "Hello")

19. Write a program to get the below output.
    --- (3, 2, 5, 7, "Hello", "Wrold")

20. Write a program to get the below output.
    --- ("Hello", "Wrold", 3, 2, 5, 7)

21. Write a program to get the below output.
    --- (3, 2,"Hello", "Wrold", "is", "Python")

22. Write a program to get the below output.
    --- (3, 2,"Hello", "Wrold", "is", "My", "Python")

23. Write a program to extract "Hell" from "Hello", "thon" from "Python".

24. Write a program to get the below output.
    --- "ython Hello Hell Hell Hell Python Python Pyhton"

25. Write a program to get the below output.
    --- (9, 7, 5, 2, 3, 1)
    --- (2, 5, 1, "World", "Hello")
    --- ("Python", "is", "Name", "My")

26. Write a program to get the below Output.
    --- (2, 3, 1)
    --- (1, "World", "Hello")
    --- ("is", "Name", "My")

27. Write a program to get the below Output.
    --- (7, 5, 2, 3)
    --- (5, 1, "World", "Hello")
    --- ("Python", "is")

28. Write a program to get the below Output.
    --- (1, 2, 7)
    --- ("World", 5)
    --- ("My", "Python")

29. Write a program to get the below Output.
    --- (1, 5)
    --- ("Hello", 2)
    --- ("My", "Name", "is", "Python")

30. Write a program to get the below Output.
    --- (1, 3, 2, 5, 7, 9)
    --- ("World", 5)
    --- ("My", "Name", "is", "Python")

31. Write a program to get the below Output.
    --- (7, 2, 1)
    --- (5, "World")

--- ("Python", "My")

32. Write a program to get the below Output.
    --- (9, 5, 3)
    --- (5, "World", 2)
    --- ("Python", "My", "Python", "Python", "thon", "thon", "thon")

33. Write a program to count the number of time "Hello" is.
34. Write a program to count the number of time 2 is.
35. Write a program to find the index location of 1.
36. Write a program to find the index location of 2nd 1.
37. Write a program to find the index of 2.
38. Write a program to find the index of 2nd 2.
39. Write a program to find the index of "Hello".
40. Write a program to find the index of 2nd "Hello".

List operations.

a = [1, 3, 2, 5, 7, 9]
b = ["Hello", "World", 1, 5, 2]
c = ["My", "Name", "is", "Python"]
d = ["Hello", "Hello", "Hello", "World", 1, 2, 1, "Hello", 5, 2]

1. Write a program to get the below output.
   --- [1, 3, 2, 5, 7, 9, "Hello", "Wrold", 1, 5, 2]

2. Write a program to get the below output.
   --- ["Hello", "Wrold", 1, 5, 2, 1, 3, 2, 5, 7, 9]

3. Write a program to get the below output.
   --- ["Hello", "Wrold", 1, 5, 2, "Hello", "Wrold", 1, 5, 2, "Hello", "Wrold", 1, 5, 2]

4. Write a program to get the below output.
   --- [1, 3, 2, 5, 7, 9, 1, 3, 2, 5, 7, 9]

5. Write a program to get the below output.
   --- ["My", "Name", "is", "Python", "My", "Name", "is", "Python", "My", "Name", "is", "Python"]

6. Write a program to get the below output.
   --- output : 2

7. Write a program to get the below output.
   --- "Hello"

8. Write a program to get the below output.
   --- "Name"

9. Write a program to get the below output.
   --- "World"

10. Write a program to get the below output.
    --- 9

11. Write a program to get the below output.
    --- "My"

12. Write a program to get the below output.
    --- ["My", "Name"]

13. Write a program to get the below output.
    --- ["is", "Python"]

14. Write a program to get the below output.
    --- ["World", 1, 5]

15. Write a program to get the below output.
    --- [3, 2, 5, 7]

16. Write a program to get the below output.
    --- [3, 2, 5, 7, "is", "Python"]

17. Write a program to get the below output.
    --- ["is", "Python", 3, 2, 5, 7]

18. Write a program to get the below output.
    --- [3, 2, 5, 7, "is", "Python", "Hello"]

19. Write a program to get the below output.
    --- [3, 2, 5, 7, "Hello", "World"]

20. Write a program to get the below output.
    --- ["Hello", "World", 3, 2, 5, 7]

21. Write a program to get the below output.
    --- [3, 2,"Hello", "World", "is", "Python"]

22. Write a program to get the below output.
    --- [3, 2,"Hello", "World", "is", "My", "Python"]

23. Write a program to extract "Hell" from "Hello", "thon" from "Python".

24. Write a program to get the below output.
    --- "ython Hello Hell Hell Hell Python Python Pyhton"

25. Write a program to get the below output.
    --- [9, 7, 5, 2, 3, 1]
    --- [2, 5, 1, "World", "Hello"]
    --- ["Python", "is", "Name", "My"]

26. Write a program to get the below Output.
    --- [2, 3, 1]
    --- [1, "World", "Hello"]
    --- ["is", "Name", "My"]

27. Write a program to get the below Output.
    --- [7, 5, 2, 3]
    --- [5, 1, "World", "Hello"]
    --- ["Python", "is"]

28. Write a program to get the below Output.
    --- [1, 2, 7]
    --- ["World", 5]
    --- ["My", "Python"]

29. Write a program to get the below Output.
    --- [1, 5]
    --- ["Hello", 2]
    --- ["My", "Name", "is", "Python"]

30. Write a program to get the below Output.
    --- [1, 3, 2, 5, 7, 9]
    --- ["World", 5]
    --- ["My", "Name", "is", "Python"]

31. Write a program to get the below Output.
    --- [7, 2, 1]
    --- [5, "World"]
    --- ["Python", "My"]

32. Write a program to get the below Output.

--- [9, 5, 3]
--- [5, "World", 2]
--- ["Python", "My", "Python", "Python", "thon", "thon", "thon"]

33. Write a program to replace the value 9 with 10 from list a.
34. Write a program to replace the "World" with "MyWorld" from list b.
35. Write a program to replace the "Name" and "is" with "King" from list c.
36. Write a program to replace the 7 with 8 from list a.
37. Write a program to delete " " from the list a without any method.
38. Write a program to add "Java" in list a at the end.
39. Write a program to add "Java" in list b at the end.
40. Write a program to add "Java" in list c at the end.

41. Write a program to add "Java" in list d at the end.
    --- sample output : x =[1, 3, 2, 5, 7, 9, "Java"]

42. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list a at the end as a single value.

43. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list b at the end as a single value.
44. Write a program to add 2 in list c at the end.

45. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list d at the end as a single value.
    --- sample output : x =[1, 3, 2, 5, 7, 9, ["Pro", "Lang", "Ruby", "Java"]]

46. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list a at the end as a different value.
47. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list b at the end as a different value.
48. Write a program to add [2] in list c at the end.

49. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list d at the end as a different value.
    --- sample output : y =[1, 3, 2, 5, 7, 9, "Pro", "Lang", "Ruby", "Java"]

50. Write a program to insert "JS" at 2nd index of list a.
51. Write a program to insert "King" in the list b at 4th index.
52. Write a program to insert "bye" at 1st index of list b.

53. Write a program to find the index of 5 in list a.
54. Write a program to find the index of "World" in list b.
55. Write a program to find the index of "My" in the list c.
56. Write a program to find the index of "is", in the list c.

57. Write a program to find the index of 1 in the list d.

58. Write a program to delete value 2 from list a.
59. Write a program to delete value "World" from list b.
60. Write a program to delete value "Name" from list c.
61. Write a program to delete value "Hello" from list d.
62. Write a program to delete value 2 from list d.
63. Write a program to delete value "World" from list a.

64. new list x = ["Even", "Hello", "Palendrome", 12, "King", 5, 9]
65. new list y = ["Brother", "Sister", "Python", "Java", "Palendrome", 5, 9]

66. Write a program to delete value at index 2 from list x.
67. Write a program to delete value at index 0 from list x.
68. Write a program to delete value at last index from list y.
69. Write a program to delete value at index 3 from list y.
70. Write a program to delete value at index 2 from list y.
71. Write a program to delete value at index -2 from list y

72. New list m = [20, 23, 30, 20, 16, 28, 49, 13, 30]
73. New list n = [1, 36, 41, 3, 8, 22, 37, 31, 13]

74. Write a program to reverse the list m
75. Write a program to reverse the list n.
76. Write a program to reverse the list a.
77. Write a program to reverse the list x.

78. Write a program to sort the list m.
79. Write a program to sort the list n in ascending order.
80. Write a program to sort the list n in descending order.
81. Write a program to sort the list m in descending order.
82. Write a program to arrange the list m in ascending order.

83. Write a program to count the number of time "Hello" is.
84. Write a program to count the number of time 2 is.
85. Write a program to find the index location of 1.
86. Write a program to find the index location of 2nd 1.
87. Write a program to find the index of 2.
88. Write a program to find the index of 2nd 2.

89. Write a program to find the index of "Hello".
90. Write a program to find the index of 2nd "Hello".
91. Write a program to add "Java" in list m at the end.
92. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list m at the end as a single value.
93. Write a program to add ["Pro", "Lang", "Ruby", "Java"] in list n at the end as a different value.
94. Write a program to insert "bye" at 1st index of list n
95. Write a program to find the index of "Lang" in the list m.
96. Write a program to delete value "Pro" from list m.
97. Write a program to delete value at index 2 from list m.
98. Write a program to reverse the list m.
99. Write a program to sort the list n in descending order.

Dictionary Datatype.

1. What is dictionary?
2. Why do we use dictionary?

d = {"name":"Raju", "lname":"Kala", "address":"Thane", "pin":400078, "contact":1234567, "email":"Raj12@gmail.com"}

cart = {"Book":"Rich dad poor dad", "Salary":300000, "company":"Learn and Code", "Platform":"YouTube"}

mydict = { "One":{1, 3, 5, 6), 1:"One", None:"Nothing", True:"Boolean", ("Hello", "World", 45):["My Tuple", "List world"]}

3. Create a empty dictionary with the name of dictionary.
4. Create a dictionary with single pair of key and value pair
5. Write a program to join both d and cart dictionary.
6. Write a program to join all the 3 dictionary cart, d and mydict.
7. Write a program to join d and mydict.

8. Write a program to find the value of key "address" from dictionary d.
9. What is the value of key "pin" from the dictionary d.
10. What is the value of key "One" from the dictionary mydict.
11. What is the value of key ("Hello","World", 45) from the dictionary mydict.

12. Write a program to replace the value of "lname" from "kala" to "Bala" from dictionary d.

13. Write a program to replace the value of pin from 400078 to 400079 from dictionary d.
14. Write a program to replace the value of contact from 1234567 to 987654321 from dictionary d.
15. Write a program to replace the value of Platform from "YouTube" to "Instagram and YouTube" from dictionary cart.

16. Write a program to get all the keys at once.
17. Write a program to get all the keys at once and then convert them into a list.
18. Write a program to get all the values at once.
19. Write a program to get all the values at once and then convert them into a list.
20. Write a program to get all the keys and values pair in a tuples inside a list.
21. Write a program to check "Name" key is exist or not.
22. Write a program to check "Company" key exist or not in dictionary cart.
23. Write a program to check "Raju" key exist or not in dictionary cart.
24. Write a program to check "address" key is exist or not in dictionary d if yes then print it's value else print "No found" message.
25. Write a program to check "King" key is exist or not in dictionary d if yes then print it's value else print "No found" message.
26. Write a program to check "Hello World" key is exist or not in dictionary d if yes then print it's value else print "Can't search" message.

27. Write a program to check "pin" key is exist if yes then print it's value else insert it with value "300369" in dictionary d.
28. Write a program to check "Done" key is exist if yes then print it's value else insert it with value "Yes" in dictionary d.
29. Write a program to check "Hello" key is exist if yes then print it's value else insert it with value "Bye" in dictionary d.

30. Write a program to insert a new key "King" and value "Queen" in the dictionary d.
31. Write a program to create empty dictionary. Then insert following keys and values at once.
    --- "One" :"One"
    --- "Two" :"Two"
    --- "Three": 3
    --- "Four" : 4
    --- "Three": 5
32. Write a program to insert key "Hello" and value "World" in dictionary d.
33. Write a program to insert key and value pairs ("Earth", "Planet"), ("Moon", "satellite") and ("sun", "star") into the dictionary cart.

34. Write a program to remove the "moon" and "satellite" pair from the dictionary cart.
35. Write a program to remove the pair "sun" and "star" from the dictionary cart.

36. Write a program to remove the pair "True" and "1234 from the dictionary d.
37. Write a program to remove the pair "Done" and "Ye" from the dictionary d.

38. Write a program to remove the last pair of dictionary d.
39. Write a program to remove the last pair of dictionary cart.
40. Write a program to remove the last item of a dictionary cart.
41. Write a program to remove the last item of a mydict dictionary.

42. Write a program to find the Value of Key "Platform" from dictionary cart.
43. Write a program to replace the value of Platform from "YouTube" to "Instagram and YouTube" from dictionary cart.
44. Write a program to check "Hello World" key is exist or not in dictionary d if yes then print it's value else print "Can't search" message.
45. Write a program to check "True" key is exist if yes then print it's value else insert it with value "1234" in dictionary d.
46. Write a program to insert key "Hello" and value "World" in dictionary d.
47. Write a program to delete the dictionary d.
48. Write a program to delete the dictionary cart.
49. Write a program to copy of dictionary mydict.
50. Write a program to delete dictionary mydict.

Set datatype

1. What is set ?
2. Why should we use it ?
3. For which scinario it is best for ?

```
s = {5, 2, 1, 6, "Hello", (100, 11, "King"), 20}

cart = { "Book", "Naruto", 4000789, "Cartoon", "Bazaar"}

MySet = { "Name", "Age", "Address", (1, 2, 3, 4), None, True}
```

1. Write a program to create an empty set.
2. Write a program to make a set with no value.

Use above set.

3. Write a program for following output with and without any function:

   ---{1, 2, 'Bazaar', 5, 6, 'Hello', 'Cartoon', 'Naruto', 20, 4000789, 'Book', (100, 11, 'King')}

4. Write a program for following output with and without any function:

   ---{None, 1, 2, 5, 6, 'Hello', 'Age', (1, 2, 3, 4), 'Name', 20, (100, 11, 'King'), 'Address'}

5. Write a program for following output with and without any function:

   ---{1, 2, 'Bazaar', 'Cartoon', 5, 6, 'Hello', 'Naruto', 20, 4000789, 'Book', (100, 11, 'King')}

6. Write a program for following output with and without any function:

   ---{None, True, 'Bazaar', 'Cartoon', 'Age', (1, 2, 3, 4), 'Name', 'Naruto', 4000789, 'Book', 'Address'}

7. Write a program for following output with and without any function:

   ---{None, True, 2, 5, 6, 'Hello', 'Age', (1, 2, 3, 4), 'Name', 20, (100, 11, 'King'), 'Address'}

   s1 = { 1, 2, 3, 4, 5}

   s2 = {1, 4, 5, 6, 7 }

8. Use above set.
9. Write a program to find the common item in s1 and s2 set with and without any method.

   --- Expected Output : {1, 4, 5}

10. Write a program to find the common item in s and cart set with and without any method.
11. Write a program to find the common item in s and s1 set with and without any method.
12. Write a program to find the common item in MySet and s1 set with and without any method.
13. Write a program to find the what all element present in "s1" but not in "s2" with and without any method.

--- expected output : {2, 3}

14. Write a program to find the what all element present in "s1" but not in "s" with and without any method.
15. Write a program to find the what all element present in "s2" but not in "s1" with and without any method.
16. Write a program to find the what all element present in "cart" but not in "s2" with and without any method.
17. Write a program to find the what all element present in "MySet" but not in "s2" with and without any method.
18. Write a program find items which are not common in both s1 and s2 set with and without any method.

--- Expected output : {2, 3, 6, 7}

19. Write a program to find items which are not common in both s and s1 set with and without any method.
20. Write a program to find items which are not common in both s2 and s2 set with and without any method.
21. Write a program to find items which are not common in both s1 and cart set with and without any method.
22. Write a program to find items which are not common in both s and MySet set with and without any method.
23. Write a program to find items which are not common in both s and s1 set with and without any method.
24. Write a program to find items which are not common in both s1 and s set with and without any method.

25. Write a program to insert "Hello" into the set s1.
26. Write a program to insert "World" into the set s2.
27. Write a program to add 100 and 101 into the set s1 one by one.
28. Write a program to add (1, 2, 3) into the set s1.
29. Write a program to insert 89 into the set s2.

30. Write a program to insert "King", "Queen" and "Jack" at once into a set s1.
31. Write a program to insert 1,2 ,"Hello" and 19 at once into a set s2.
32. Write a program to insert "King", "Arjun" and "S" at once into a set s.
33. Write a program to insert 12, 23, 34 and "Hey" at once into a set s.
34. Write a program to insert 1 and 2 at once into a set s1.
35. Write a program to insert "bye" and "Hi" at once into a set s2.
36. Write a program to insert "Om" and "Rajan" at once into a set s2.

37. Write a program to remove the first item of set s1.
38. Write a program to remove the first item of set s2.
39. Write a program to remove the first item of set s1.
40. Write a program to remove the first item of set s.
41. Write a program to remove the first item of set cart.

42. Write a program to remove 1 from s2.
43. Write a program to remove 23 from set s2.
44. Write a program to remove "bye" from set s2.
45. Write a program to remove "Om" from s2.
46. Write a program to remove "King" from set s.
47. Write a program to remove "Hello" from s2.

48. Write a program to find the what all element present in "s2" but not in "s" with and without any method.
49. Write a program to insert 89 into the set s2.
50. Write a program to insert "Om" and "Rajan" at once into a set s2.
51. Write a program to remove the first item of set cart.
52. Write a program to remove "Hello" from s2.

List comprehension

1. Create a list of number from 1 to 10.
2. Create a list of number from 10 to 1.
3. Create a list of number from -1 to -10.
4. Create a list of number from 1 to 30 and each number have difference of 3 numbers.
5. Create a List of Squares of numbers from 1 to 10
   --- output [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
6. Create a List of Even numbers from 1 to 20
   --- output [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
7. Generate a list of characters from a string
   X = “Helloworld”
   --- output ['H', 'e', 'l', 'l', 'o', 'w', 'o', 'r', 'l', 'd']
8. Create a list of lengths of words in a sentence
   X =”This is a sample sentence”
   --- output [4, 2, 1, 6, 9]
9. Generate a list of tuples containing a number and its square
   --- output [(1, 1), (2, 4), (3, 9), (4, 16), (5, 25)]
10. Create a list of lowercase letters
    --- output ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']
11. Generate a list of uppercase letters
    --- output ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']
12. Create a list of even numbers squared and odd numbers cubed from 1 to 10
    --- output [1, 4, 27, 16, 125, 36, 343, 64, 729, 100]
13. Generate a list of common multiples of 3 and 5 up to 100
    --- output [15, 30, 45, 60, 75, 90]
14. Create a list of reversed strings from another list
    X = ['apple', 'banana', 'cherry']
    --- output ['elppa', 'ananab', 'yrrehc']
15. Generate a list of prime numbers from 1 to 50
    --- output [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47]
16. Create a list of squares of even numbers and cubes of odd numbers from -5 to 5

--- output [-125, 16, -27, 4, -1, 0, 1, 4, 27, 16, 125]

17. Generate a list of strings with their lengths from another list
    X = ['apple', 'banana', 'cherry']
    --- output [('apple', 5), ('banana', 6), ('cherry', 6)]

18. Create a list of first characters from a list of words
    X = ['apple', 'banana', 'cherry']
    --- output ['a', 'b', 'c']

19. Generate a list of numbers with their squares if the number is even
    X = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    --- output [4, 16, 36, 64, 100]

20. Create a list of uppercase words from a sentence
    X = This is a sample sentence.
    --- output ['THIS', 'IS', 'A', 'SAMPLE', 'SENTENCE.']

21. Generate a list of strings with vowels removed
    X = ['apple', 'banana', 'cherry']
    --- output ['ppl', 'bnn', 'chrry']

22. Create a list of numbers that are divisible by both 3 and 5 from 1 to 100
    --- output [15, 30, 45, 60, 75, 90]

23. Generate a list of numbers with their signs reversed
    X = [-2, 3, -5, 7, -11]
    --- output [2, -3, 5, -7, 11]

24. Create a list of words with their lengths from a sentence
    X = This is a sample sentence.
    --- output [('This', 4), ('is', 2), ('a', 1), ('sample', 6), ('sentence.', 9)]

25. Generate a list of positive numbers from another list.
    a = [1, -2, 3, -4, 5, -6]
    --- output [1, 3, 5]

Functions

Use for Function Home Work

1. Write a function that takes 2 number and calculate the sum.
2. Write a function that takes 3 number and calculate the sum.

3. Write a function that takes length and breath and find the area of a rectangle.
4. Write a function accept the number and find the square.
5. Write a function to find the cube of number accepted by the function.
6. Write a function to check the number is divisible of 2 or not.
7. Write a function to check the number is divisible of 11 or not.
8. Write a function to check the number is even or not.
9. Write a function to check the number is odd or not.
10. Write a function to accept n numbers from the user and print sum of all number.
11. Write a function to accept n numbers and print only even.

Lambda/user Define function and Map

1. Write a function to find the Square of each number.

   --- list1 = [12, 2, 5, 2, 8]
2. Write a function to find the cube of each number.

   --- list1 = [12, 2, 5, 2, 8]
3. Write a function to print the square root of each number in a list.

   --- list1 = [12, 2, 5, 2, 8]
4. Write a function to change the case from upper to low of each

   --- value in a list1 = ["hello", "king", "is", "My", "nAme"]
5. Write a function to make first letter of each to upper and rest

   --- should be in lower list1 = ["hEy", "kiNG", "pYTHON"]
6. Write a function to reverse the each word of a given list.

   --- list1 = ["Hello", "Coder", "Python", "Java"]
7. Write a function to swap the case of each letter in a list

   --- list1 = ["hEy", "kiNG", "pYTHON"']
8. Write a function to remove the last letter of each word of given

   --- list1 = ["hello", "King", "Python", "Java", "coder"]
9. Write a function to reverse the each word of each list.

   --- list1 = ["Hello", "King", "Python", "Java", "Coder"]
10. Write a function to find the last digit of given list of numbers.

    --- list1 = [234, 12, 567, 345, 98]
11. Write a function to reverse each number in a list.

--- list1 = [123, 12, 456, 987]

12. Write a function to find the middle letter of each word in a list

--- list1 = ["Hello", "King", "Python", "Java", "C#1"]

13. Write a function to show the ascii value of give values

--- list1 = ['2', '0', "a", "2", "3", "@", '2']

14. Write a program to calculate the sum of all the digit of each value in a list.

--- list1 = [11, 234, 121, 90, 879]

15. Write a function to retrieve the last digit.

--- list1 = [11, 234, 121, 90, 879]

Lambda/ User Define function and filter

16. Write a function to find the all even number in a list or tuple.
    i. --- list1 = [1, 22, 33, 24, 56]
17. Write a function to find all the odd number in the list or tuple.
    i. --- list1 = [3, 4, 2, 6, 3, 1]
18. Write a function to find all the numbers who are all divisible of 5
    i. --- list1 = [23, 45, 40, 23, 25]
19. Write a function to find all the word whose length is even.
    i. --- list1 = ["Hello", "king", "of", "Jungle", "Tiger"]
20. Write a function to find all the word whose last letter is a vowel
    i. --- list1 = ["Hello", "king", "of", "Junge", "Tiger"]
21. Write a function to find all the number which are comes in table of 11.
    i. --- list1= [11, 22, 234, 445, 121]
22. Write a function to find all the palindrome number from the list.
    i. --- list1 = [11, 23, 121, 212, 909]

Lambda/ User Define and Reduce.

23. Write a program to find the minimum value of given list.

--- list1 = [23, 13, 45, 23, 9, 87]

24. Write a program to find the max value of given list.

--- [23, 13, 45, 23, 9, 87]

25. Write a program to find the largest even number from the list.
26. Write a program to calculate the sum of all the number in a list.
27. Write a program to find the string whose length is greater than others.

    --- list1 = ["Hello", "king", "Python", "Java", "coding"]
28. Write a program to find the character whose ascii value is greater among all

    --- list1 = ['2', '0', "a", "2", "3", "@", '2']
29. Write a program to find the character whose ascii value is smallest among all

    --- list1 = ['2', '0', "a", "2", "3", "@", '2']

30. Write a program to merge the 3 list.

    --- list1 = [2, 3,4,5], list2 = [12, 13, 14, 14], list3 = [90, 89, 78, 67]

Recursion Function

1. Write a recursion program to print "Hello" till the system not show "out of memory error".
2. Write a recursion program to print "Hello" only 5 time.
3. Write a recursion program to print number 1 to 5.
4. Write a recursion program to print number from 11 to 18.
5. Write a recursion program to accept number from user and print sequence of that number.

    --- e.g. input 11 and 15 ==> output 11 12 13 14 15.
6. Write a recursion program to print the table of 2.
7. Write a recursion program to print the table of number given by user.
8. Write a recursion function to print the table of 2 in reverse order.
9. Write a recursion function that ask 5 time the user name.
10. Write a recursion function to print the alphabets from "a" to "z".
11. Write a recursion function to print the alphabets from "A" to "Z".
12. Write a recursion function to print each character of "Hello" one by one.
13. Write a recursion function to print each element of a list ==> [1,4, 2, 6, "Hello"]

14. Write a recursion function to print each element of a list in reverse order ==> [1,4, 2, 6, "Hello"]
15. Write a recursion function to count the number of elements of a list ==> [1,4, 2, 6, "Hello"]
16. Write a recursion function to count each element in a "Hello world" string.
17. Write a recursion function to print length of each string in list ==> ["King", "Naruto", "Vijay", "King", "World", "Hello"]
18. Write a recursion function to print index location of each character of a string = "Hello, World".
19. Write a recursion function to print index and value of each string in list ==> ["King", "Naruto", "Vijay", "King", "World", "Hello"]
20. Write a recursion function to print the ascii value of the string == > "King Naruto 12"
21. Write a recursion function that user for user name and password again and again till they enter a right name and password.

    --- dictionaryOfUsername = {"user1": "pass123", "user2":"pass567"}
22. Write a recursion function to calculate a sum of 1 to 9.
23. Write a recursion function that accept the number from user and print the sum of 1 to n (here n is user given number).
24. Write a recursion function to calculate the factorial of 5.
25. Write a recursion function that accept number from user and calculate it's factorial.
26. Write a recursion function to print the 10th fibonacci number.
27. Write a recursion function that accept input as number (i.e. index location of fibonacci number) and print that fibonacci number.
28. Write a recursion function to reverse a number using Recursion.

OOPs

1.Shape class for different shapes.

a. There should be a base class which have.

Attribute: Name of shape.

Methods: Area, Parameter, details of shape

b. A rectangle class which inherit the shape class for its method.

Attributes: Length and breadth.
Method : Area, Parameter, details.

c. A rectangle class which acquires the properties of shape.
Attributes : side
Method : Area, Parameter, details.

d. A circle class.
Attributes : radius, Pi
Method : area, circumference, details.

e. Triangle class (Derived from Shape)
Attributes: side1, side2, side3
Methods: area, perimeter, display

2. Create a Students marks management system.

a) Store all the data into a list.
b) Add a new student
c) Search a student base on roll number.
d) Show all the students details stored.
e) Delete the student details with respect to roll number.
f) Update the student marks with respect to roll number.

Beginner-Level Projects

1. Student Management System

- Concepts Should Covered:

Encapsulation, Basic Inheritance, Abstraction

- Description:

Manage student records such as names, grades, and courses. You can perform basic CRUD operations (Create, Read, Update, Delete) for student data.

- Features:

a. Create a `Student` class with attributes like `name`, `id`, `grades`.
b. Implement methods to add, update, and display student data.
c. Optionally, implement simple inheritance with subclasses like `GraduateStudent` or `UndergraduateStudent`.

2. Library Management System

- Concepts Should Covered:

  Encapsulation, Basic Inheritance, Polymorphism

- Description:

  Build a system to manage books, patrons, and borrow/return functionality. A simple text-` based system to keep track of library items.

- Features:

  a. `Book` class with attributes like `title`, `author`, and `ISBN`.
  b. `Patron` class to manage users.
  c. Implement methods for borrowing and returning books.
  d. Optionally, extend the `Book` class to different types, such as `Magazine` and `Ebook`.

3. Banking System

- Concepts Should Covered:

  Encapsulation, Basic Inheritance

- Description:

  A simple banking system that allows users to create accounts, deposit and withdraw money, and transfer funds.

- Features:

  a. `Account` class with attributes like `account_number`, `balance`, `account_type`.
  b. Methods to deposit, withdraw, and transfer money between accounts.
  c. Optionally, add a `SavingsAccount` subclass with additional features like interest calculation.

Intermediate-Level Projects

4. E-commerce Store

- Concepts Should Covered:

  Inheritance, Polymorphism, Abstraction

- Description:

  Build an e-commerce application with products, categories, and shopping carts. Implement discounts, pricing strategies, and simple order management.

- Features:

    a. `Product` class with attributes like `name`, `price`, and `category`.
    b. Subclasses like `DiscountedProduct` or `SpecialOfferProduct` to handle different pricing strategies.
    c. `Cart` class to hold items and calculate the total cost, with methods to add/remove products.
    d. Optionally, implement simple checkout and order confirmation functionalities.

- Complexity:

    Intermediate-level, includes polymorphism (for different product types) and some abstraction (order handling).

5. Hotel Reservation System

- Concepts Should Covered:

    Encapsulation, Inheritance, Polymorphism, Abstraction

- Description:

    A hotel reservation system to book rooms, track availability, and manage customer reservations. Includes different room types and pricing strategies.

- Features:

    a. `Room` class with attributes like `room_number`, `room_type`, `price_per_night`.
    b. Subclasses like `StandardRoom`, `SuiteRoom`, and `LuxuryRoom`.
    c. `Reservation` class to track customer bookings and payment status.
    d. Polymorphism to calculate prices (e.g., with discounts or seasonal rates).

6. Inventory Management System

- Concepts Should Covered:

    Inheritance, Polymorphism, Abstraction

- Description:

    Create a system for tracking inventory, orders, and stock levels. Use different product types and manage order fulfillment.

- Features:

    a. `Product` class with attributes like `name`, `SKU`, `price`, and `stock_quantity`.
    b. Subclasses for different product categories (e.g., `PerishableProduct` with expiry dates).
    c. `Order` class to manage customer orders and track inventory levels.
    d. Polymorphism for pricing or stock updates, depending on the type of product.

Advanced-Level Projects

7. Social Media System

- Concepts Should Covered:

  Encapsulation, Inheritance, Polymorphism, Abstraction, Design Patterns

- Description:

  Build a social media platform that allows users to create profiles, post messages, follow other users, and like posts.

- Features:

  a. `User` class with attributes like `username`, `email`, `followers_count`.
  b. `Post` class to manage posts with methods like `like`, `comment`, and `delete`.
  c. Subclasses for different types of posts (e.g., `TextPost`, `ImagePost`).
  d. Implement polymorphism for liking and interacting with different types of posts.
  e. Use abstraction for user and post management.

8. Weather Monitoring System

- Concepts Should Covered:

  Encapsulation, Inheritance, Polymorphism, Abstraction, External API Integration

- Description:

  Design a weather monitoring system that tracks weather conditions like temperature, humidity, and pressure. Use different types of weather stations to collect data.

- Features:

  a. `WeatherStation` class with attributes like `location`, `temperature`, `humidity`, and `pressure`.
  b. Subclasses for different types of stations (e.g., `UrbanWeatherStation`, `RuralWeatherStation`).
  c. Polymorphism to define how each station collects and displays weather data.
  d. Integrate an external weather API (e.g., OpenWeatherMap) to fetch real-time data.
  e. Use abstraction to manage weather data processing without exposing implementation details.

9. Banking System with Multiple Features

- Concepts Should Covered:

  Encapsulation, Inheritance, Polymorphism, Abstraction, Real-World Modeling

- Description:

  Expand the basic banking system to include different account types, transaction history, and security features like authentication.

- Features:

  a. `Account` class with additional methods for transaction history, and an `Authentication` class for managing user login.
  b. Add multiple account types (e.g., `SavingsAccount`, `CheckingAccount`, `CreditAccount`).
  c. Implement polymorphism for applying different transaction rules (e.g., withdrawal limits for different account types).
  d. Abstraction to hide implementation details of account transactions.