

Rays Edutech Private Limited

APPLICATIONS BASED On SIMPLE WHILE AND FOR STRUCTURE

- Wap in "Python" language to display "Python is General Purpose language" five times.
- Wap in "Python" language to display "I like 'Python' very much!" three times and each after two lines.
- Wap in "Python" language to display "I like 'Python' very much!" continuously until a zero is pressed.
- Wap in "Python" language to display the counting starting from 1 and up to 30.
- Wap in "Python" language to display the following format: -

1 2 3	Up to 19
1 3 5	
1 4 7	Up to 19
	•
19 18 17	Up to 1
19 17 15	=
19 16 13	
2 4 8	Up to 20
3 6 9	
4 8 12	Up to 40
	1
20 18 16	Up to 2
30 27 23	

1	4 5	,	Op to 100
1	8 2	27 <mark></mark>	Up to 1000
1	16	81	Up to 10000

40 38 36......Up to 4

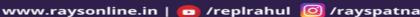
- 10 100...... Up to 1000000000
- 123456789
- 987654321
- Wap in "Python" language to accept thirty numbers calculate and display the sum, product and average value.
- Wap in "Python" language to accept N numbers calculate and display the sum, product and average value.



Rays Edutech Private Limited

- Wap in "Python" language to accept N numbers calculate and display the total count of even and odd numbers.
- Wap in "Python" language to accept N numbers calculate and display the total count of $+^{ve}$ even and -ve odd numbers.
- 10. Wap in "Python" language to accept N characters one-by-one calculate and display the total count of digits and alphabets.
- 11. Wap in "Python" language to accept N numbers check and display the smallest and largest value.
- 12. Wap in "Python" language to accept a number calculates and display the total count of digits.
- 13. Wap in "Python" language to accept a number calculates and display the sum, product and average of the digits.
- 14. Wap in "Python" language to generate even series from 1 to 50.
- 15. Wap in "Python" language to generate odd series from 1 to 50.
- 16. Wap in "Python" language to accept a number, display it in reverse order
- 17. Wap in "Python" language to generate a table of any number.
- 18. Wap in "Python" language to accept a number, check and display whether the number is prime or not.
- 19. Wap in "Python" language to accept initial and final position and find the prime numbers between the initial and final position.
- 20. Wap in "Python" language to accept a number, check and display message whether it perfect number or not.
- 21. Wap in "Python" language to accept a number, check and display whether the number is Armstrong or not.
- 22. Wap in "Python" language to accept initial and final position, print Armstrong number between initial and final position.
- 23. Wap in "Python" language to accept a number and display its factorial value.
- 28. Wap in "Python" language to accept two numbers check and display the Highest Common Factor or Greatest Common Factor
- 29. Wap in "Python" language to accept two numbers check and display the Lowest Common Factor.
- 30. Wap in "Python" language to display 20 terms of Fibonacci series. (i.e 0,1,1,2,3,.).
- Wap in "Python" language to accept a positive integer value, determine and print its binary equivalent.
- 32. Wap in "Python" language to accept a positive value, convert into hexadecimal equivalent.







Rays Edutech Private Limited

- 33. Wap in "Python" language to display the total count of Leap Years between 1000 and 2009.
- 34. Wap in "Python" language to calculate and display the sum of the following series: -

• 1 + 2 + 3 + +	Up to N terms.
• $1^2 + 2^2 + 3^2 + \dots +$	Up to N terms.
• $1 + X^2 + X^3 + \dots +$	Up to N terms.
• $1 + 1/X + 1/X^2 + \dots +$	Up to N terms.
• $X + X^2/2! + X^3/3! + \dots +$	Up to N terms.
• $X - X^3/3! + X^5/5! - \dots +$	Up to N terms.
• $X^2/2! - X^4/4! + X^6/6! - \dots +$	Up to N terms.
• $1 - X^2/2! + X^4/4! - X^6/6! + \dots$	Up to N terms.

35. Wap in "Python" language to calculate and display the Fibonacci series up to n terms.

[Hint: **0 1 1 2 3 5 8 13 21**]

36. Write a program in "Python" to generate the following given series:-

a.	1	b. 1	c. 11111	d. 5	e. 0	f. 12345
	12	21	1111	55	101	23451
	123	321	1 11	555	21012	34512
	1234	4321	11	5555	3210123	45123
	12345	54321	1	55555	43210123	4 51234
h.		I.		J.		K.
	*		*		*	0
	**		**		***	10
	***		***		****	010
	***		****	, and	*****	1010
	*****	*	****		*****	01010