

List placeholder

list01.

a o'zgaruvchiga **list** turidagi qiymatni taminlab, **a** ning turini chiqaring.

Output: type

Input:	Output:
	<code><class 'list'></code>

list02.

a o'zgaruvchiga bo'sh listni ta'minlab natijani chiqaring.

Input: *a* .

Output: list.

Input:	Output:
	<code>[]</code>

List03.

N ta elementda iborat bo'lgan va qiymatlari 0 dan tashkil topgan **list** yarating va natijani chiqaring.

Input: *N (int)*.

Output: list.

Input:	Output:
5	<code>[0, 0, 0, 0, 0]</code>

List04.

N ta elementda iborat bo'lgan va qiymatlari 1 dan tashkil topgan **list** yarating va natijani chiqaring.

Input: N (*int*).

Output: list.

Input:	Output:
5	[1, 1, 1, 1, 1]

List05.

N ta elementda iborat bo'lgan va qiymatlari juft sonlarda tashkil topgan **list** yarating va natijani chiqaring.

Input: N (*int*).

Output: list.

Input:	Output:
5	[2, 4, 6, 8, 10]

List06.

N ta elementda iborat bo'lgan va qiymatlari manfiy sonlarda tashkil topgan **list** yarating va natijani chiqaring.

Input: N (*int*).

Output: list.

Input:	Output:
5	[-1, -2, -3, -4, -5]

List07.

N ta elementda iborat bo'lgan va qiymatlari musbat sonlarda tashkil topgan **list** yarating va natijani chiqaring. List elementlari **k** dan boshlab 2 qadamdan oshib borsin.

Input: *N, k. (int)*

Output: list.

Input:	Output:
5 5	[5, 7, 9, 11, 13]

List08.

N ta elementda iborat bo'lgan **list** yarating:

- list elementlari turi (*type*) turli xil bo'lsin.

Input: *N (int).*

Output: list.

Input:	Output:
5	[1, 2.0, 3, 4.0, 5]

List09.

N ta elementda iborat bo'lgan **list** yarating:

- list elementlari juft indexda **str** turiga mansub bo'lsin.
- list elementlari toq indexda **int** turiga mansub bo'lsin.

Input: *N (int)*.

Output: list.

Input:	Output:
6	['0', 1, '2', 3, '4', 5]

List10.

N ta elementda iborat bo'lgan **list** yarating:

- list elementlari 2 ning darajalaridan iborat bo'lsin
- list elementlari o'sish tartibida joylashtiring.

Input: *N (int)*.

Output: list.

Input:	Output:
3	[1, 2, 4]

List11.

N ta elementda iborat bo'lgan **list** yarating:

- list elementlari 2 ning darajalaridan iborat bo'lsin
- list elementlari kamayish tartibida joylashtiring.

Input: N (*int*).

Output: list.

Input:	Output:
5	[16, 8, 4, 2, 1]

List12.

N ta elementda iborat bo'lgan **list** yarating:

- list elementlari 0 dan N gacha bo'lgan haqiqiy sonlarda iborat bo'lsin
- list elementlari 1 ga oshib borsin.

Input: N (*int*).

Output: list.

Input:	Output:
5	[0.0, 1.0, 2.0, 3.0, 4.0]

List13.

N ta elementda iborat bo'lgan **list** yarating:

- list elementlari haqiqiy sonlarda iborat bo'lsin
- list elementlari K ga oshib borsin.

Input: N, K (*int*).

Output: list.

Input:	Output:
5 3	[3.0, 6.0, 9.0, 12.0, 15.0]

List14.

N son berilgan shu soning raqamlaridan iborat bo'lgan list yarating

Input: N (*int*).

Output: list.

Input:	Output:
123	[1, 2, 3]

List index and Slicing

ListIndex1.

L1 list berilgan.

- L1 listning birinchi elementini toping: **first_iteam**

Input: *L1 (str)*.

Output: first_iteam (*int*).

Input:	Output:
3,6,5,2	3

ListIndex2.

L1 list berilgan.

- L1 listning birinchi elementini toping: **first_iteam**
- L1 listning oxirgi elementini toping: **last_iteam**

Input: *L1 (str)*.

Output: first_iteam, last_iteam (*int*).

Input:	Output:
2,4,3,1,5,6	2 6

ListIndex3.

L1 list berilgan.

- L1 listning birinchi va oxirgi elementlar yig'indisini toping.

Input: *L1 (str)*.

Output: Natijani chiqaring (*int*).

Input:	Output:
9,5,4,3,6,2	11

ListIndex4.

L1 list berilgan.

- L1 listning birinchi va oxirgi elementlarning kattasini toping.

Input: *L1 (str)*.

Output: Natijani chiqaring (*int*).

Input:	Output:
6,7,9,4,3,8	8

ListIndex5.

L1 list berilgan.

- L1 listning boshidan 3 ta elementini chiqaring.

Input: *L1 (str)*.

Output: Natijani chiqaring.

Input:	Output:
a,v,f,d,s,q,w,e	['a', 'v', 'f']

ListIndex6.

L1 list berilgan.

- L1 listning oxiridagi 4 ta elementini chiqaring.

Input: $L1$ (*str*).

Output: Natijani chiqaring.

Input:	Output:
1,2,7,4,6,3,1	[4, 6, 3, 1]

ListIndex7.

L1 list berilgan.

- L1 listning N index dan K index gacha bo'lgan elementlarini chiqaring.

Input: $L1, N, K$ ($0 \leq N, K < \text{len}(L1)$).

Output: Natijani chiqaring.

Input:	Output:
5,3,4,34,100 1 3	[3, 4]

ListIndex8.

L1 list berilgan.

- L1 listning elementlarning xar 3 ta elementlarini chiqaring.

Input: $L1$ (*str*).

Output: Natijani chiqaring.

Input:	Output:
1,2,3,4,5,6,7	[1, 4, 7]

ListIndex9.

L1 list berilgan.

- L1 listning elementlarning oxiridan xar N ta elementlarini chiqaring.

Input: $L1, N$

Output: natijani chiqaring.

Input:	Output:
1,2,3,4,5 2	[5, 3, 1]

List element counting

ListCounting1.

L1 list berilgan.

- L1 list elementlari ichidan eng kattasini toping.

Input: *L1 (str)*.

Output: Natijani chiqaring (*int*).

Input:	Output:
1, 2, 3, 4, 5	5

ListCounting2.

L1 list berilgan.

- L1 listning elementlarning eng kichik elementini toping.

Input: *L1 (str)*.

Output: Natijani chiqaring (*int*).

Input:	Output:
1, 2, 3, 4, 5	1

ListCounting3.

L1 list berilgan.

- L1 listning elementlarning eng kichik elementini toping: minimum
- L1 listning elementlarning eng katta elementini toping: maximum

Input: *L1 (str)*.

Output: minimum,maximum (*int*).

Input:	Output:
3,2,5,7,8,9	2 9

ListCounting4.

L1 list berilgan.

- L1 listning elementlarning eng katta elementining indexni toping.

Input: *L1 (str)*.

Output: Natijani chiqaring (*int*).

Input:	Output:
3,4,9,3,5,6	2
6,5,4,2,5,7,1	5

ListCounting5.

L1 list berilgan.

- L1 listning juft elementlarini toping.

Input: *L1* (*str*).

Output: Natijani chiqaring (*list*).

Input:	Output:
3,4,9,3,5,6	[4, 6]
2,3,48,6,1,7	[2, 48, 6]