Python sort() sort() list.sort(cmp=None, key=None, reverse=False) cmp -- , key -- reverse -- reverse = True reverse = False sort() aList = ['123', 'Google', 'Runoob', 'Taobao', 'Facebook']; aList.sort();print("List : ")print(aList) List : ['123', 'Facebook', 'Google', 'Runoob', 'Taobao'] vowels = ['e', 'a', 'u', 'o', 'i'] vowels.sort(reverse=True) print(':')print(vowels) :['u', 'o', 'i', 'e', 'a'] def takeSecond(elem): return elem[1] random = [(2, 2), (3, 4), (4, 1), (1, 3)] random.sort(key=takeSecond) print('')print(random) [(4, 1), (2, 2), (1, 3), (3, 4)] Python Python 0 1 Python 6 Python Python list1 = ['Google', 'Runoob', 1997, 2000]list2 = [1, 2, 3, 4, 5]list3 = ["a", "b", "c", "d"]list4 = ['red', 'green', 'blue', 'yellow', 'white', 'black'] 0 1 #!/usr/bin/python3list = ['red', 'green', 'blue', 'yellow', 'white', 'black']print(list[0])print(list[1])print(list[2]) redgreenblue -1 -2 #!/usr/bin/python3list = ['red', 'green', 'blue', 'yellow', 'white', 'black']print(list[-1])print(list[-2])print(list[-3]) blackwhiteyellow [] #!/usr/bin/python3nums = [10, 20, 30, 40, 50, 60, 70, 80, 90]print(nums[0:4]) [10, 20, 30, 40] #!/usr/bin/python3list = ['Google', 'Runoob', "Zhihu", "Taobao", "Wiki"]# print ("list[1]: ", list[1])# print ("list[1:-2]: ", list[1:-2]) list[1]: Runooblist[1:-2]: ['Google', 'Zhihu'] append() #!/usr/bin/python3list = ['Google', 'Runoob', 1997, 2000]print (" : ", list[2])list[2] = 2001print (" : ", list[2])list[2] = 2001print (" : ", list[2])list1 = ['Google', 'Runoob', 'Taobao']list1.append('Baidu')print (" : ", list1) append() : 1997 : 2001 : ['Google', 'Runoob', 'Taobao', 'Baidu'] del #!/usr/bin/python3list = ['Google', 'Runoob', 1997, 2000]print (" : ", list)del list[2]print (" : ", list) : ['Google', 'Runoob', 1997, 2000] : ['Google', 'Runoob', 2000] remove() Python + * + * Python len([1, 2, 3]]3 [1, 2, 3] + [4, 5, 6][1, 2, 3, 4, 5, 6] ['Hi!'] * 4['Hi!', 'Hi!', 'Hi!', 'Hi!'] 3 in [1, 2, 3]True for x in [1, 2, 3]: print(x, end=" ")1 2 3 Python Python L=['Google', 'Runoob', 'Taobao'] Python L[2]Taobao' L[-2]'Runoob': count from the right L[1:]['Runoob', 'Taobao'] >>> L=['Google', 'Runoob', 'Taobao']>>> L[2]'Taobao'>>> L[-2]'Runoob'>>> L[1:]['Runoob', 'Taobao']>>> >>> squares = [1, 4, 9, 16, 25]>>> squares += [36, 49, 64, 81, 100]>>> squares[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]>>> >>> a = ['a', 'b', 'c']>>> n = [1, 2, 3]>>> x = [a, n]>>> x[['a', 'b', 'c'], [1, 2, 3]]>>> x[0]['a', 'b', 'c']>>> x[0][1]'b' operator eq Python operator # operator import operatora = [1, 2]b = [2, 3]c = [2, 3]print("operator.eq(a,b): ", operator.eq(a,b))print("operator.eq(c,b): ", operator.eq(c,b)) operator.eq(a,b): Falseoperator.eq(c,b): True Python& Python: Python: >>>range(10) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]>>> range(1, 11) [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]>>> range(0, 30, 5) [0, 5, 10, 15, 20, 25]>>> range(0, 10, 3) [0, 3, 6, 9]>>> range(0, -10, -1) [0, -1, -2, -3, -4, -5, -6, -7, -8, -9]>>> range(0)[]>>> range(1, 0)[] Markdown (*)(+)(-) * * * + + + - - - - . 1. 2. 3. Markdown 1. 3. 27. 3 5. 6. 7. - - - - - - - - - 1. 1. 2. 2. 1. 2. 3. 1. - A - B 1. 1 2. 2 - C2. 1. - - 2. - - 2-4 2 3 GitHub Markdown - [ ] - [x] - [ ] ## ### - [x] - [x] - [ ] UI ### - [ ] - [x] - [ ] - [ ] - [ ] - [ ] API - [ ] ### - [ ] - [ ] - [ ] x [ ][x] 1. 2. > - - Markdown Linux Linux ls list directory contents) ls [-alrtAFR] [name...] : -a --all . -A --almost-all . .. -l-h --human-readable -l KBMB-t-r --reverse -t-S -S-R --recursive-F --classify / * --color--color=auto-i --inode inode -n --numeric-uid-gid UID GID-d --directory -l-1-m-Q --quote-name--group-directories-first-time-style= +%Y-%m-%d ls -l # ls -a # ls -lh # ls -t # ls -R # ls -l /etc/passwd # /etc/passwd ls -la ls -lShr /var/log ls -lhR /var/log ls -ld /etc ls -ltr # ls /bin dev lib media net root srv upload wwwboot etc lib64 misc opt sbin sys usrhome lost+found mnt proc selinux tmp var /bin : ls -lR /bin \ ls "my file.txt" # "my file.txt"ls my\ file.txt # "my file.txt"ls -- --filename # "-filename" ls * ? [...] ls *.txt # .txtls file?.txt # file?.txt?ls [abc]*.txt # abc.txt s : ls -ltr s* ls -l - d l c b s p ls -l 9 rwx r w x - -rw-r--r-- 1 user group 4096 Feb 21 12:00 file.txt file.txt ls -lt | head -5 5 ls | wc -l () ls --color ls ls Linux ls man ls Linux C++ C++ STL #include std::list mylist; T mylist.push_back(value); mylist.pop_back(); mylist.erase(iterator); mylist.front(); mylist.back(); for (auto it = mylist.begin(); it != mylist.end(); ++it) #include #include int main() { std::list lst1; // list std::list lst2(5); // 5list std::list lst3(5, 10); // 510 std::list lst4 = {1, 2, 3, 4}; // return 0;} #include #include int main() { // std::list numbers; // numbers.push_back(10); numbers.push_back(20); numbers.push_back(30); // std::cout

- http://katyababash.com/images/uploads/file/699e5585-5c6d-4366-898f-d780fc048d8f.pdf
- http://groupkaishan.com/d/files/3d11f56a-c113-4d45-a81c-5c5ec30d8bef.pdf
- what to put on an altar for day of the dead
- do you route clockwise or counterclockwise
- types of metal cutting machine
- http://saringkarnwood.com/UserFiles/file/4d3c6fb5-8304-41c2-82c5-5d0faa70f413.pdf
- mabogomifa
- http://pwr-tech.ru/userfiles/file/padijukepo_mibibelipokikez_wipew.pdf
- https://jovanrajsic.net/storage/files/36834631120.pdf
- qixuhe
- what is allusion in literature and examples