

I'm not a bot























```
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 enum C# enum { enumeration list }; enum name enumeration list 0. enum Days { Sun, Mon, tue, Wed, thu, Fri, Sat }; using System;public class EnumTest{ enum Day { Sun, Mon, Tue, Wed, Thu, Fri, Sat }; static void Main() { int x = (int)Day.Sun; int y = (int)Day.Fri; Console.WriteLine("Sun = {0}", x); Console.WriteLine("Fri = {0}", y); } } Sun = 0Fri = 5 Python - 01 Python6 Python Python list1 = ['physics', 'chemistry', 1997, 2000]list2 = [1, 2, 3, 4, 5]list3 = ["a", "b", "c", "d"] 0 list1 = ['physics', 'chemistry', 1997, 2000]list2 = [1, 2, 3, 4, 5, 6, 7] print "list1[0]:" , list1[0]print "list2[1:5]:" , list2[1:5] list1[0]; physicslist2[1:5]; [2, 3, 4, 5] append() list = [] list.append('Google') list.append('Runoob')print list append() ['Google', 'Runoob'] del list1 = ['physics', 'chemistry', 1997, 2000] print list1 del list1[2]print "After deleting value at index 2 : "print list1 ['physics', 'chemistry', 1997, 2000]After deleting value at index 2 : ['physics', 'chemistry', 2000] remove() Python + * + * Python len(1, 2, 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,1 2 3 Python Python >>>L = ['Google', 'Runoob', 'Taobao']>>> L[2]Taobao>>> L[2]Taobao>>> L[1:]Runoob, 'Taobao']>>> Python L[2]Taobao L[1:]Runoob, 'Taobao'] Python& Python: Python. java Gson Google Java Java JSON JSON Java Gson API Java GsonJSONObjectJSON JSONObjectJSON JSONArrayJSON Gson Java JSON RESTful API Gson Java JSON Gson Gson Gson Maven pom.xml com.google.code.gson gson 2.8.9 Java JSON import com.google.gson.Gson;public class GsonDemo { public static void main(String[] args) { Gson gson = new Gson(); // JavajSON Person person = new Person("", 30); String json = gson.toJson(person); System.out.println(json); // : {"name":"","age":30} //JSONJava Person parsedPerson = gson.fromJson(json, Person.class); System.out.println(parsedPerson.getName()); // : } }class Person { private String name; private int age; // public Person(String name, int age) { this.name = name; this.age = age; } // getter (Gsongetter/setter) public String getName() { return name; } public int getAge() { return age; } } Gson Java JSON class User { private String name; private int age; // getter setter } User user = new User("", 25);String json = gson.toJson(user); // : {"name":"","age":25} JSON Java String json = "{\n\"name\": \"\", \"age\": 30}"; User user = gson.fromJson(json, User.class); System.out.println(user.getName()); // : // List people = Arrays.asList( new Person("", 28), new Person("", 32));String jsonList = gson.toJson(people); // Type listType = new TypeToken(){}.getType();List parsedPeople = gson.fromJson(jsonList, listType); Gson gson = new GsonBuilder().setDateFormat("yyyy-MM-dd HH:mm:ss").create();Date now = new Date();String dateJson = gson.toJson(now);Date parsedDate = gson.fromJson(dateJson, Date.class); String json = "{\n\"name\": \"\", \"age\": 25}";JSONObject element = JsonParser.parseString(json);JSONObject object = element.getAsJsonObject();String name = object.get("name").getAsString();int age = object.get("age").getAsInt(); Gson gson = new GsonBuilder().setPrettyPrinting().create(); Gson gson = new GsonBuilder().serializeNulls() // null .create(); Gson gson = new GsonBuilder().setFieldNamingPolicy(FieldNamingPolicy.LOWER_CASE_WITH_UNDERSCORES).create(); // userNameuser_name Gson Gson class Order { private String orderId; private List products; // } Order order = new Order("123", Arrays.asList( new Product("", 1, 2999.00), new Product("", 2, 199.00)));String json = gson.toJson(order); TypeToken TypeToken String json = "{\n\"name\": \"\", \"price\": 5.5}, {\n\"name\": \"\", \"price\": 3.2}";Type productListType = new TypeToken(){}.getType();List products = gson.fromJson(json, productListType); JsonSerializer jsonDeserializer implements JsonSerializer { private static final SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd"); @Override public JsonElement serialize(Date date, Type type, JsonSerializationContext context) { return new JsonPrimitive(format.format(date)); } // } Gson gson = new GsonBuilder().registerTypeAdapter(Date.class, new DateSerializer().create()); Gson class User { @SerializedName("user_name") // JSON private String name; @Expose(serialize = false) // private String password; @Since(1.0) // private String email; } GsonBuilder GsonBuilder Gson gson = new GsonBuilder().setPrettyPrinting() // .serializeNulls() // null .setDateFormat("yyyy-MM-dd") // .disableHtmlEscaping() // HTML .create(); Gson Gson APIJsonReader/JsonWriter Gson JSON main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Google"); sites.add("Runoob"); sites.add("Taobao"); sites.add("Weibo"); System.out.println(sites.get(1)); // } } 0 Runoob ArrayList set() set(int index, E element) indextelement import java.util.ArrayList;public class RunoobTest { public static void main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Google"); sites.add("Runoob"); sites.add("Taobao"); sites.add("Weibo"); sites.add("Wiki, Weibo) ArrayList remove() import java.util.ArrayList;public class RunoobTest { public static void main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Google"); sites.add("Runoob"); sites.add("Taobao"); sites.add("Weibo"); sites.remove(3); // System.out.println(sites); } } [Google, Runoob, Taobao] ArrayList size() import java.util.ArrayList;public class RunoobTest { public static void main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Google"); sites.add("Runoob"); sites.add("Taobao"); sites.add("Weibo"); System.out.println(sites.size()); } } 4 for import java.util.ArrayList;public class RunoobTest { public static void main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Google"); sites.add("Runoob"); sites.add("Taobao"); sites.add("Weibo"); for (int i = 0; i < sites.size(); i++) { System.out.println(sites.get(i)); } } } GoogleRunoobTaobaoWeibo for-each import java.util.ArrayList;public class RunoobTest { public static void main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Google"); sites.add("Runoob"); sites.add("Taobao"); sites.add("Weibo"); for (String i : sites) { System.out.println(i); } } } GoogleRunoobTaobaoWeibo ArrayList String booleanBoolean byteByte shortShort intInteger longLong floatFloat doubleDouble charCharacter BigIntegerBigDecimal BigInteger ArrayList li=new ArrayList(); // ArrayList li=new ArrayList(); // ArrayList ( Integer ) import java.util.ArrayList;public class RunoobTest { public static void main(String[] args) { ArrayList myNumbers = new ArrayList(); myNumbers.add(10); myNumbers.add(15); myNumbers.add(20); myNumbers.add(25); for (int i : myNumbers) { System.out.println(i); } } } 10152025 ArrayList Collections java.util sort() import java.util.ArrayList;import java.util.Collections; // Collections public class RunoobTest { public static void main(String[] args) { ArrayList sites = new ArrayList(); sites.add("Taobao"); sites.add("Wiki"); sites.add("Runoob"); sites.add("Weibo"); Collections.sort(sites); // for (String i : sites) { System.out.println(i); } } } GoogleRunoobTaobaoWeiboWiki import java.util.ArrayList;import java.util.Collections; // Collections public class RunoobTest { public static void main(String[] args) { ArrayList myNumbers = new ArrayList(); myNumbers.add(15); myNumbers.add(20); myNumbers.add(34); myNumbers.add(8); myNumbers.add(12); Collections.sort(myNumbers); // for (int i : myNumbers) { System.out.println(i); } } } 8121520334 Java ArrayList Java ArrayList API Java Python3 list = [10, 11, 12, 13, 14, 15] ; [15, 14, 13, 12, 11, 10] def Reverse(lst): return [ele for ele in reversed(lst)] lst = [10, 11, 12, 13, 14, 15] print(Reverse(lst)) [15, 14, 13, 12, 11, 10] def Reverse(lst): new_lst = lst[::-1] return new_lst lst = [10, 11, 12, 13, 14, 15] print(Reverse(lst)) [15, 14, 13, 12, 11, 10] Python3 C++ C++ STL #include <list> ; 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
```