

原创

cJSON的使用方法

2016-11-10 21:56:15

Rotation.

阅读数 33164

☆ 收藏

更多

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。
本文链接：<https://blog.csdn.net/fengxinlinux/article/details/53121287>

首先，要介绍一下什么是json格式。

一.JSON格式简述

[JSON](#)(JavaScript Object Notation) 是一种轻量级的数据交换格式。易于人阅读和编写，同时也易于机器解析和生成。它基于JavaScript（Standard ECMA-262 3rd Edition, December 1999）的一个子集。JSON采用完全独立于语言的文本格式，但是也使用了类似于C语言家族的习惯（包括C, C++, C#, Java, JavaScript, Perl, Python等）。这成为理想的数据交换语言。（来自“开源中国”资料）。

cJSON从名字可知，整个项目都是以极标准的C来写的，意思说，可以跨各种平台使用了。[cJSON](#)是一个超轻巧，携带方便，单文件，简单的可以作为ANSI-C标准的JSON解

cJSON 开源项目位置：[点击打开链接](#)

更加详细的解释和示例请查看 <http://www.json.org/> 主页。

cJSON，目前来说，就只有两个文件，一个cJSON.c 一个cJSON.h文件。使用的时候，自己创建好一个main.c文件后，将头文件include进去。

如果是在linux pc上，请使用以下命令进行编译：

```
gcc *.c cJSON.c -lm
```

记得编译时末尾链接libm库。

二.JSON结构体

熟悉使用cJSON库函数可从cJSON结构体入手，cJSON结构体如下所示：

```
1 typedef struct cJSON {
2     struct cJSON *next,*prev;
3     struct cJSON *child;
4     int type;
5     char *valuestring;
6     int valueint;
7     double valuedouble;
8     char *string;
9 } cJSON;
```

几点说明

- 1.cJOSN结构体为一个双向列表，并可通过child指针访问下一层。
- 2.type变量决定数据项类型（键的类型），数据项可以是字符串可以是整形，也可以是浮点型。如果是整形值的话可从valueint，如果是浮点型的话可从valuedouble取出，

3.string可理解为节点的名称，综合此处的第2点可理解为“键”的名称。

cJSON作为Json格式的解析库，其主要功能无非就是构建和解析Json格式了，用途就是一端将要发送的数据已cjson形式封装，然后发送，另一端收到此数据后，再解析，得到想要的数



24



12



，还是按cjsr

三.封装成json形式

接下来直接通过几个例子代码，——解析。

第一，创建json数据串。这数据串，可能是对象，也可能是数组，也可能是它们的各种组合，其中再加上一些键值对。有一点要先说明：它们的组合，符合JSON的继承格式一据串的特点。

<1>创建一个对象，并向这个对象里添加字符串和整型键值：

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<string.h>
4  #include"cJSON.h"
5
6  int main()
7  {
8      cJSON * usr;
9      cJSON *arry;
10
11     usr=cJSON_CreateObject(); //创建根数据对象
12     cJSON_AddStringToObject(usr,"name","fengxin"); //加入键值，加字符串
13     cJSON_AddStringToObject(usr,"passwd","123");
14     cJSON_AddNumberToObject(usr,"num",1); //加整数
15
16     char *out = cJSON_Print(usr); //将json形式打印成正常字符串形式
17     printf("%s\n",out);
18
19     // 释放内存
20     cJSON_Delete(usr);
21     free(out);
22
23
24 }
```

运行结果：

```
1  {
2      "name": "fengxin",
3      "passwd": "123",
4      "num": 1
5  }
```

若干说明

1. cJSON_CreateObject函数可创建一个根数据项，之后便可向该根数据项中添加string或int等内容,返回的是一个 cJSON的指针，注意，在这个指针使用后，需要手动用 cJSON_Delete(root); 进行内存回收。
2. cJSON_AddNumberToObject向节点中添加子节点，例如此处添加name节点，节点值为字符串"fengxin"



举报

3. 需要注意的是 json 格式的数据，虽然也是一个字符串的样子，但这个时候还是无法当成普通的字符串进行使用，需要调用 cJSON_PrintUnformatted(root) 或者 cJSON_Print(root); 来将json对象转换成普通的字符串，并且都是以该json对象的根为基点。两个API的区别即是：一个是没有格式的；也就是转换后的字符串中间不会有空格存在，而cJSON_Print(root);打印出来是人看起来很舒服的格式。
4. 因为函数内部封装有malloc函数，所以使用free函数释放被out占用的内存空间

<2> 创建一个数组，并向数组添加一个字符串和一个数字：

```
1 int create_js(void)
2 {
3     cJSON *root, *js_body;
4     root = cJSON_CreateArray();
5     cJSON_AddItemToArray(root, cJSON_CreateString("Hello world"));
6     cJSON_AddItemToArray(root, cJSON_CreateNumber(10));
7     {
8         // char *s = cJSON_Print(root);
9         char *s = cJSON_PrintUnformatted(root);
10        if(s){
11            printf(" %s \n",s);
12            free(s);
13        }
14    }
15    if(root)
16        cJSON_Delete(root);
17
18    return 0;
19 }
20
21
22 int main(int argc, char **argv)
23 {
24     create_js();
25     return 0;
26 }
```

运行结果：

```
["Hello world",10]
```

<3> 对象里面包括一个数组，数组里面包括对象，对象里面再添加一个字符串和一个数字：

```
1 int create_js(void)
2 {
3     cJSON *root, *js_body, *js_list;
4     root = cJSON_CreateObject();
5     cJSON_AddItemToObject(root, "body", js_body = cJSON_CreateArray());
6     cJSON_AddItemToArray(js_body, js_list = cJSON_CreateObject());
7     cJSON_AddStringToObject(js_list, "name", "fengxin");
8     cJSON_AddNumberToObject(js_list, "status", 100);
9
10    {
11        // char *s = cJSON_Print(root);
12        char *s = cJSON_PrintUnformatted(root);
13        if(s){
14            printf(" %s \n",s);
15            free(s);
16        }
17    }
18    if(root)
19        cJSON_Delete(root);
20 }
```



24



12



举报

```
21 |     return 0;22 | }
23 |
24 | int main(int argc, char **argv)
25 | {
26 |     create_js();
27 |     return 0;
28 | }
```

运行结果:

```
{ "body": [{ "name": "fengxin", "status": 100 } ] }
```

<4>其他组合就依次类推，只要搞清楚父子关系即可。随便嵌套随便玩。不再贴了。

四.解析json得到数据

解析数据包的过程和组装数据包的过程相反

处理流程:

1, 先将普通的json串处理成json对象，也就是所谓的创建json root的过程，只有一行代码即可:

```
1 | cJSON *root;
2 | root = cJSON_Parse(js_string);
```

2, 开始拿关键字，但如果关键字还有父层或者祖层，那就需要先从父层开拿，所谓剥洋葱是也！

先说没有父层的:

```
out="{ \"name\": \"fengxin\", \"passwd\": \"123\", \"num\": 1 }
```

```
1 | #include<stdio.h>
2 | #include<stdlib.h>
3 | #include<string.h>
4 | #include"cJSON.h"
5 |
6 | int main()
7 | {
8 |     cJSON *json,*json_name,*json_passwd,*json_num;
9 |     char* out="{ \"name\": \"fengxin\", \"passwd\": \"123\", \"num\": 1 }";
10 |
11 |     json = cJSON_Parse(out); //解析成json形式
12 |     json_name = cJSON_GetObjectItem( json , "name" ); //获取键值内容
13 |     json_passwd = cJSON_GetObjectItem( json , "passwd" );
14 |     json_num = cJSON_GetObjectItem( json , "num" );
15 |
16 |     printf("name:%s,passwd:%s,num:%d\n",json_name->valuestring,json_passwd->valuestring,json_num->valueint);
17 |
18 |     cJSON_Delete(json); //释放内存
19 |     free(out);
20 |
21 |
22 |
23 | }
```



显示结果:

```
name:fengxin,passwd:123,num:1
```

需要注意的是：上面的type 已经在cJSON.h里面定义好了，有自己的意义。如果是在严格的场所，应该先判定该 item的type，然后去拿值。而如果有父层的话，无非就是接着向下拿就是了

3.处理这串数据：

```
out={"list":{"name":"xiao hong","age":10},"other":{"name":"hua hua"}}

1 char *s = "{\"list\":{\"name\":\"xiao hong\",\"age\":10},\"other\":{\"name\":\"hua hua\"}}";
2 cJSON *root = cJSON_Parse(s);
3 if(!root) {
4     printf("get root faild !\n");
5     return -1;
6 }
7
8 cJSON *js_list = cJSON_GetObjectItem(root, "list");
9 if(!js_list) {
10     printf("no list!\n");
11     return -1;
12 }
13 printf("list type is %d\n",js_list->type);
14
15 cJSON *name = cJSON_GetObjectItem(js_list, "name");
16 if(!name) {
17     printf("No name !\n");
18     return -1;
19 }
20 printf("name type is %d\n",name->type);
21 printf("name is %s\n",name->valuestring);
22
23 cJSON *age = cJSON_GetObjectItem(js_list, "age");
24 if(!age) {
25     printf("no age!\n");
26     return -1;
27 }
28 printf("age type is %d\n", age->type);
29 printf("age is %d\n",age->valueint);
30
31 cJSON *js_other = cJSON_GetObjectItem(root, "other");
32 if(!js_other) {
33     printf("no list!\n");
34     return -1;
35 }
36 printf("list type is %d\n",js_other->type);
37
38 cJSON *js_name = cJSON_GetObjectItem(js_other, "name");
39 if(!js_name) {
40     printf("No name !\n");
41     return -1;
42 }
43 printf("name type is %d\n",js_name->type);
44 printf("name is %s\n",js_name->valuestring);
45
46 if(root)
47     cJSON_Delete(root);
48     return 0;
```



24



12



举报

打印结果:

```
1 list type is 6
2 name type is 4
3 name is xiao hong
4 age type is 3
5 age is 10
6 list type is 6
7 name type is 4
8 name is hua hua
```

所谓子子孙孙无穷尽也，按照上面那个方法推下去即可。

4.json 里数组怎么取？

```
{\"list\":[\"name1\",\"name2\"]}
```

```
1 int main(int argc, char **argv)
2 {
3     char *s = \"{\\\"list\\\":[\\\"name1\\\",\\\"name2\\\"]}\";
4     cJSON *root = cJSON_Parse(s);
5     if(!root) {
6         printf("get root failed !\\n");
7         return -1;
8     }
9     cJSON *js_list = cJSON_GetObjectItem(root, "list");
10    if(!js_list){
11        printf("no list!\\n");
12        return -1;
13    }
14    int array_size = cJSON_GetArraySize(js_list);
15    printf("array size is %d\\n",array_size);
16    int i = 0;
17    cJSON *item;
18    for(i=0; i< array_size; i++) {
19        item = cJSON_GetArrayItem(js_list, i);
20        printf("item type is %d\\n",item->type);
21        printf("%s\\n",item->valuestring);
22    }
23
24    if(root)
25        cJSON_Delete(root);
26    return 0;
27 }
```

5.如果json数组里面又搞了对象怎么办？

不怕搞对象，就怕这样搞对象？ 无他，就是稍微复杂了一点，全是体力活儿：

- <1> 既然是数组里面，那肯定要先测量数组的大小，然后根据大小循环拿；
- <2> 拿到一个数组项，然后把这个项先转化成普通的json字符串，也就是 char *s 能接受的。
- <3> 再次将这个json字符串，转化为一个json对象
- <4> 按照拿普通对象中的东西那样开干就行了。

```
{\"list\":[{\\\"name\\\":\\\"xiao hong\\\",\\\"age\\\":10},{\\\"name\\\":\\\"hua hua\\\",\\\"age\\\":11}]}
```



list 是一个数组，数组里面有两个对象，那么代码如下

```
1 int main(int argc, char **argv)
2 {
3     char *s = "{\"list\": [{\"name\": \"xiao hong\", \"age\": 10}, {\"name\": \"hua hua\", \"age\": 11}]}";
4     cJSON *root = cJSON_Parse(s);
5     if(!root) {
6         printf("get root failed !\n");
7         return -1;
8     }
9     cJSON *js_list = cJSON_GetObjectItem(root, "list");
10    if(!js_list){
11        printf("no list!\n");
12        return -1;
13    }
14    int array_size = cJSON_GetArraySize(js_list);
15    printf("array size is %d\n", array_size);
16    int i = 0;
17    cJSON *item, *it, *js_name, *js_age;
18    char *p = NULL;
19    for(i=0; i< array_size; i++) {
20        item = cJSON_GetArrayItem(js_list, i);
21        if(!item) {
22            //TODO...
23        }
24        p = cJSON_PrintUnformatted(item);
25        it = cJSON_Parse(p);
26        if(!it)
27            continue ;
28        js_name = cJSON_GetObjectItem(it, "name");
29        printf("name is %s\n", js_name->valuestring);
30        js_age = cJSON_GetObjectItem(it, "age");
31        printf("age is %d\n", js_age->valueint);
32    }
33 }
34
35 if(root)
36     cJSON_Delete(root);
37 return 0;
38 }
```

24

12

文章最后发布于: 2016-11-1

cJSON学习笔记

阅读数 5万+

JSON(JavaScriptObjectNotation)是一种轻量级的数据交换格式。易于人阅读和编写... 博文 来自: 物联网 lo...

想对作者说点什么


yy900313 2周前 #8楼

如何对某个字段值进行修改

鲁啦 4个月前 #7楼


用C函数怎么对本地的Json文件做修改呢

举报



吕白_

11个月前 #6楼



查看 12 条热评

- 全面详解c语言使用cJSON解析JSON字符

为什么选择cJSON来解析JSON字符串？因为简洁又简单，而且效率又快， cJSON工程... 博文 来自： 17岁boy...

阅读数 3万+
- cJSON使用以及在stm32中的应用（转自https://blog.csdn.net/jipeng...

1. cJSON使用最近一段时间在整理自己进入研究生阶段后学习过程中的一些内容，之... 博文 来自： 苴生的博客

阅读数 1242
- 学会了这些技术，你离BAT大厂不远了

每一个程序员都有一个梦想，梦想着能够进入阿里、腾讯、字节跳动、百度等一线互... 博文 来自： 平头哥的...

阅读数 18万+
- 高级会计师评审条件放宽

高级会计师评审难度

9003阅读


- cJSON源码解析

关于cjson的介绍和使用方法就不在这里介绍了，详情请查看上一篇博客cjson使用方... 博文 来自： fengxin的...

阅读数 4680
- 使用cJSON解析JSON字符串

使用cJSON解析JSON字符串一、为何选择cJSON；二、cJSON的核心结构体；三、... 博文 来自： lintax的专...

阅读数 4万+
- 一些实用的GitHub项目

最近整理了一些在GitHub上比较热门的开源项目关于GitHub，快速了解请戳这里其中... 博文 来自： 路漫漫其...

阅读数 3万+
- cJSON 使用介绍

首先，要介绍一下什么是json格式。一.JSON格式简述JSON(JavaScriptObjectNotati... 博文 来自： D&T的博客

阅读数 7069
- cJSON 使用笔记

cJSON： /* cJSON Types: */#define cJSON_Invalid (0)#define cJSON_False (1 &... 博文 来自： SongYuL...

阅读数 752

24




12















一本可陪伴一辈子的笔记本

可擦可写，可循环利用，支持OCR识别，让你的笔记本变得智能

广告

- 全球最厉害的 14 位程序员！

来源|ITWorld整理自网络全球最厉害的14位程序员是谁？今天就让我们一起来了解一... 博文 来自： GitHubD...

阅读数 2万+
- 什么是大公司病（太形象了）

点击蓝色“五分钟学算法”关注我哟加个“星标”，天天中午 12:15，一起学算法作... 博文 来自： 程序员吴...

阅读数 1万+



xukai871105

205篇文章

排名:463

关注



17岁boy想当攻城狮

168篇文章

排名:7000+

关注



eliote_C

16篇文章

排名:千里之外

关注



平头哥的技术博文

65篇文章

排名:5000+

关注

- 分享靠写代码赚钱的一些门路

作者mezod，译者josephchang10如今，通过自己的代码去赚钱变得越来越简单，不... 博文 来自： qq_3357...

阅读数 6284
- MySQL数据库—SQL汇总

一、准备下文整理常见SQL语句的用法，使用MySQL5.7测试，参考了尚硅谷MySQL... 博文 来自： Sirm23333

阅读数 2万+
- nginx学习，看这一篇就够了：下载、安装。使用：正向代理、反向代理、...

文章目录前言一、nginx简介1. 什么是 nginx 和可以做什么事情2.Nginx 作为 web 服... 博文 来自： 冯安晨

阅读数 1万+

程序员的遗憾：忽视数学必定后悔！

如果你没有抓住机会学习数学，迟早有一天会后悔！



举报

程序员真是太太太太有趣了！！

阅读数 2万+

点击上方“程序员小明”，选择“星标”今晚可以不加班！网络上虽然已经有了很多... 博文 来自： 程序员小明

cJSON使用

阅读数 175

cJSON库使用：首先从https://sourceforge.net/projects/cjson/官网下载json库，用... 博文 来自： a136987...

cJson的使用方法

阅读数 122

cJson的使用方法 学长讲了cJson的基础后，做一下整理，并从网上找了一些资料做深... 博文 来自： lalala323...

【安全】Web渗透测试（全流程）

阅读数 7285

1 信息收集1.1域名、IP、端口域名信息查询：信息可用于后续渗透IP信息查询：确认... 博文 来自： qqchaoz...

Ubuntu 19.10 安装 美化 优化 conky Docker

阅读数 3650

Ubuntu终极美化1.附上我的截图： 博文 来自： Hunter80...

AI实训路线推荐：考虑转型/跳槽程序员免费领取，自测专业程度！

如何能够短时间内抓住技术重点，打造属于自己的“offer收割机”？

cJSON_Delete调用时segmentfault

阅读数 2479

cJSON_DeletecJSON_Delete释放cJSON构造object时申请的内存资源通常用法cJSO... 博文 来自： sliver.che...

Spring高级技术梳理

阅读数 5954

Spring高级技术梳理 序言正文SpringDate部分Spring全家桶之SpringData——预科... 博文 来自： 时间静止

五款高效率黑科技神器工具，炸裂好用，省时间

阅读数 2万+

loonggg读完需要4分钟速读仅需2分钟感觉我好久好久没有给大家分享高质量的软件... 博文 来自： 非著名程...

史上最详细的IDEA优雅整合Maven+SSM框架（详细思路+附带源码）

阅读数 5万+

网上很多整合SSM博客文章并不能让初探ssm的同学思路完全的清晰，可以试着关掉... 博文 来自： 程序员宜...

cJSON介绍及使用

阅读数 2425

JSON(JavaScriptObjectNotation)是一种轻量级的文本数据交换格式，易于让人阅读... 博文 来自： 网络资源...



螺旋地桩比普通地桩好在哪

螺旋地桩生产厂家

1.5万阅读

刚出炉！年度榜单：Python三连冠，碾压Java！你怎么看？

阅读数 3270

新一年度的编程语言榜单来了，如果你每年都在看，会发现编程榜单跟娱乐圈的“金... 博文 来自： CSDN资讯

浅谈滴滴派单算法

阅读数 1万+

本文作者：王犇滴滴|首席算法工程师导读：说到滴滴的派单算法，大家可能感觉到既... 博文 来自： DiDi_Tec...

黑客入门，从HTB开始

阅读数 2万+

Hack the box 是国外的一个靶机平台，里面的靶机包含多种系统类型，并且里面可以... 博文 来自： neal1991...

cJSON使用解析

阅读数 321

cJSON_Parse(pdata); //传递数据接口 提供一个JSON块，这将返回一个可以查询的C... 博文 来自： zyf方的博客

cjson

阅读数 480

1.JSON格式简述JSON(JavaScriptObjectNotation)是一种轻量级的数据交换格式。 ... 博文 来自： 咸鱼看到...

[推动全社会公益氛围形成，使公益与空气和阳光一样触手可及。](#)

公益缺你不可，众多公益项目等你PICK——百度公益 让公益像「空气和阳光」一样触手可及！
gongyi.baidu.com


Git 天天用 但是 Git 原理你了解吗？


阅读数 2万+

Git原理做技术一定要知其然知其所以然，意思就是：知道它是这样的，更知道它为什... 博文 来自： ljk126wy...

24



12


















举报

<div><div>cjson使用教程</div><div>cjson使用教程本文简单介绍cJSON后，说明读取json文件、解析json文件、生成json... 博文 来自： 绯浅you...</div></div>	阅读数 1344	
<div><div>JSON格式以及cJSON的使用及注意事项</div><div>前言在STM32 OTA例程中，设备端（STM32F769探索板）与云端交换数据使用的是J... 博文 来自： 我是小超斌</div></div>	阅读数 205	
<div><div>SQL基本语法入门 看这里就够了</div><div>SQL执行顺序第一步：执行FROM第二步：WHERE条件过滤第三步：GROUP BY 分组... 博文 来自： Mr_Fei的...</div></div>	阅读数 1万+	
<div><div>2019年10月中国编程语言排行榜</div><div>2019年10月2日，我统计了某招聘网站，获得有效程序员招聘数据9万条。针对招聘信... 博文 来自： juwikuan...</div></div>	阅读数 1万+	
<div><div>如何学习人工智能，学习AI的一般路线</div><div>下面给大家介绍一些人工智能的学习步骤</div></div>		
<div><div>程序员必须掌握的核心算法有哪些？</div><div>由于我之前一直强调数据结构以及算法学习的重要性，所以就有一些读者经常问我， ... 博文 来自： 帅地</div></div>	阅读数 7万+	
<div><div>动画：用动画给女朋友讲解 TCP 四次分手过程</div><div>作者 小鹿来源 公众号：小鹿动画学编程写在前边大家好，我们又见面了，做为一个业... 博文 来自： 一个不甘...</div></div>	阅读数 3万+	
<div><div>cjson使用笔记（转）</div><div>转载来源：http://www.cnblogs.com/chineseboy/p/3959852.html ... 博文 来自： 奥丁的博客</div></div>	阅读数 1195	
<div><div>从入门到精通，Java学习路线导航（附学习资源）</div><div>引言最近也有很多人来向我"请教"，他们大都是一些刚入门的新手，还不了解这个行... 博文 来自： java_sha...</div></div>	阅读数 4988	
<div><div>让程序员崩溃的瞬间（非程序员勿入）</div><div>今天给大家带来点快乐，程序员才能看懂。来源：https://zhuanlan.zhihu.com/p/47... 博文 来自： stronger...</div></div>	阅读数 25万+	
<div><div> 婴儿听力测试没通过 婴儿的听力怎么检查 3.3万阅读</div></div>		
<div><div>我花了一夜用数据结构给女朋友写个H5走迷宫游戏</div><div>起因又到深夜了，我按照以往在csdn和公众号写着数据结构！这占用了我大量的时间... 博文 来自： bigsai</div></div>	阅读数 18万+	
<div><div>Python 植物大战僵尸代码实现(1):图片加载和显示切换</div><div>功能实现如下： 支持的植物类型：太阳花，豌豆射手，寒冰射手，坚果，樱桃炸弹。... 博文</div></div>	阅读数 2473	
<div><div>程序员实用工具网站</div><div>目录 1、搜索引擎 2、PPT 3、图片操作 4、文件共享 5、应届生招聘 6、程序员面试...</div></div>	阅读数 20万+	
<div><div>我的 Input框 不可能这么可爱</div><div>作者：陈大鱼头 github：KRISACHAN &lt;input /&gt; 标签是我们日常开发中非常...</div></div>	阅读数 1万+	
<div><div>对计算机专业来说学历真的重要吗？</div><div>我本科学校是渣渣二本，研究生学校是985，现在毕业五年，校招笔试、面试，社招面...</div></div>	阅读数 9万+	
<div><div> 高级会计师评审条件放宽 高级会计师评审难度 9003阅读</div></div>		
<div><div>java学习路线导航【教学视频+博客+书籍整理】</div><div>在博主认为，学习java的最佳学习方法莫过于视频+博客+书籍+总结，前三者博主将...</div></div>	阅读数 1万+	
<div><div>TCP三次握手原理</div><div>TCP协议\TCP三次握手</div></div>	阅读数 8394	

 24



 12


















举报

求小姐姐抠图竟遭白眼？痛定思痛，我决定用 Python 自力更生！ 点击蓝色“Python空间”关注我丫加个“星标”，每天一起快乐的学习大家好，我是 ...	阅读数 3653 博文	
python学习目录 这是我学习python的一套流程，从入门到上手 一、Python入门、环境搭建、变量、...	阅读数 1万+ 博文	
刷了几千道算法题，这些我私藏的刷题网站都在这里了！ 遥想当年，机缘巧合入了 ACM 的坑，周边巨擘林立，从此过上了"天天被虐似死狗"的...	阅读数 2万+ 博文	
2019年18家大厂Java面试题整理了350道（分布式+微服务+高并发+性能... 前言 2019年还有不到2个月的时间就结束了，这一你，你收获了多少？前段时间一直...	阅读数 2328 博文	
JavaScript 为什么能活到现在？ 作者 司徒正美 责编 郭芮 出品 CSDN（ID：CSDNnews） JavaScript能发展到现...	阅读数 4407 博文	
别翻了，这篇文章绝对让你深刻理解java类的加载以及ClassLoader源码分... 点进文章的盆友不如先来出一道非常常见的面试题，如果你能做出来，可能你早已掌...	阅读数 8070 博文	
项目中的if else太多了，该怎么重构？ 介绍 最近跟着公司的大佬开发了一款IM系统，类似QQ和微信哈，就是聊天软件。我...	阅读数 1万+ 博文	
致 Python 初学者 欢迎来到“Python进阶”专栏！来到这里的每一位同学，应该大致上学习了很多 Pyth...	阅读数 3万+ 博文	
重磅！云+X 案例征集正式启动啦！ 云计算的概念已经悄然走过十余年，尽管我们对于它的关注，没有像人工智能、物联...	阅读数 8240 博文	
前端开发学习常用网站网址及介绍（都是免费的） 在开发的时候，想记住所有的单词基本是不可能的，所以需要进入文档，只要理清...	阅读数 8230 博文	
YouTube排名第一的励志英文演讲《Dream(梦想)》 Idon’ t know what that dream is that you have, I don’t care how disappointing...	阅读数 1万+ 博文	
《吊打面试官》系列-Redis常见面试题（带答案） 你知道的越多，你不知道的越多 点赞再看，养成习惯 GitHub上已经开源，有面试点...	阅读数 6617 博文	
动画图解：十大经典排序算法动画与解析，看我就够了！（配代码完整版） 排序算法是《数据结构与算法》中最基本的算法之一。 排序算法可以分为内部排序和...	阅读数 2万+ 博文	
“狗屁不通文章生成器” 登顶GitHub热榜，分分钟写出万字形式主义大作 一、垃圾文字生成器介绍 最近在浏览GitHub的时候，发现了这样一个骨骼清奇的雷人...	阅读数 2万+ 博文	
程序员：我终于知道post和get的区别 是一个老生常谈的话题，然而随着不断的学习，对于以前的认识有很多误区，所以还...	阅读数 2万+ 博文	
Python 植物大战僵尸代码实现(2):植物卡片选择和种植 这篇文章要介绍的是： - 上方植物卡片栏的实现。 - 点击植物卡片，鼠标切换为植物...	阅读数 1175 博文	
python json java mysql pycharm android linux json格式 c# 漂亮字 c# 上取整 除法 c#substring c#中延时关闭 c#线段拖拉 c# 第三方报表 c# vs多行注释快捷键 bitmap c# c# 图片转 二进制字符串 c# 程序嵌入桌面		


24



12

















Rotation.
TA的个人主页 >

私信

关注

原创

粉丝

获赞

评论

访问



举报

9024338521342万+

等级: 博客5

积分: 4291

勋章:   

周排名: 4232

总排名: 1万+

最新文章

ES6 Promise用法总结

机器学习-朴素贝叶斯

统计学习方法——K近邻模型

机器学习-感知机

SPA（单页面应用）和MPA（多页面应用）

归档

2019年9月

4篇

2019年8月

10篇

2019年7月

4篇

2019年4月

1篇

2019年1月

4篇

2018年12月

1篇

2018年8月

1篇

2017年12月

2篇

展开

分类专栏



c++ 学习之旅

13篇



Linux网络编程

12篇



c语言

9篇



系统编程

13篇



数据结构与算法

3篇

展开

热门文章

用malloc动态申请一个二维数组的三种方法

阅读数 41935

cJSON的使用方法

阅读数 33107

Python3之urllib库的使用总结

阅读数 32683

Python3爬虫抓取网易云音乐热评实战

阅读数 20019


c++ initializer_list详解

阅读数 18262

最新评论

判断两个单链表是否相交及找到第一个...

qq_14841287: 4.哈希表法。使用hash过于复杂吧,用一个数组保存每个链表元素就可以了;遍历 ...


24




12

















举报

Python3使用SMTP协议发送...

LLLLLLFY: 啊是我没开授权🙄

Python3使用SMTP协议发送...

LLLLLLFY: [reply]LLLLLLFY[/reply] 连垃圾邮件都没收到。。。

Python3使用SMTP协议发送...

LLLLLLFY: run成功了但就是没收到邮件是什么情况啊，哭了🙄

cJSON的使用方法

yy900313: 如何对某个字段值进行修改

腾讯云

学生服务器套餐

10元/月

• 1核2G • 1M宽带

• 50GB存储

立即购买

程序人生

CSDN资讯

QQ客服

kefu@csdn.net

客服论坛

400-660-0108

工作时间 8:30-22:00

关于我们 招聘 广告服务 网站地图

百度提供站内搜索 京ICP备19004658号

©1999-2019 北京创新乐知网络技术有限公司

网络110报警服务 经营性网站备案信息

北京互联网违法和不良信息举报中心

中国互联网举报中心 家长监护 版权申诉

24

12

举报