

目录

Python教程

Python简介

安装Python

第一个Python程序

Python基础

函数

高级特性

面向对象编程

模块

面向对象编程

错误、调试和测试

IO编程

进程和线程

正则表达式

常用内建模块

datetime

collections

base64

struct

hashlib

hmac

itertools

contextlib

urllib

XML

YAML

YAMLParser

常用第三方模块

virtualenv

图形界面

网络编程

电子邮件

访问数据库

Web开发

异步IO

实战

FAQ

期末总结

base64

关于作者

廖雪峰

北京 朝阳区

加关注

廖雪峰

自己的Python课程

Python商业爬虫全解

让天下没有爬不到的数据!

Python爬虫

+

数据分析

Python机器学习

+

深度学习

找廖雪峰老师

廖雪峰老师

自己的Java课程

Java高级架构师

更专业 更权威

源码分析专题

+

微服务架构专题

高并发分布式专题

+

性能优化专题

找廖雪峰老师

python免费公开课

编程学习网

授课模式：在线直播+课后视频，从零基础到中高端开发工程师

查看详情

腾讯云学生服务器体验套餐

10元/月

腾讯云

校园专享服务器套餐10元/月

腾讯云服务器高性能计算能力,更有学生优惠套餐云服务器/域名/存储等服务.

base64

阅读: 110589

Base64是一种用64个字符来表示任意二进制数据的方法。

用记事本打开 .exe、.jpg、.pdf 等文件时，我们都会看到一大堆乱码，因为二进制文件包含很多无法显示和打印的字符，所以，如果要让记事本的文本处理软件能处理二进制数据，就需要一个二进制到字符串的转换方法。Base64是一种常见的二进制编码方法。

Base64的原理很简单，首先，准备一个包含4个字符的数组：

['A', 'B', 'C', ..., 'X', 'Y', 'Z', ..., 'a', 'b', ..., 'z', '0', '1', ..., '9', '+']

然后，对二进制数据进行处理，每3个字节一组，一共是 3*8=24 bit，化为4组，每组正好6个bit：

b1 b2 b3

00010011 01101001 10011001

00010011 01101001 10011001

n1 n2 n3 n4

这样我们得到4个数字作为索引，然后查表，获得相应的4个字符，就是编码后的字符串。

所以，Base64编码会3个字节 的二进制数据编码为4个字节的文本数据，长度增加33%，好处是编码后的文本数据可以在邮件正文、网页等直接显示。

如果要编码的二进制数据不是3的倍数，最后会剩下1个或2个字节怎么办？Base64用 '\n' 字节在末尾补足后，再在编码的末尾加上1个或2个 '=' 号，表示补了多少字节，解码的时候，会自动去掉。

Python内置的 base64 可以直接进行base64的编码：

>>> import base64>>> base64.b64encode(b'binary\00string')b'YmluYXVyMD0string='>>> base64.b64decode(b'YmluYXVyMD0string')b'binary\00string'

由于标准的base64编码后可能出现字符 '+' 和 '/'，在URL中就不能直接作为参数，所以又有一种"url safe"的base64编码，其实就是把字符 '+' 和 '/' 分别变成 '-' 和 '_'：

>>> base64.urlsafe_b64encode(b'YmluYXVyMD0string')b'YmluYXVyMD0string='>>> base64.urlsafe_b64decode(b'YmluYXVyMD0string')b'YmluYXVyMD0string'

还可以自己定义64个字符的排列顺序，这样就可以自定义Base64编码，不过，通常情况下完全没有必要。

Base64是一种通过查表的编码方法，不用于加密，即使使用自定义的编码表也不行。

Base64适用于小段内容的编码，比如数字证书签名、Cookie的内容等。

由于 '+' 字符也可能出现在Base64编码中，但 '+' 用在URL、Cookie里面会造成歧义，所以，很多Base64编码后会吧 '+' 去掉：

移除Base64+'>>> 'abc' -> 'TjZJA='# 移除去掉:'>>> 'abc' -> 'TjZJA'

去掉 '=' 后怎么解码呢？因为Base64是把3个字节变为4个字节，所以，Base64编码的长度永远是4的倍数，因此，需要加上 '=' 把Base64字符串的长度变为4的倍数，就可以正常解码了。

小结

Base64是一种任意二进制到文本字符串的编码方法，常用于在URL、Cookie、网页中传输少量二进制数据。

练习

请写一个能处理去掉 '=' 的base64解码函数：

-*- coding: utf-8 -*->>> import base64>>> def safe_base64_decode(s):>>> pass

测试：>>> assert b'abcd' == safe_base64_decode(b'YWJjZAA=='), safe_base64_decode('YWJjZAA==')>>> assert b'abcd' == safe_base64_decode(b'YWJjZA'), safe_base64_decode('YWJjZA')>>> print('ok')

参考源码

do_base64.py

读后有收获可以请作者喝咖啡，读后有任何疑问加群讨论：

咖啡

二维码

还可以分享给朋友：

分享到微博

上一页 下一页

廖雪峰

官方独家

Python商业爬虫全解

找廖雪峰老师

ACM金牌得主

全球顶尖企业一线数据科学家倾力推荐

人工智能与自然语言/计算机视觉课程培训

Artificial Intelligence For NLP/CV Courses

无offer退全款

廖雪峰推荐

JAVA进阶教程

原价1699元

0元领取

腾讯云

高性能云服务器

首台5折

立即选购

Python内部教材+100G全套学习视频

python大型免费高级进阶公开课

Python全栈实战课程限时免费领取

评论

作业

Regel_R88 created at 2 days ago, Last updated at 2 days ago

def safe_base64_decode(s):>>> if len(s)%4 != 0:>>> l = (len(s)+4)//4>>> s = s.ljust(l, b'>>> b = base64.b64decode(s)>>> return b

全部讨论 回复

提交作业

花了一个晚上终于搞懂了，提交本章作业答案

Vikend created at June 4, 2019 3:33 PM, Last updated at June 4, 2019 3:33 PM

首先了解base64编码的核心过程就是8bit分组到6bit分组，然后根据6bit的二进制转10进制到字符串中查表，最终得到这4个字符的过程。那么总结一下就是：>>> ①编码时对象的位数要是3的倍数（最终编码时对象位数要被8整除）>>> ②解码时对象的位数要是4的倍数（最终解码时对象位数要被8整除）>>> 除数分析题目，题目的要求就是被解码对象带不带 '=' 都要能够被正常解码，那么重点就在被解码对象的位数不符合要求了。这样就好写代码了。>>> import base64>>> def safe_base64_decode(s):>>> l = len(s) % 4>>> if l != 0:>>> for i in (0, l):>>> s += b'>>> return base64.b64decode(s)>>> else:>>> return base64.b64decode(s)>>> assert (safe_base64_decode(b'TjZJA==') == b'abcd'), safe_base64_decode('TjZJA==')>>> assert (safe_base64_decode(b'TjZJA') == b'abcd'), safe_base64_decode('TjZJA')>>>>>> ok

Read More

全部讨论 回复

交作业

方大张哥 created at May 23, 2019 6:45 PM, Last updated at May 23, 2019 6:45 PM

def safe_base64_decode(s):>>> while len(s) % 4 != 0:>>> s += b'>>> return base64.b64decode(s)

全部讨论 回复

测试

Geek created at May 17, 2019 10:06 PM, Last updated at May 17, 2019 10:06 PM

def safe_base64_decode(s):>>> l = (len(s)+4)//4>>> s = s.ljust(l, b'>>> n = len(s) % 4>>> return base64.b64decode(s+l*n)

全部讨论 回复

提交作业

张朝晖 created at May 5, 2019 2:45 PM, Last updated at May 5, 2019 2:45 PM

def safe_base64_decode(s):>>> a=len(s)%4>>> s+=a*b'>>> return base64.b64decode(s)

全部讨论 回复

base64练习

张朝晖 created at April 25, 2019 9:47 PM, Last updated at April 25, 2019 9:47 PM

def safe_base64_decode(s):>>> sum = len(s)%4>>> s += b'>>> return base64.b64decode(s)

全部讨论 回复

非递归的数

北岸心海 created at April 22, 2019 3:30 PM, Last updated at April 22, 2019 3:30 PM

def safe_base64_decode(s):>>> s+=b'="(4-(len(s) % 4)) if (len(s) % 4)>0 else 0)>>> return base64.b64decode(s)

全部讨论 回复

作业

Geek created at March 8, 2019 11:50 PM, Last updated at April 21, 2019 11:33 PM

#作业 import base64>>> def safe_base64_decode(s):>>> if len(s)%4 != 0: s += b'>>> return safe_base64_decode(s)>>> else: return base64.b64decode(s)>>> assert b'abcd' == safe_base64_decode(b'YWJjZAA=='), safe_base64_decode('YWJjZAA==') assert b'abcd' == safe_base64_decode(b'YWJjZA'), safe_base64_decode('YWJjZA') print('ok')>>> ok

全部讨论 回复

测试

Geek created at April 21, 2019 11:33 PM

def safe_base64_decode(s):>>> while len(s) % 4: if isinstance(s, bytes): s += b'>>> else: s += b'>>> return base64.b64decode(s)

全部讨论 回复

没必要递归吧

天香 created at April 19, 2019 4:36 PM, Last updated at April 19, 2019 4:36 PM

import base64>>> def safe_base64_decode(s):>>> yushu = len(s)%4>>> if yushu == 0: return base64.b64decode(s)>>> else: s = s + b'>>> (4-yushu)>>> return base64.b64decode(s)>>> UTF-8应用于数据存储较多，而base64用于数据交换和二进制数据传输？>>>

全部讨论 回复

交作业

马海龙 created at April 16, 2019 4:21 PM, Last updated at April 16, 2019 4:21 PM

<pre><code>class"json">:>>> def safe_base64_decode(s)>>> while len(s) % 4 != 0:>>> if(isinstance(s, bytes)):>>> s += b'>>> else: s = s + b'>>> (4-len(s)%4)>>> return base64.b64decode(s)>>> </code></pre>>>

全部讨论 回复

交作业

中微 created at April 4, 2019 9:56 PM, Last updated at April 4, 2019 9:56 PM

def safe_base64_decode(s):>>> while len(s) % 4 > 0:>>> s += b'>>> return base64.b64decode(s)

全部讨论 回复

看来很多人和我思路一样

小强 created at March 30, 2019 6:22 PM, Last updated at March 30, 2019 6:22 PM

def safe_base64_decode(s):>>> s_1=s+(len(s)%4)*b'>>> return base64.urlsafe_b64decode(s_1)

全部讨论 回复

逗了，忘记写return了，还一直在找，今天感觉怪怪的

Zafon created at March 25, 2019 8:11 PM, Last updated at March 25, 2019 9:42 AM

def safe_base64_decode(s):>>> if isinstance(s, bytes):>>> if s[-2:] == b'>>> :>>> return base64.b64decode(s)>>> elif s[-1] == b'>>> :>>> return base64.b64decode(s)>>> else:>>> s = s + b'>>> :>>> return base64.b64decode(s)

全部讨论 回复

Zafon

Created at March 26, 2019 9:42 AM

def safe_base64_decode(s):>>> try:>>> l = len(s)%4>>> if l!=0:>>> return base64.b64decode(s)>>> else:>>> s+=b'>>> :>>> return base64.b64decode(s)>>> finally:>>> pass>>>

Zi_sunrise的答案

def safe_base64_decode(s):>>> while len(s) % 4 > 0:>>> s += b'>>> :>>> return base64.b64decode(s)>>>

用户1234567890总忘不能别人用的答案

def safe_base64_decode(s):>>> s = s + bytes(' '*join('' for i in range(4-len(s)%4))>>> , encoding='utf-8')>>> return base64.urlsafe_b64decode(s)>>>

第二个答案够简洁的，我喜欢

全部讨论 回复

作业

1234567890总忘不能别人 used created at March 22, 2019 2:55 PM, Last updated at March 22, 2019 3:55 PM

def safe_base64_decode(s):>>> snew = s + bytes(' '*join('' for i in range(4-len(s)%4))>>> , encoding='utf-8')>>> return base64.urlsafe_b64decode(snew)

全部讨论 回复

作业

张朝晖 created at March 20, 2019 3:13 PM, Last updated at March 20, 2019 3:13 PM

def safe_base64_decode(s):>>> return base64.b64decode(s)>>> if len(s)%4==0 else (s + b'>>> *((len(s)%4)))>>>

全部讨论 回复

练习答案

Zi_sunrise created at March 14, 2019 5:01 PM, Last updated at March 14, 2019 5:01 PM

def safe_base64_decode(s):>>> while len(s) % 4 > 0:>>> s += b'>>> :>>> return base64.b64decode(s)>>>

全部讨论 回复

作业 19.03.11

陈子雷 created at March 11, 2019 11:23 PM, Last updated at March 11, 2019 11:23 PM

感谢 Cn阿金的递归思路，很有启发>>> def safe_base64_decode(s):>>> return base64.b64decode(s)>>> if len(s) % 4 == 0: else safe_base64_decode(s + b'>>> ')

全部讨论 回复

楼上的递归思想给我启发

张朝晖 created at March 5, 2019 11:24 PM, Last updated at March 5, 2019 11:24 PM

import base64>>> def safe_base64_decode(s):>>> print(isinstance(s,bytes))>>> 如果是非用if type(s)=='str' 语句，因为type语句会转换数据类型>>> (if isinstance(s,bytes):>>> sstr=s+'>>> ')>>> print(s)>>> if len(s) % 4 == 0:>>> return base64.b64decode(s)>>> else:>>> s += b'>>> :>>> return safe_base64_decode(s'>>> ')>>> print(safe_base64_decode(b'YWJjZAA=='))>>> print(safe_base64_decode(b'YWJjZA'))>>>

全部讨论 回复

测试

张朝晖 created at March 2, 2019 7:13 PM, Last updated at March 2, 2019 7:13 PM

def safe_base64_decode(s):>>> rem = len(s) % 4>>> if rem != 0: for i in range(4-rem):>>> s += b'>>> :>>> return base64.b64decode(s)

全部讨论 回复

交作业

Always created at March 2, 2019 3:21 PM, Last updated at March 2, 2019 3:21 PM

def safe_base64_decode(s):>>> L=len(s)>>> Y=L%4>>> for i in range(Y):>>> s+=b'>>> :>>> return base64.b64decode(s)

全部讨论 回复

发表评论

登录后发表评论