

实例 1: 随机生成验证码

很多网站的注册登录业务都加入了验证码技术,以区分用户是人还是计算机,有效地防止刷票、论坛灌水、恶意注册等行为。目前验证码的种类层出不穷,其生成方式也越来越复杂,常见的验证码是由大写字母、小写字母、数字组成的六位验证码。

本实例要求编写程序,实现随机生成六位验证码的功能。

实例目标

- 掌握导入模块的方式
- 掌握 random 模块的基本用法

实例分析

本实例的六位验证码由 6 个字符组成,每个字符都是随机字符,要实现随机字符的功能需要用到随机数模块 random。使用 random 模块生成六位验证码的基本实现思路为:

- (1) 导入 random 模块
- (2) 创建一个空字符串 code_list。
- (3) 生成6个随机字符逐个拼接到code list 后面。

以上实现思路中的步骤(3)是验证码功能的核心部分,此部分主要实现生成6个随机字符的功能。为确保每次生成的字符类型只能为大写字母、小写字母、数字的任一种,这里可使用1、2、3分别代表这三种类型:

- 若产生随机数 1, 代表生成大写字母;
- 若产生随机数 2, 代表生成小写字母;
- 若产生随机数 3, 代表生成数字。

此外,为确保每次生成的是所选类型中的字符,这里需要按三种类型给随机数指定范围,即数字类型对应的数值范围为 0~9,大写字母对应的 ACSII 码范围为 65~90,小写字母对应的 ACSII 码范围为 97~122,如此便可以使用 randint()函数生成一个随机类型中的随机字符。

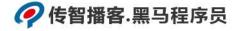
代码实现

完整的代码如下所示:

```
import random

def verifycode():
    code_list = ''
    # 每一位验证码都有三种可能(大写字母, 小写字母, 数字)
```

网址: yx.boxuegu.com 教学交流QQ/微信号: 2011168841



```
for i in range(6): # 控制验证码生成的位数
      state = random.randint(1, 3)
      if state == 1:
         first_kind = random.randint(65, 90) # 大写字母
         random_uppercase = chr(first_kind)
         code_list = code_list + random_uppercase
      elif state == 2:
         second kinds = random.randint(97, 122) # 小写字母
         random lowercase = chr(second kinds)
         code_list = code_list + random_lowercase
      elif state == 3:
         third kinds = random.randint(0, 9)
         code_list = code_list + str(third_kinds)
   return code_list
if __name__ == '__main__':
   verifycode = verifycode()
   print(verifycode)
```

以上代码定义了一个生成六位验证码的函数 verifycode(),该函数中首先定义了一个空字符串 code_list,然后使用 for 语句控制循环执行的次数,即字符的个数,每次循环的基本过程:根据生成的随机数 1、2 或 3 执行不同的分支;生成指定范围内的随机数;将该随机数转换为字符后拼接到 code_list 中,最后返回 code_list。

代码测试

运行程序,结果如下所示:

```
vt3199
```

再次运行程序,结果如下所示:

DfJHeY

网址: yx.boxuegu.com 教学交流QQ/微信号: 2011168841