


 [c0a21162 / ProjExD](#) Public[Code](#) [Issues 2](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#) [main](#) ▾

...

[ProjExD](#) / [ex01](#) / [alphabet.py](#) / <> Jump to ▾

c0a21162 修正 #1

 1 contributor

53 lines (46 sloc) | 1.55 KB

...

```
1  import random
2  import datetime
3
4  num_of_alphabet = 26 #全アルファベット数
5  num_of_all_chars = 10 #対象文字数
6  num_of_abs_chars = 2 #欠損文字数
7  num_of_trials = 2
8
9  def shutudai(alphabet):
10     all_chars = random.sample(alphabet,num_of_all_chars)
11     print("対象文字 : ")
12     for c in all_chars:
13         print(c , end = "")
14     print()
15
16     abs_chars = random.sample(alphabet,num_of_abs_chars)
17     #print("欠損文字(デバック用) : ")
18     for c in abs_chars:
19         print(c,end="")
20     print()
21     return abs_chars
22
23
24 def kaitou(abs_chars):
25     num = int(input("欠損文字はいくつあるでしょうか?"))
26     if num != num_of_abs_chars:
27         print("不正解です")
28     else:
29         print("正解です。それでは、具体的に欠損文字を一つずつ入力してください : ")
30         for i in range(num):
31             ans = input(f"{i + 1} 目の文字を入力してください : ")
32             if ans not in abs_chars:
```

```
33         print("不正解です。")
34         return false
35     else:
36         abs_chars.remove(ans)
37     print("全部正解です。")
38     return True
39
40
41 if __name__ == "__main__":
42     st = datetime.datetime.now()
43     alphabet = [chr(i + 65) for i in range(num_of_alphabet)]
44     print(alphabet)
45     for i in range(num_of_trials):
46         abs_chars = shutudai(alphabet)
47         ret = kaitou(abs_chars)
48         if ret :
49             break
50     else:
51         print("-"*20)
52     ed = datetime.datetime.now()
53     print(f"所要時間 : {(ed-st).seconds}秒")
```