

邮箱轰炸-2





1. 探索新知

项目流程

改编成窗口版发送邮件程序

1. 搭建界面
2. 绑定发送事件
3. 配置邮件内容

1. 封装函数

为了不影响制作tkinter窗口，需要把发送邮件的代码封装成函数

```
user='zhangsan444444z@163.com'  
password='FMORERKAQJWEBSUX'  
def send_mail():  
    to="zhangsan444444z@163.com"  
    subject = "新年祝福"  
    content = "新年快乐祝您新年大吉!"  
    .....  
    .....
```

2. 创建窗口

```
def send_mail():  
    .....  
    .....  
  
w = tkinter.Tk()  
w.geometry('800x600')  
w.resizable(0,0)  
w.mainloop()
```

窗口大小不可变



1. 探索新知

resize + able

```
w.resizable(0,0)
```

写1表示可变

宽不可变

高不可变

3. 设置背景

背景图片制作成标签后，放到窗口上



```
w = tkinter.Tk()
w.geometry('800x600')
w.resizable(0,0)
```

加载背景图片

```
bg = tkinter.PhotoImage(file='email.png')
bg_label = tkinter.Label(w, image=bg)
bg_label.place(x=0, y=0)
```

创建一个标签来显示图片

```
w.mainloop()
```

将标签放置窗口(0,0)坐标



1. 探索新知

4. 添加组件

依次添加所有组件



5. 收件人

Entry() 创建输入框，用于输入收件人的邮件地址

收件人

```
e1 = tkinter.Entry(w, font=("黑体", 20), width=18)
e1.place(x=150, y=190)
```

```
w.mainloop()
```

宽度为18个字符大小

6. 邮件内容

Text() 多行文本框，用于输入邮件具体内容

宽度为18个字符大小，高度为8行文字

```
t2 = tkinter.Text(w, font=("黑体", 20), width=18, height=8)
t2.place(x=150, y=255)
```

```
w.mainloop()
```



1. 探索新知

7. 剩余组件

独立完成轰炸次数、轰炸间隔输入框

```
e3 =tkinter.Entry(w,font=("黑体",20),width=8)  
e3.place(x=508, y=190)
```

```
e4 =tkinter.Entry(w,font=("黑体",20),width=8)  
e4.place(x=508, y=322)
```

```
w.mainloop()
```



8. 发送按钮

Button() 定义发送按钮

按钮文字

```
b5 = tkinter.Button(w, text="发送", font=("黑体",20),width=8)  
b5.place(x=508,y=425)
```

```
w.mainloop()
```

怎么实现点击按钮去发送邮件呢?

给按钮绑定 send_mail() 函数





1. 探索新知

9. 多线程发邮件

但直接绑定 `send_mail()` 会导致发送过程中界面卡死，所以我们借助多线程去调用 `send_mail()`



举例：

```
def func():
```

.....

```
t = threading.Thread(target=func)
```

```
t.start()
```

用新的线程调用 `func()` 函数

```
import threading
```

```
def send_mail():
```

.....

```
def send():
```

```
    t = threading.Thread(target=send_mail)
```

```
    t.start()
```

.....

```
b5 = tkinter.Button(w, ..., command=send)
```

```
b5.place(x=508, y=425)
```

```
w.mainloop()
```

创建新线程 `t`，目标任务为 `send_mail()`

启动线程 `t`

点击按钮绑定 `send`



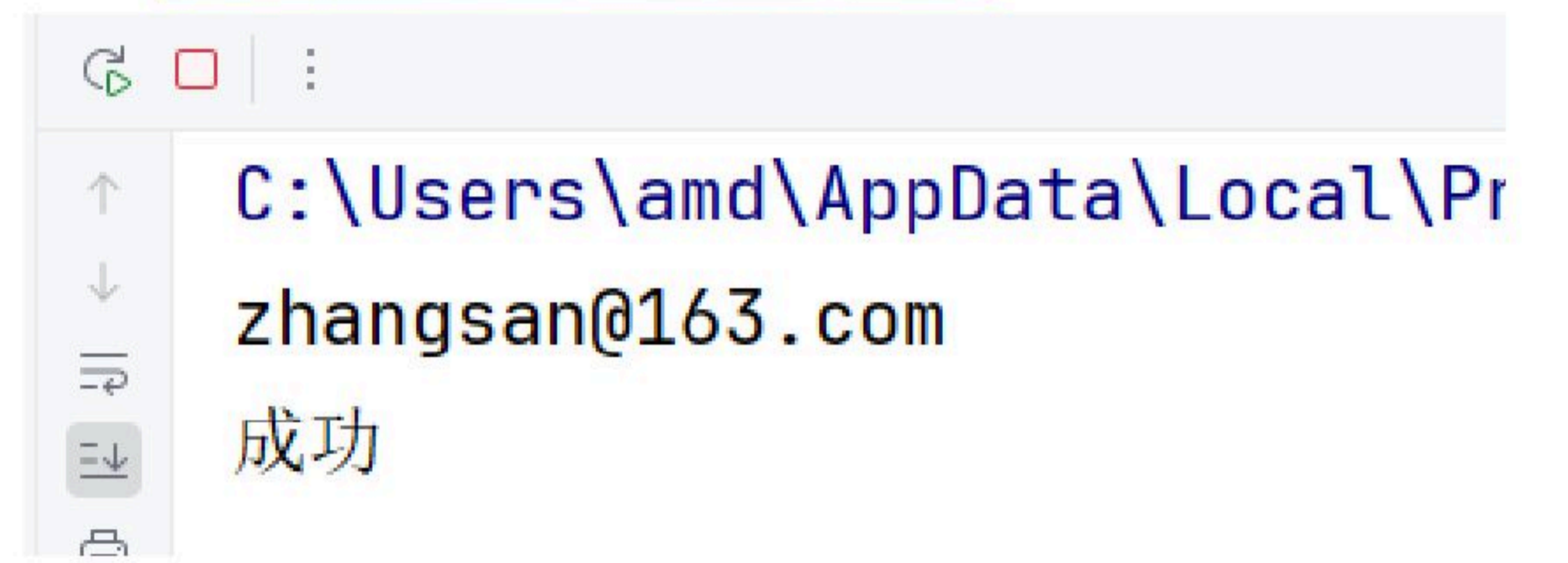
1. 探索新知

10. 获取输入信息

get() 获取输入框的输入信息

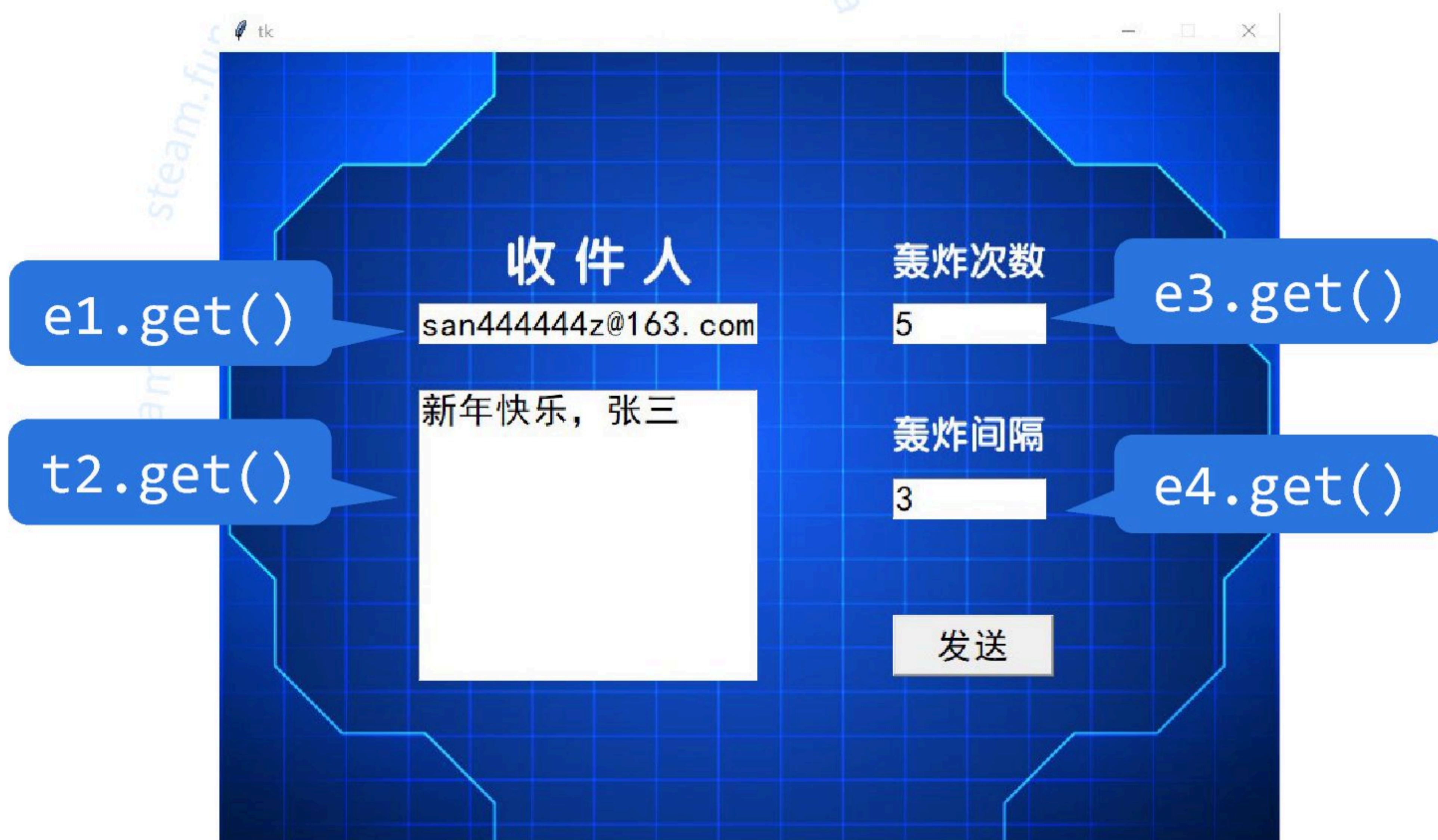
```
def send_mail():
    to = e1.get()
    print(to)
    #to="zhangsan444444z@163.com"
```

删除或注释



```
def send_mail():
    to = e1.get()
    content = t2.get("1.0", "end")
    count = int(e3.get())
    delay = int(e4.get())
```

获取文本框所有文字





1. 探索新知

11. 配置邮件内容

邮件的信息不再从程序中设置, 而通过输入框得到。

```
subject = "新年祝福"  
content = "新年快乐, 祝您新年大吉!"
```

邮件主题 To: 收件人邮箱

```
subject = "To: {}".format(to)  
#content = "新年快乐, 祝您新年大吉!"
```

邮件内容通过输入获取



12. 设置频率

轰炸次数

```
for i in range(count):  
    s.sendmail(msg_from, to, msg.as_string())  
    print('成功')  
    time.sleep(delay)
```

轰炸间隔



1. 探索新知

完整程序

```
import smtplib
from email.mime.text import MIMEText
from email.utils import formataddr
import time
import tkinter
import threading
user='zhangsan444444z@163.com'
password= 'FMORERKAQJWEBSUX'
def send_mail():
    to = e1.get()
    content = t2.get("1.0", "end")
    count = int(e3.get())
    delay= int(e4.get())
    subject = "To:{}".format(to)

    msg = MIMEText(content, 'html', 'utf-8')
    msg['Subject'] = subject
    msg['From'] = formataddr(('张三', user))
    msg['to'] = to
    s = smtplib.SMTP_SSL("smtp.163.com", 465)
    s.login(user, password)
    for i in range(count):
        s.sendmail(user, to, msg.as_string())
        print('成功')
        time.sleep(delay)
```



1. 探索新知

```
def send():
    t = threading.Thread(target=send_mail)
    t.start()
w = tkinter.Tk()
w.geometry('800x600')
w.resizable(0,0)
bg = tkinter.PhotoImage(file='email.png')
bg_label = tkinter.Label(w, image=bg)
bg_label.place(x=0, y=0)
e1 = tkinter.Entry(w, font=("黑体", 20), width=18)
e1.place(x=150, y=190)
t2 = tkinter.Text(w, font=("黑体", 20), width=18, height=8)
t2.place(x=150, y=255)
e3 = tkinter.Entry(w, font=("黑体", 20), width=8)
e3.place(x=508, y=190)
e4 = tkinter.Entry(w, font=("黑体", 20), width=8)
e4.place(x=508, y=322)
b5 = tkinter.Button(w, text="发送", font=("黑体", 20), width=8, command=send)
b5.place(x=508, y=425)
w.mainloop()
```



2. 强化练习

1. 获取Text() 多行文本框的全部文字代码是 ()

- A `get(0, "end")`
- B `get("1.0", "end")`
- C `get("start", "end")`
- D `get("begin", "end")`

2. 以下说法正确的是 ()

```
t = threading.Thread(target=send_mail)
t.start()
```

- A 该代码段定义了 `send_mail` 的函数，并在主线程中立即执行该函数。
- B 该代码段创建了一个线程对象 `t`，但并未启动线程。
- C 该新线程会并发执行 `send_mail` 函数，而主线程不会等待它完成。
- D 会在主线程中同步运行 `send_mail` 函数，并在函数结束后创建一个新线程。



2. 强化练习

3. 以下说法正确的是 ()

```
for i in range(count):  
    s.sendmail(user, to, msg.as_string())  
    print('成功')  
    time.sleep(delay)
```

- A 循环发送邮件 count 次，并在每次发送后等待delay秒开始下一次发送。
- B sendmail() 参数分别为发件人地址，收件人地址，邮件内容。
- C 这行代码会使循环在每次发送后的等待 delay 秒，然后继续下一次循环。
- D 以上选项都正确。

4. 属于发件人一方的信息有(多选)()

```
to = "zhangsan@example.com"  
subject = "To:{}".format(to)
```

- A format() 用于替换字符串中的 {} 占位符，并将 to 的值插入到 subject 字符串中。
- B format() 函数会将字符串 to 自动转为大写字母，然后替换占位符 {}。
- C format() 不需要提供任何参数。
- D format() 函数使用时只能接受数字类型的参数，不能处理字符串类型的参数。



2. 强化练习

5. 窗口宽度不可变、高度可变的正确代码是()

A `w.resizable(0,0)`

B `w.resizable(1,1)`

C `w.resizable(1,0)`

D `w.resizable(0,1)`

3. 术语箱

text	文本
command	命令
send	发送
resize	改变大小
able	能够

4. 课后挑战

【爱的轰炸】

给妈妈或者爸爸的邮箱发送一封邮件。

内容自定义，比如夸一夸妈妈！

