

# python 编写 windows 服务

作者: daodaoliang

版本: 0.0.1

邮箱: daodaoliang@yeah.net

- 1 开发环境的准备.
- 2. 项目工程文件布局.
- 3. 服务文件源码.
- 4. 服务的安装和启动.
  - 4.1 安装服务.
  - 4.2 启动服务.
  - 4.3 重启服务.
  - 4.4 关闭服务.
  - 4.5 卸载服务.
- 5. py2exe 打包程序.
- 6. 用innosetup 进行安装包的打包.

## 1 开发环境的准备

---

- 首先要正确的安装 `python` 请[点击我](#)完成所需进行安装 ;
- 然后安装 `pywin32` 请[点击我](#)完成所需安装 ;
- 最后安装 `py2exe` 请[点击我](#) 完成所需安装 ;

完成上述操作后基本上就完成了服务环境的搭建了 , 接下来就是编写服务程序并打包发布了 ;

## 2. 项目工程文件布局

---



### 3. 服务文件源码

```
import sys

import win32service

import win32event

import win32serviceutil

import servicemanager

import winerror


class MyServer(win32serviceutil.ServiceFramework):

    _svc_name_ = "sync_vip"

    _svc_display_name_ = "sync_vip"

    _svc_description_ = "daodaoliang@yeah.net"


    def __init__(self, args):

        win32serviceutil.ServiceFramework.__init__(self, args)

        self.hWaitStop = win32event.CreateEvent(None, 0, 0, None)

        self.isAlive = True


    def SvcStop(self):

        # 先告诉SCM停止这个过程

        self.ReportServiceStatus(win32service.SERVICE_STOP_PENDING)
```

```

# 设置事件

win32event.SetEvent(self.hWaitStop)


def SvcDoRun(self):

    import time

    import win32evtlogutil

    win32evtlogutil.ReportEvent(self._svc_name_, servicemanager.PYS_SERVICE_S
TARTED, 0,

                                servicemanager.EVENTLOG_INFORMATION_TYPE, (se
lf._svc_name_, ''))

    while self.isAlive:

        print('你的业务逻辑')

        win32evtlogutil.ReportEvent(self._svc_name_, servicemanager.PYS_SERVICE_S
TOPPED, 0,

                                servicemanager.EVENTLOG_INFORMATION_TYPE, (se
lf._svc_name_, ''))

        self.ReportServiceStatus(win32service.SERVICE_STOPPED)

    return


if __name__ == '__main__':

    import os

    if len(sys.argv) == 1:

        try:

            evtsrc_dll = os.path.abspath(servicemanager.__file__)

            servicemanager.PrepareToHostSingle(MyServer)

            servicemanager.Initialize('MyServer', evtsrc_dll)

            servicemanager.StartServiceCtrlDispatcher()

        except win32service.error, details:

            if details[0] == winerror.ERROR_FAILED_SERVICE_CONTROLLER_CONNECT:

                win32serviceutil.usage()

    else:

        win32serviceutil.HandleCommandLine(MyServer)

```

## 4. 服务的安装和启动

---

在完成上述的编写后，其实服务程序就已经可以运行了，运行方式如下：

### 4.1 安装服务

```
python my_server.py install
```

### 4.2 启动服务

```
python my_server.py start
```

### 4.3 重启服务

```
python my_server.py restart
```

### 4.4 关闭服务

```
python my_server.py stop
```

### 4.5 卸载服务

```
python my_server.py remove
```

**防坑提示：**一定要以管理员权限运行该命令，或者在管理员账号下运行，不然会运行失败！

## 5. py2exe 打包程序

---

setup.py源码

```
from distutils.core import setup  
  
import py2exe
```

```
setup(console=["sync_vip.py"],  
      version="0.0.2",  
      name="my_server",  
      description="daodaoliang@yeah.net", requires=['PySQLPool']  
    )
```

然后用终端进入setup.py 所在的目录执行

```
python my_server.py py2exe
```

执行完毕后会生成 `build` 和 `dist` 文件夹，其中 `dist` 里面的内容就是待发布的文件；

## 6. 用innosetup 进行安装包的打包

---

自行百度去吧，这个很简单；