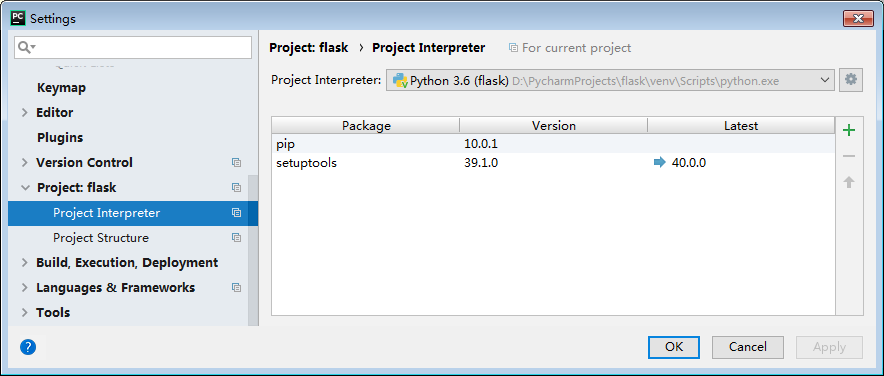
# Flask创建网站

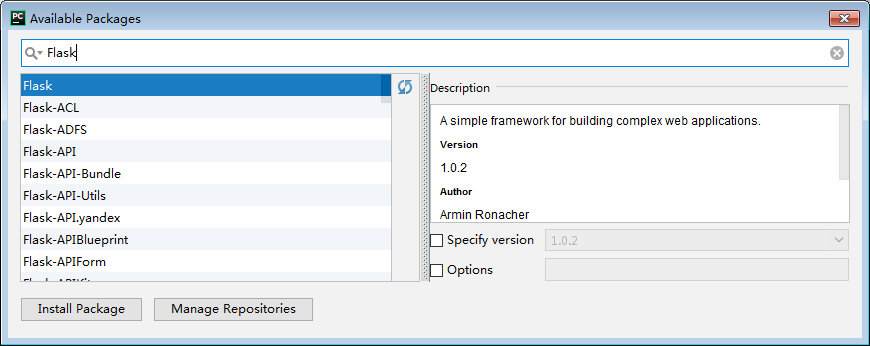
Flask是一个使用 [Python](https://baike.baidu.com/item/Python) 编写的轻量级 Web 应用框架。其 [WSGI](https://baike.baidu.com/item/WSGI) 工具箱采用 Werkzeug ，[模板引擎](https://baike.baidu.com/item/%E6%A8%A1%E6%9D%BF%E5%BC%95%E6%93%8E/907667" \t "_blank)则使用 Jinja2 。Flask使用 BSD 授权。

## 在pycharm中安装Flask库

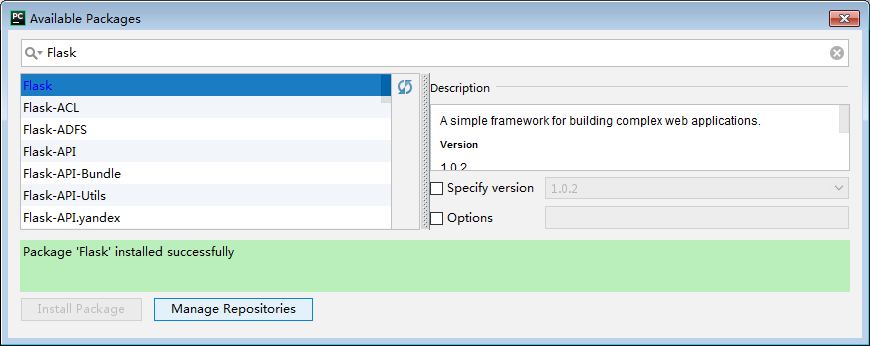
菜单File->Settings。选择Project Interpreter。



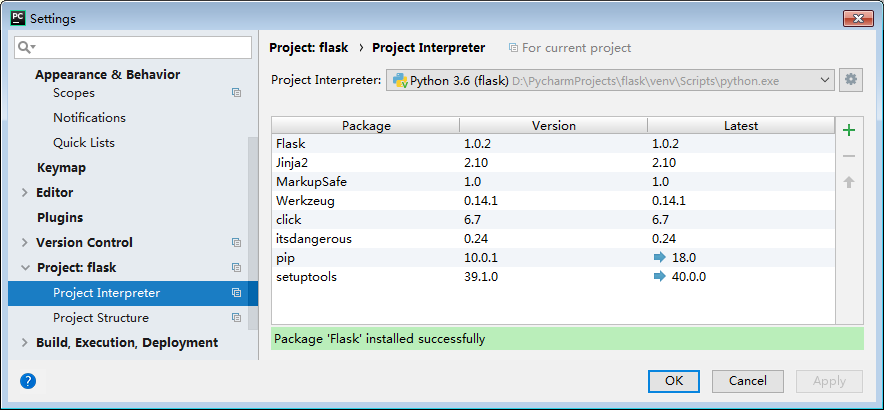
点击右边的“+”号，再弹出的对话框中搜索Flask。



然后单击“Install Package”按钮，开始安装，等待一段时间即可安装成。



关闭Available Packages窗口，回到上一个窗口。



可以看到，Flask安装成功后，还附带安装了Werkzeug工具箱和Jinja2模板引擎等库。

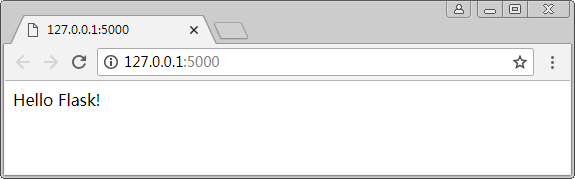
## 创建网站

### 新建入口程序main.py

**from** flask **import** Flask  
  
app = Flask(\_\_name\_\_)  
  
@app.route(**'/'**)  
**def** index():  
 **return "Hello Flask!"  
  
if** \_\_name\_\_ == **'\_\_main\_\_'**:  
 app.run()

然后执行菜单Run-》Run…，然后再弹出的小对话框里选择main项目，即可启动网站。

接着在浏览器里输入127.0.0.1:5000即可看到运行效果。



### 返回HTML代码

@app.route('/')  
def index():  
 html = """  
 <html>  
 <head>  
 <title> Home Page - wave12 blog</title >  
  
 </head >  
 <body>  
 <h1>Hello, welcome to wave12 blog! </h1>  
 <h2>This blog is created by Flask!<h2>  
 </body>  
 </html>  
 """  
 return html



### 返回HTML模板文件

**模板文件夹**

默认情况下,Flask 在程序文件夹中的 templates 子文件夹中寻找模板。

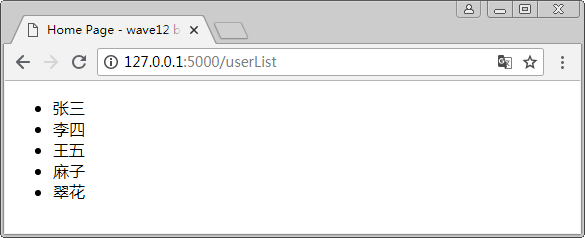
新建templates文件夹，并在其下新建index.html文件。

@app.route(**'/'**)  
**def** index():  
 **return** render\_template(**"index.html"**)

### 多个url路由

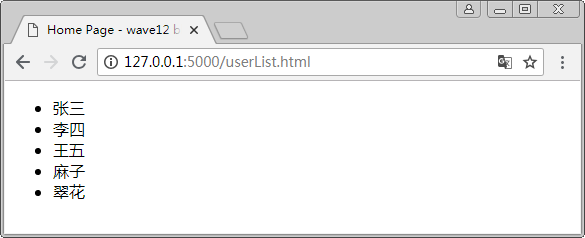
路由，通俗的将就是用户访问网站的网址。再增加一个userList的路由

@app.route(**'/userList'**)  
**def** userList():  
 **return** render\_template(**"userList.html"**)



### 路由伪静态

@app.route(**'/userList.html'**)  
**def** userListHtml():  
 **return** render\_template(**"userList.html"**)



看上去是访问静态页面，实际上是经过后台的路由方法来控制。

### 多个路由指向同一方法

@app.route(**'/userList'**)  
@app.route(**'/userList.html'**)  
**def** userList():  
 **return** render\_template(**"userList.html"**)

### 向模板传递信息

@app.route(**'/product'**)  
**def** product():  
 id = **"100023"  
 return** render\_template(**"product.html"**, idValue=id)

### 动态路由

就是动态传递参数。

@app.route(**'/product/<string:id>'**)  
**def** productId(id):  
 **return** render\_template(**"product.html"**, idValue=id)

也快而已写成

@app.route(**'/product/<string:id>.html'**)

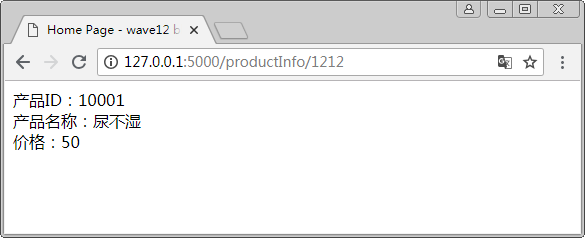
### 向模板中传递多个参数

就是动态传递参数。

@app.route(**'/productDetail/<string:id>'**)  
**def** productDetail(id):  
 pName = **"产品"** + id  
 **return** render\_template(**"productDetail.html"**, idValue=id, productName=pName)

### 向模板中传递json数据

@app.route(**'/productInfo/<string:id>'**)  
**def** productInfo(id):  
 *# 根据id查询数据库，得到产品的具体信息* product = {  
 **"id"**: **"10001"**,  
 **"productName"**:**"尿不湿"**,  
 **"price"**: 50  
 }  
 **return** render\_template(**"productJson.html"**, product=product)



### 模板中展现图片

默认设置下，Flask在程序根目录中名为static的子目录中寻找静态文件。

新建static文件夹，在其中再新建images子文件夹，放入Flask.jpg图片。然后再模板文件里调用。

<**body**>  
 <**h1**>Hello, welcome to wave12 blog! </**h1**>  
 <**h2**>This blog is created by Flask!<**h2**>  
 <**img src="static/images/Flask.jpg"** />  
</**body**>