

传参:

```
@app.route('/a/<id>')
```

```
def get_id(id):
```

```
    return u'获取到的id是:%s' % id
```

使用该方式可以获取到浏览器中链接的参数, 获取到参数后可以根据参数进行一系列操作  
要传递的参数用<>表示

如果要限定参数的数据类型:

1. 如果没有指定具体的数据类型, 那么默认就是使用string数据类型。string不能接收参数中含有 /
2. int数据类型只能传递int类型。
3. float数据类型只能传递float类型。
4. path数据类型和string有点类似, 都是可以接收任意的字符串, 但是path可以接收路径, 也就是说可以包含斜杠 / ,斜杠分开的部分视作同一整体传入
5. uuid数据类型只能接收符合uuid的字符串。uuid是一个全宇宙都唯一的字符串, 一般可以用来作为表的主键。
6. any数据类型可以在一个url中指定多个路径。当两个不同的url需要映射到同一视图函数时采用,

例如:

```
@app.route('/<any(blog,article):url_path>/<id>/')
```

```
def detail(url_path,id):
```

```
    if url_path == 'blog':
```

```
        return '博客详情: %s' % id
```

```
    else:
```

```
        return '用户详情: %s' % id
```

接收用户传递的参数:

1. 第一种: 使用path的形式 (将参数嵌入到路径中) , 就是上面讲的。
2. 第二种: 使用查询字符串的方式, 就是通过?key=value的形式传递的。

```
@app.route('/d/')
```

```
def d():
```

```
    wd = request.args.get('wd')
```

```
    return '您通过查询字符串的方式传递的参数是: %s' % wd
```

3. 如果你的这个页面的想要做SEO优化，就是被搜索引擎搜索到，那么推荐使用第一种形式（path的形式）。如果不在乎搜索引擎优化，那么就可以使用第二种（查询字符串的形式）。