

# 蓝图:

1. 蓝图的作用就是让我们的Flask项目更加模块化, 结构更加清晰。可以将相同模块的视图函数放在同一个蓝图下, 同一个文件中, 方便管理。

2. 基本语法:

\* 在蓝图文件中导入Blueprint:

```
python
from flask import Blueprint
user_bp = Blueprint('user', __name__)
。
```

\* 在主app文件中注册蓝图:

```
python
from blueprints.user import user_bp
app.register_blueprint(user_bp)
```

3. 如果想要某个蓝图下的所有url都有一个url前缀, 那么可以在定义蓝图的时候, 指定url\_prefix参数:

```
python
user_bp = Blueprint('user', __name__, url_prefix='/user/')
```

在定义url\_prefix的时候, 要注意后面的斜杠, 如果给了, 那么以后在定义url与视图函数的时候, 就不要再在url前面加斜杠了。

4. 蓝图模版文件的查找:

\* 如果项目中的templates文件夹中有相应的模版文件, 就直接使用了。

\* 如果项目中的templates文件夹中没有相应的模版文件, 那么就到在定义蓝图的时候指定的路径中寻找。并且蓝图中指定的路径可以为相对路径, 相对的是当前这个蓝图文件所在的目录。比如:

```
news_bp =
Blueprint('news', __name__, url_prefix='/news', template_folder='zhiliao')
```

因为这个蓝图文件是在blueprints/news.py, 那么就会到blueprints这个文件夹下的zhiliao文件夹中寻找模版文件。

5. 蓝图中静态文件的查找规则:

\* 在模版文件中, 加载静态文件, 如果使用url\_for('static'), 那么就只会app指定的静态文件夹目录下查找静态文件。

- \* 如果在加载静态文件的时候，指定的蓝图的名字，比如news.static，那么就会到这个蓝图指定的static\_folder下查找静态文件。

## 6. url\_for反转蓝图中的视图函数为url:

- \* 如果使用蓝图，那么以后想要反转蓝图中的视图函数为url，那么就应该在使用url\_for的时候指定这个蓝图。比如news.news\_list。否则就找不到这个endpoint。在模版中的url\_for同样也是要满足这个条件，就是指定蓝图的名字。

- \* 即使在同一个蓝图中反转视图函数，也要指定蓝图的名字。