

expressManager-帮助文档

[Releases](#)

安装与环境配置

以 Debian 10 为例, root 权限执行.

安装 PHP

```
1 apt update
2 apt install php php-fpm
```

安装 Nginx

```
1 apt install nginx
2 systemctl status nginx #查看nginx是否正常启动(Active: active (running))
```

```
1 root@vm:/var/www/html# systemctl status nginx
2 ● nginx.service - A high performance web server and a reverse proxy server
3    Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset:
4           enabled)
5           Active: active (running) since Wed 2023-02-08 12:08:38 UTC; 57min ago
6           Docs: man:nginx(8)
7    Main PID: 31132 (nginx)
8       Tasks: 2 (limit: 1163)
9       Memory: 4.4M
10      CGroup: /system.slice/nginx.service
11              └─31132 nginx: master process /usr/sbin/nginx -g daemon on;
                  master_process on;
                  └─31133 nginx: worker process
```

配置 Nginx

```
1 rm /etc/nginx/sites-enabled/default
2 vim /etc/nginx/sites-enabled/default
```

将 `/etc/nginx/sites-enabled/default` 修改为以下内容

```
1 server {
2     listen 80 default_server;
3     listen [::]:80 default_server;
4     root /var/www/html;
5     index index.php index.html index.htm index.nginx-debian.html;
6     add_header Access-Control-Allow-Origin *;
```

```

7         add_header Access-Control-Allow-Methods 'GET, POST, OPTIONS';
8         add_header Access-Control-Allow-Headers 'DNT,X-Mx-ReqToken,Keep-Alive,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type,Authorization';
9         if ($request_method = 'OPTIONS') {
10             return 204;
11         }
12         server_name localhost;
13         location / {
14             try_files $uri $uri/ =404;
15         }
16         location ~ \.php$ {
17             include snippets/fastcgi-php.conf;
18             # With php-fpm (or other unix sockets):
19             fastcgi_pass unix:/run/php/php7.3-fpm.sock;
20             # With php-cgi (or other tcp sockets):
21             #fastcgi_pass 127.0.0.1:9000;
22         }
23     }

```

安装 MySQL

参考:<https://github.com/chens2438/docs/blob/main/debian/d-mysql.md>

配置MySQL登陆信息

```

1  cd /var/www/html
2  vim connectDB.in

```

填入你的 MySQL 登陆信息

```

1  localhost
2  root
3  root密码

```

配置 C++ 连接 MySQL

```

1  #apt install libmysqld-dev libmysqlclient-dev
2  apt install libmariadb-dev libmariadb-dev-compat libmariadb-dev

```

设置 MySQL 数据库

如图所示

- ▼ 数据库
 - ▼ expressDB
 - ▼ 表
 - > express 16K
 - > user 16K

设置express表属性

表名:

	列名	#	数据类型	非空	自增	键	默认	额外的
列	expressID	1	varchar(100)	[v]	[]	PRI		
约束	pickupID	2	varchar(100)	[]	[]			
外键	company	3	varchar(100)	[]	[]			
引用	weight	4	varchar(100)	[]	[]			
触发器	receiver	5	varchar(100)	[]	[]			
索引	receiverPhone	6	varchar(100)	[]	[]			
分区	receiverAddr	7	varchar(100)	[]	[]			
Statistics	receiverZip	8	varchar(100)	[]	[]			
DDL	sender	9	varchar(100)	[]	[]			
虚拟	senderPhone	10	varchar(100)	[]	[]			
	senderAddress	11	varchar(100)	[]	[]			
	senderZip	12	varchar(100)	[]	[]			
	picked	13	varchar(100)	[]	[]			
	inDate	14	varchar(100)	[]	[]			

设置user表属性

表名:
 引擎:
 自增:
 字符集:
 排序规则:
 描述:

	列名	#	数据类型	非空	自增	键	默认	额外的
列	phone	1	varchar(100)	[v]	[]			
约束	passwd	2	varchar(100)	[]	[]			
外键	userType	3	varchar(100)	[]	[]			
引用								

编译

```

1 | cd /var/www/html && rm -rf expressManager/
2 | git clone https://github.com/chens2438/expressManager.git

1 | g++ expressManager/cpp/main.cpp -std=c++17 -o expressManager/cpp/main `mysql_config`
  | --cflags --libs` && chmod 777 -R expressManager/

```

访问

`http://IP地址/expressManager`