

# Deepfox 快速上手指南

云数信息科技（深圳）有限公司

## 添加用户



使用管理员账户登陆后，[用户管理系统](#)菜单会出现在菜单栏的最下方。

在用户管理系统界面中，您可以方便的注册新用户、重置用户密码或禁用用户。

用户管理系统

注册新用户

显示 10 项结果

用户名	注册时间	最近登录时间
adminx	2017-05-27 01:48:09	2017-05-27 01:48:13
admin <span>管理员</span>	2017-05-27 01:33:17	2017-07-24 18:32:16
bdcsc <span>管理员</span>	2017-06-02 09:43:48	2017-06-14 18:29:06
valency	2017-05-27 01:47:50	2017-05-27 01:47:53

显示第 1 至 4 项结果，共 4 项

**注意!** 新增用户时系统会预设一套密码，请仔细保管并于首次登陆后修改。

# 新增模型

---

在 Deepfox 中新增模型非常容易，步骤如下：

1. 打开[数据分析系统](#)的模型管理界面：



2. 点击[新增模型](#)按钮：



3. 输入 HiveQL 或 Python 代码，并提供模型名称、模型代号（即 ID）、参数列表（仅限 Python），点击确定即可生成模型。

**注意！** 如需以表格形式显示模型分析结果，则需使得输出格式为二维数组；

如采用 Python 代码类型，输出结果请赋值给 `query_results` 变量；

如需进行 Hive 查询，请使用预设方法 `hive_query_json`。

新增 / 修改模型

✕

1	<code>SELECT gender, COUNT(*) FROM customer_info GROUP BY gender</code>
---	---

模型名称	银行项目: 性别统计分布	模型代号	bank:model:gender	代码类型	HiveQL
参数列表	逗号分隔, 仅对 Python 代码类型生效				
图表标题	NULL	图表类型	表格	图表表头	逗号分隔

确定

新增 / 修改模型

✕

1	<code>statement = "SELECT t.r, COUNT(*) FROM (SELECT CASE "</code>
2	<code>statement += "WHEN age BETWEEN 0 AND 30 THEN '0-29' "</code>
3	<code>statement += "WHEN age BETWEEN 30 AND 40 THEN '30-39' "</code>
4	<code>statement += "WHEN age BETWEEN 40 AND 50 THEN '40-49' "</code>
5	<code>statement += "ELSE '50-' END AS r FROM customer_info "</code>
6	<code>statement += "WHERE residence " + RESIDENCE</code>
7	<code>statement += " AND age " + AGE</code>
8	<code>statement += " AND years_registered " + YEARS_REGISTERED</code>
9	<code>statement += " AND avg_saving_all_time " + AVG_SAVING_ALL_TIME</code>
10	<code>statement += " AND e_bank_id " + E_BANK_ID</code>
11	<code>statement += " AND mobile_bank_id " + MOBILE_BANK_ID</code>
12	<code>statement += " AND wechat_bank_id " + WECHAT_BANK_ID</code>
13	<code>statement += ") AS t GROUP BY t.r"</code>
14	<code>query_results = hive_query_json(statement)</code>

模型名称	银行项目: 年龄统计分布	模型代号	bank:model:age_hist	代码类型	HiveQL
参数列表	RESIDENCE,AGE,YEARS_REGISTERED,AVG_SAVING_ALL_TIME,E_BANK_ID,MOBILE_BANK_ID,WECHAT_BANK				
图表标题	年龄统计分布	图表类型	无	图表表头	逗号分隔

确定

# 运行模型

从模型预览界面点击运行即可立刻执行模型，如下图所示：

`bank:model:credit_card_activate` 银行项目：每日信用卡激活情况

```
1 SELECT * FROM (
2 SELECT t, credit_card_new_register FROM daily_summary
3 ORDER BY t DESC LIMIT 7) AS a
4 ORDER BY a.t ASC
```

ID	执行账户 ID	运行时间
19	admin	2017-06-15 16:23:06
13	admin	2017-05-27 17:43:43

仅显示最近 5 条运行结果，[查看全部](#)

运行

系统亦支持直接调用 API 来获取运行历史或运行模型，例如：

```
Request URL: http://127.0.0.1:9004/api/franz/data/history/get/
Request Method: POST
Form Data:
report: bank:model:credit_card_activate
conf: null
```

上述请求将获取模型 `bank:model:credit_card_activate` 的最近运行历史，如果模型曾经于某个时间段内运行过，我们不建议重新运行，而应该采用缓存的运行结果作为输出。

ⓘ 该模型已使用指定的参数在下列时间运行过，请问是否重复运行？

```
2017-06-15 16:23:06
```

如果确定要运行模型，可以调用类似以下的 API：

```
Request URL: http://127.0.0.1:9004/api/franz/data/report/execute/  
Request Method: POST  
Form Data:  
report: bank:model:credit_card_activate  
conf: null
```

运行结果如下图所示：

```
account: "admin"  
conf: ""  
id: 19  
report: "bank:model:credit_card_activate"  
result: "[["2016-08-08", 2658.0], ["2016-08-09", 2276.0], ["2016-08-10",  
2745.0], ["2016-08-11", 2612.0], ["2016-08-12", 2090.0], ["2016-08-13",  
2214.0], ["2016-08-14", 2713.0]]"  
t_execute: "2017-06-15T16:23:06.716540"
```

## 更多细节

---

如需了解更多 Deepfox 使用方法，请联系您的销售经理或技术支援。

我们提供完整的 Deepfox 培训课程，如有需要，请联系您的销售经理。