# 1 登录后台

第1份原始数据编辑从UnityGameTools Flutter 后台管理系统开始。

编辑一个信号名称，首字母大写开头,单词之间用下划线间隔。

比如：Game\_Role\_Asset\_Store

## Android Studio Flutter

### 生成谷歌结构体

在项目根目录打开终端：

|  |
| --- |
| protoc --dart\_out=grpc:lib/src/generated -Iprotos protos/\*.proto |

## Visual Studio Code

RUN CMD: 在\game\_quantum\_service\grpc\_go> 路径下打开终端  
protoc --go-grpc\_out=. trigger\_data.proto  
protoc --go\_out=. trigger\_data.proto  
  
protoc --go-grpc\_out=. game\_role\_data.proto  
protoc --go\_out=. game\_role\_data.proto  
  
protoc --go-grpc\_out=. game\_role\_asset\_store.proto  
protoc --go\_out=. game\_role\_asset\_store.proto  
  
protoc --go-grpc\_out=. game\_signal\_dictionary.proto  
protoc --go\_out=. game\_signal\_dictionary.proto  
  
protoc --go-grpc\_out=. game\_signal\_parameter\_data.proto  
protoc --go\_out=. game\_signal\_parameter\_data.proto  
  
protoc --go-grpc\_out=. game\_signal\_parameter\_dictionary.proto  
protoc --go\_out=. game\_signal\_parameter\_dictionary.proto  
  
protoc --go-grpc\_out=. search\_data.proto  
protoc --go\_out=. search\_data.proto  
  
  
protoc --go-grpc\_out=. system\_func\_list.proto  
protoc --go\_out=. system\_func\_list.proto

# 2 Data Tools Create

## 1 go db com struct

Go com 文件名.go

## 2 go DAO

## 3 go grpc system

接受客户端的远程调用方法

## 4 Go导出游戏C#数据.go

这里有Dart Client Data Com

## 5 Dart data com.dart

## 6 Dart GRPC System

跟服务器通信的系统

6