



《手写OS操作系统》小班二期招生，全程直播授课，大牛带你掌握硬核技术！

点此

慕课网首页

免费课

实战课

体系课

慕课教程

专栏

手记

企业服务



我的

从所有教程的词条中查询...

首页 > 慕课教程 > Go工程师体系课全新版 > 2. protobuf定义

## 全部开发者教程

14. go操作Rocketmq开发环境搭建

15. 什么是服务雪崩

16. grpc的超时和重试

17. 常见的幂等性解决方案

### 第19周 链路追踪、限流、熔断、降级

1. 什么是链路追踪

2. 链路追踪技术选型

3. jaeger安装和架构

4. opentracing解析

5. grpc下添加jaeger

6. 熔断、限流技术选型

### 第20周 api网关、部署

1. api网关对比

2. kong的安装和配置

3. 配置jwt

4. 什么是敏捷开发

5. 敏捷开发流程和配置



bobby · 更新于 2022-11-08

← 上一节 1. model定义 3. 商品列表页功能 下一节 →

## prot文件

goods.proto

<> 代码块

预览 复制

```
1 syntax = "proto3";
2 import "google/protobuf/empty.proto";
3 option go_package = ".;proto";
4
5 service Goods{
6     //商品接口
7     rpc GoodsList(GoodsFilterRequest) returns(GoodsListResponse);
8     //现在用户提交订单有多个商品，你得批量查询商品的信息吧
9     rpc BatchGetGoods(BatchGoodsIdInfo) returns(GoodsListResponse); //批量获取商品信息
10    rpc CreateGoods(CreateGoodsInfo) returns (GoodsInfoResponse);
11    rpc DeleteGoods(DeleteGoodsInfo) returns (google.protobuf.Empty);
12    rpc UpdateGoods(CreateGoodsInfo) returns (google.protobuf.Empty);
13    rpc GetGoodsDetail(GoodInfoRequest) returns(GoodsInfoResponse);
14
15    //商品分类
16    rpc GetAllCategorysList(google.protobuf.Empty) returns(CategoryListResponse);
17    //获取子分类
18    rpc GetSubCategory(CategoryListRequest) returns(SubCategoryListResponse);
19    rpc CreateCategory(CategoryInfoRequest) returns(CategoryInfoResponse); //新建
20    rpc DeleteCategory(DeleteCategoryRequest) returns(google.protobuf.Empty); //删除
21    rpc UpdateCategory(CategoryInfoRequest) returns(google.protobuf.Empty); //修改
22
23    //品牌和轮播图
24    rpc BrandList(BrandFilterRequest) returns(BrandListResponse); //批量获取品牌信息
25    rpc CreateBrand(BrandRequest) returns(BrandInfoResponse); //新建品牌信息
26    rpc DeleteBrand(BrandRequest) returns(google.protobuf.Empty); //删除品牌
27    rpc UpdateBrand(BrandRequest) returns(google.protobuf.Empty); //修改品牌信息
28
29    //轮播图
30    rpc BannerList(google.protobuf.Empty) returns(BannerListResponse); //获取轮播图
31    rpc CreateBanner(BannerRequest) returns(BannerResponse); //添加banner图
32    rpc DeleteBanner(BannerRequest) returns(google.protobuf.Empty); //删除轮播图
33    rpc UpdateBanner(BannerRequest) returns(google.protobuf.Empty); //修改轮播图
34
35    //品牌分类
36    rpc CategoryBrandList(CategoryBrandFilterRequest) returns(CategoryBrandListResponse);
37    //通过category获取brands
38    rpc GetCategoryBrandList(CategoryInfoRequest) returns(BrandListResponse);
39    rpc CreateCategoryBrand(CategoryBrandRequest) returns(CategoryBrandResponse);
40    rpc DeleteCategoryBrand(CategoryBrandRequest) returns(google.protobuf.Empty);
41    rpc UpdateCategoryBrand(CategoryBrandRequest) returns(google.protobuf.Empty);
```

意见反馈

收藏教程

标记书签



```
44 message CategoryListRequest {
45     int32 id = 1;
46     int32 level = 2;
47 }
48
49
50 message CategoryInfoRequest {
51     int32 id = 1;
52     string name = 2;
53     int32 parentCategory = 3;
54     int32 level = 4;
55     bool isTab = 5;
56 }
57
58
59 message DeleteCategoryRequest {
60     int32 id = 1;
61 }
62
63 message QueryCategoryRequest {
64     int32 id = 1;
65     string name = 2;
66 }
67
68 message CategoryInfoResponse {
69     int32 id = 1;
70     string name = 2;
71     int32 parentCategory = 3;
72     int32 level = 4;
73     bool isTab = 5;
74 }
75
76 message CategoryListResponse {
77     int32 total = 1;
78     repeated CategoryInfoResponse data = 2;
79     string jsonData = 3;
80 }
81
82 message SubCategoryListResponse {
83     int32 total = 1;
84     CategoryInfoResponse info = 2;
85     repeated CategoryInfoResponse subCategorys = 3;
86 }
87
88
89
90 message CategoryBrandFilterRequest {
91     int32 pages = 1;
92     int32 pagePerNums = 2;
93 }
94
95 message FilterRequest {
96     int32 pages = 1;
97     int32 pagePerNums = 2;
98 }
99
100 message CategoryBrandRequest{
101     int32 id = 1;
102     int32 categoryId = 2;
103     int32 brandId = 3;
104 }
105 message CategoryBrandResponse{
106     int32 id = 1;
```

```
109 }
110
111 message BannerRequest {
112     int32 id = 1;
113     int32 index = 2;
114     string image = 3;
115     string url = 4;
116 }
117
118 message BannerResponse {
119     int32 id = 1;
120     int32 index = 2;
121     string image = 3;
122     string url = 4;
123 }
124
125 message BrandFilterRequest {
126     int32 pages = 1;
127     int32 pagePerNums = 2;
128 }
129
130 message BrandRequest {
131     int32 id = 1;
132     string name = 2;
133     string logo = 3;
134 }
135
136 message BrandInfoResponse {
137     int32 id = 1;
138     string name = 2;
139     string logo = 3;
140 }
141
142 message BrandListResponse {
143     int32 total = 1;
144     repeated BrandInfoResponse data = 2;
145 }
146
147 message BannerListResponse {
148     int32 total = 1;
149     repeated BannerResponse data = 2;
150 }
151
152 message CategoryBrandListResponse {
153     int32 total = 1;
154     repeated CategoryBrandResponse data = 2;
155 }
156
157
158
159 message BatchGoodsIdInfo {
160     repeated int32 id = 1;
161 }
162
163
164 message DeleteGoodsInfo {
165     int32 id = 1;
166 }
167
168 message CategoryBriefInfoResponse {
169     int32 id = 1;
170     string name = 2;
171 }
172
```

```
174     int32 id = 1;
175     bool isTab = 2;
176 }
177
178 message GoodInfoRequest {
179     int32 id = 1;
180 }
181
182 message CreateGoodsInfo {
183     int32 id = 1;
184     string name = 2;
185     string goodsSn = 3;
186     int32 stocks = 7; //库存,
187     float marketPrice = 8;
188     float shopPrice = 9;
189     string goodsBrief = 10;
190     string goodsDesc = 11;
191     bool shipFree = 12;
192     repeated string images = 13;
193     repeated string descImages = 14;
194     string goodsFrontImage = 15;
195     bool isNew = 16;
196     bool isHot = 17;
197     bool onSale = 18;
198     int32 categoryId = 19;
199     int32 brandId = 20;
200 }
201
202 message GoodsReduceRequest {
203     int32 GoodsId = 1;
204     int32 nums = 2;
205 }
206
207 message BatchCategoryInfoRequest {
208     repeated int32 id = 1;
209     int32 goodsNums = 2;
210     int32 brandNums = 3;
211 }
212
213 message GoodsFilterRequest {
214     int32 priceMin = 1;
215     int32 priceMax = 2;
216     bool isHot = 3;
217     bool isNew = 4;
218     bool isTab = 5;
219     int32 topCategory = 6;
220     int32 pages = 7;
221     int32 pagePerNums = 8;
222     string keyWords = 9;
223     int32 brand = 10;
224 }
225
226
227 message GoodsInfoResponse {
228     int32 id = 1;
229     int32 categoryId = 2;
230     string name = 3;
231     string goodsSn = 4;
232     int32 clickNum = 5;
233     int32 soldNum = 6;
234     int32 favNum = 7;
235     float marketPrice = 9;
236     float shopPrice = 10;
237     string goodsDesc = 11;
238     string goodsBrief = 12;
239     string goodsFrontImage = 15;
240     repeated string images = 13;
241     repeated string descImages = 14;
242     bool shipFree = 12;
243     bool onSale = 18;
244     bool isHot = 17;
245     bool isNew = 16;
246     bool isTab = 5;
247     int32 topCategory = 6;
248     int32 pages = 7;
249     int32 pagePerNums = 8;
250     string keyWords = 9;
251     int32 brand = 10;
252 }
```

```
239     bool shipFree = 13;
240     repeated string images = 14;
241     repeated string descImages = 15;
242     string goodsFrontImage = 16;
243     bool isNew = 17;
244     bool isHot = 18;
245     bool onSale = 19;
246     int64 addTime = 20;
247     CategoryBriefInfoResponse category = 21;
248     BrandInfoResponse brand = 22;
249 }
250
251 message GoodsListResponse {
252     int32 total = 1;
253     repeated GoodsInfoResponse data = 2;
254 }
255
```

### proto生成

<> 代码块

```
1  protoc -I . goods.proto --go_out=plugins=grpc:.
```

1. model定义 ◀ 上一节      下一节 ▶ 3. 商品列表页功能

✎ 我要提出意见反馈