

PB-gen pbentity

 v2.4

proto

Content Menu

-
-
- [gen daoentity](#)
 -
 -

```

$ gf gen pbentity -h
USAGE
    gf gen pbentity [OPTION]

OPTION
    -p, --path            directory path for generated files storing
    -k, --package         package path for all entity proto files
    -l, --link            database configuration, the same as the ORM
configuration of GoFrame
    -t, --tables          generate models only for given tables,
multiple table names separated with ','
    -f, --prefix          add specified prefix for all entity names
and entity proto files
    -r, --removePrefix    remove specified prefix of the table,
multiple prefix separated with ','
    -rf, --removeFieldPrefix remove specified prefix of the field,
multiple prefix separated with ','
    -n, --nameCase        case for message attribute names, default
is "Camel":
                                | Case           | Example           |
                                |-----|-----|
                                | Camel         | AnyKindOfString  |
                                | CamelLower    | anyKindOfString  |
default
                                | Snake         | any_kind_of_string |
                                | SnakeScreaming | ANY_KIND_OF_STRING |
                                | SnakeFirstUpper | rgb_code_md5      |
                                | Kebab         | any-kind-of-string |
                                | KebabScreaming | ANY-KIND-OF-STRING |
    -j, --jsonCase        case for message json tag, cases are the
same as "nameCase", default "CamelLower".
                                set it to "none" to ignore json tag
generating.
    -o, --option          extra protobuf options
    -h, --help            more information about this command

EXAMPLE
    gf gen pbentity
    gf gen pbentity -l "mysql:root:12345678@tcp(127.0.0.1:3306)/test"
    gf gen pbentity -p ./protocol/demos/entity -t user,user_detail,
user_login
    gf gen pbentity -r user_ -k github.com/gogf/gf/example/protobuf
    gf gen pbentity -r user_

CONFIGURATION SUPPORT
Options are also supported by configuration file.
It's suggested using configuration file instead of command line
arguments making producing.
The configuration node name is "gf.gen.pbentity", which also supports
multiple databases, for example(config.yaml):
gfcli:
  gen:
    - pbentity:
        link:      "mysql:root:12345678@tcp(127.0.0.1:3306)/test"
        path:      "protocol/demos/entity"
        tables:    "order,products"
        package:   "demos"
    - pbentity:
        link:      "mysql:root:12345678@tcp(127.0.0.1:3306)/primary"
        path:      "protocol/demos/entity"
        prefix:    "primary_"
        tables:    "user, userDetails"
        package:   "demos"
        option:
            option go_package      = "protobuf/demos";
            option java_package    = "protobuf/demos";
            option php_namespace  = "protobuf/demos";

```

✔ makemake pbentity

gfcli.gen.pbentity			-
path	manifest /protobuf /pbentity	proto	protobuf /pbentity
package	go.mod	protogo_packagepackage	-
link		mysql, postgresql, dsn ORM	-
prefix			order_ user_
removePrefix		,	gf_
removeFieldPrefix		,	f_
tables			user, user_detail
nameCase	CamelLower	messageCamelCamelLowerSnakeSnakeScreamingSnakeFirstUpperKebabKebabScreaming	Snake
option		proto option	

gen daoentity

- entityGolang
- gen daoentityGolangHTTPHTTpgen daoentityinternalHTTP API
- GRPCgen daoentityGRPCGRPCprotoGRPCgen pbentitypbentity protoGRPCgen daoentity
- pbentityentitygen daoentity