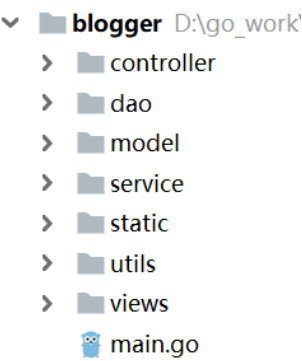


# 一、项目目录规范



- model: 实体
- dao: 数据层
- service: 业务逻辑
- controller: 页面控制相关
- static: css js
- utils: 工具
- views: HTML 模板
- main.go: 入口，定义路由

# 二、需求分析

- 

# 三、数据库分析

- article 文章表

id	bigint	20		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	文章id
category_id	bigint	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	分类id
content	longtext			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	文章内容
title	varchar	1024		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	文章标题
view_count	int	255		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	阅读次数
comment_count	int	255		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	评论次数
username	varchar	128		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	作者
status	int	10	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	状态，正常为1
summary	varchar	256		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	文章摘要
create_time	timestamp		CURRENT_TIMESTAMP	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	发布时间
update_time	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	更新时间

- category: 分类表

<input type="checkbox"/>	列名	数据类型	长度	默认	主键?	非空?	Unsigned	自增?	Zerofill?	注释
<input checked="" type="checkbox"/>	id	bigint	20		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	分类id
<input type="checkbox"/>	category_name	varchar	255		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	分类名字
<input type="checkbox"/>	category_no	int	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	分类排序
<input type="checkbox"/>	create_time	timestamp		CURRENT_TIM	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	update_time	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

### ● 评论表 comment

<input type="checkbox"/>	列名	数据类型	长度	默认	主键?	非空?	Unsigned	自增?	Zerofill?	注释
<input checked="" type="checkbox"/>	id	bigint	20		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	评论id
<input type="checkbox"/>	content	text			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	评论内容
<input type="checkbox"/>	username	varchar	64		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	评论作者
<input type="checkbox"/>	create_time	timestamp		CURRENT_TIM	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	评论发布时间
<input type="checkbox"/>	status	int	255	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	评论状态: 0, 删除: 1.
<input type="checkbox"/>	article_id	bigint	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

### ● leave 留言表

<input type="checkbox"/>	列名	数据类型	长度	默认	主键?
<input checked="" type="checkbox"/>	id	bigint	20		<input checked="" type="checkbox"/>
<input type="checkbox"/>	username	varchar	255		<input type="checkbox"/>
<input type="checkbox"/>	email	varchar	255		<input type="checkbox"/>
<input type="checkbox"/>	content	text			<input type="checkbox"/>
<input type="checkbox"/>	create_time	timestamp		CURRENT_TIM	<input type="checkbox"/>
<input type="checkbox"/>	update_time	timestamp			<input type="checkbox"/>

## 四、主页的实现

### 1. 实体类

- 分类的结构体
- 文章的结构体（可以不写文章内容，因为是大文本）
- model/category.go

```
package model

//`id``category_name``category_no``create_time``update_time`

// 定义分类结构体
type Category struct {
    CategoryId int64 `db:"id"`
    CategoryName string `db:"category_name"`
    CategoryNo int `db:"category_no"`
}
```

### ● model/article.go

```
package model

import "time"

//`id``category_id``content``title``view_count``comment_count``username``STATUS`
//`summary``create_time``update_time`
```

```

// 定义文章结构体
type ArticleInfo struct {
    Id          int64 `db:"id"`
    CategoryId  int64 `db:"category_id"`
    // 文章摘要
    Summary     string `db:"summary"`
    Title       string `db:"title"`
    ViewCount   uint32 `db:"view_count"`
    // 时间
    CreateTime  time.Time `db:"create_time"`
    CommentCount uint32 `db:"comment_count"`
    Username    string `db:"username"`
}

// 用于文章详情页的实体
// 为了提升效率
type ArticleDetail struct {
    ArticleInfo
    // 文章内容
    Content string `db:"content"`
    Category
}

// 用于文章上下页
type ArticleRecord struct {
    ArticleInfo
    Category
}

```

## 2. 数据层

- `init()`数据库初始化函数
- 分类相关的操作（添加，查询，查 1 个分类，查多个分类，查所有分类）
- 文章相关的操作（添加文章（投稿），查询所有文章，根据文章 id 查看内容）
- `dao/db/db.go`

```

package db

import (
    _ "github.com/go-sql-driver/mysql"
    "github.com/jmoiron/sqlx"
)

```

```

var (
    DB *sqlx.DB
)

// 初始化
func Init(dns string) error {
    var err error
    DB, err = sqlx.Open("mysql", dns)
    if err != nil {
        return err
    }
    // 查看是否连成功
    err = DB.Ping()
    if err != nil {
        return err
    }
    DB.SetMaxOpenConns(100)
    DB.SetMaxIdleConns(16)
    return nil
}

```

- dao/db/category.go

```

package db

import "go_5_blog/model"

// 添加分类
func InsertCategory(category *model.Category) (categoryId int64, err error) {
    sqlstr := "insert into category(category_name,category_no) value(?,?)"
    result, err := DB.Exec(sqlstr, category.CategoryName, category.CategoryNo)
    if err != nil {
        return
    }
    categoryId, err = result.LastInsertId()
    return
}

// 获取单个分类
func GetCategoryById(id int64) (category *model.Category, err error) {
    category = &model.Category{}
    sqlstr := "select id ,category_name,category_no from category where id = ?"
    err = DB.Get(category, sqlstr, id)
    return
}

```

--

### 3. 业务逻辑层

- 

### 4. controller 层

- 

### 5. 项目入口

- 

## 五、投稿的实现

### 1. 业务逻辑层

- 

### 2. controller 层

-

## 六、文章详情页的实现

### 1. 实体类

- 

### 2. 数据层

- 

### 3. 业务逻辑层

- 

### 4. controller 层

- 

## 七、文章评论的实现

## 1. 实体类

- 

## 2. 数据层

- 

## 3. 业务逻辑层

- 

## 4. controller 层

- 

# 八、博客留言的实现

## 1. 实体类

-

## 2. 数据层

- 

## 3. 业务逻辑层

- 

## 4. controller 层

-