



This repository ▾

Search or type a command 🔍

ExploreGistBlogHelp

Macro-Bin📄✂️📌

📄 nswbmw / N-blog

👁 Watch ▾ 92

★ Star 387

🍴 Fork 245

HomePagesHistory

New Page

第9章 增加标签和标签页面

Edit PagePage HistoryClone URL

现在我们来给博客增加标签和标签页。

我们设定：每篇文章最多有三个标签（少于三个也可以），当点击主页左侧标签页（/tag）链接时，跳转到标签页并列出所有已存在标签；当点击任意一个标签链接时，跳转到该标签页并列出所有含有该标签的文章。

添加标签

首先我们来实现添加标签功能。

修改 `post.ejs`，在 `<input type="text" name="title" />
` 后添加：

标签: `
`
`<input type="text" name="tag1" /><input type="text" name="tag2" /><input type="text" name="tag3" />
`

打开 `index.js`，修改 `app.post('/post')` 将

`var currentUser = req.session.user,
post = new Post(currentUser.name, req.body.title, req.body.post);`

修改为：

`var currentUser = req.session.user,
tags = [{"tag":req.body.tag1},{"tag":req.body.tag2},{"tag":req.body.tag3}],
post = new Post(currentUser.name, req.body.title, tags, req.body.post);`

修改 `post.js`，将

`function Post(name, title, post) {
 this.name = name;
 this.title= title;
 this.post = post;
}`

修改为：

`function Post(name, title, tags, post) {
 this.name = name;
 this.title = title;
 this.tags = tags;
 this.post = post;
}`

将

`var post = {
 name: this.name,
 time: time,
 title: this.title,
 post: this.post,
 comments: []
};`

修改为：

`var post = {
 name: this.name,
 time: time,
 title: this.title,
 tags: this.tags,`

https://github.com/nswbmw/N-blog/wiki/第9章--增加标签和标签页面

1/4

```
    post: this.post,
    comments: []
  };
```

现在我们就可以在发表文章的时候添加标签了，接下来我们修改 `index.ejs` 、 `user.ejs` 和 `article.ejs` 来显示文章的标签。

修改 `index.ejs` 、 `user.ejs` 和 `article.ejs`，将

```
<p class="info">
  作者: <a href="/u/<%= post.name %>"><%= post.name %></a> |
  日期: <%= post.time.minute %>
</p>
```

修改为：

```
<p class="info">
  作者: <a href="/u/<%= post.name %>"><%= post.name %></a> |
  日期: <%= post.time.minute %> |
  标签:
  <% post.tags.forEach(function (tag, index) { %>
    <% if (tag.tag) { %>
      <a class="tag" href="/tags/<%= tag.tag %>"><%= tag.tag %></a>
    <% } %>
  <% }) %>
</p>
```

最后，在 `style.css` 中添加如下样式：

```
.tag{background-color:#ff0000;border-radius:3px;font-size:14px;color:#ffffff;display:inline-block;padding:0 5px;margin-bottom:8px;
.tag:hover{text-decoration:none;background-color:#ffffff;color:#000000;-webkit-transition:color .2s linear;}
```

至此，我们给博客添加了标签功能。赶紧看看效果吧！

添加标签页

接下来我们给博客增加标签页。

修改 `header.ejs`，在 `archive` 下一行添加：

```
<span><a title="标签" href="/tags">tags</a></span>
```

修改 `index.js`，在 `app.get('/archive')` 后添加如下代码：

```
app.get('/tags', function (req, res) {
  Post.getTags(function (err, posts) {
    if (err) {
      req.flash('error', err);
      return res.redirect('/');
    }
    res.render('tags', {
      title: '标签',
      posts: posts,
      user: req.session.user,
      success: req.flash('success').toString(),
      error: req.flash('error').toString()
    });
  });
});
```

打开 `post.js`，在最后添加：

```
//返回所有标签
Post.getTags = function(callback) {
  //打开数据库
  mongodb.open(function (err, db) {
    if (err) {
      return callback(err);
    }
    db.collection('posts', function (err, collection) {
      if (err) {
```

```
    mongodb.close();
    return callback(err);
  }
  //distinct 用来找出给定键的所有不同值
  collection.distinct("tags.tag", function (err, docs) {
    mongodb.close();
    if (err) {
      return callback(err);
    }
    callback(null, docs);
  });
});
});
};
```

注意：这里我们用了 `distinct`（详见《[mongodb权威指南](#)》）返回 `tag` 键的所有不同值，因为有时候我们发表文章的标签是一样的，所以这样避免了获取重复的标签。

在 `views` 文件夹下新建 `tags.ejs`，添加如下代码：

```
<%- include header %>
<% posts.forEach(function (tag, index) { %>
  <% if (tag) { %>
    <a class="tag" href="/tags/<%= tag %>"><%= tag %></a>
  <% } %>
<% }) %>
<%- include footer %>
```

至此，我们就给博客添加了标签页。

添加特定标签的页面

现在我们来添加特定标签的页面，即当点击任意一个标签链接时，跳转到该标签页并列出所有含有该标签的文章。

修改 `post.js`，在最后添加如下代码：

```
//返回含有特定标签的所有文章
Post.getTag = function(tag, callback) {
  mongodb.open(function (err, db) {
    if (err) {
      return callback(err);
    }
    db.collection('posts', function (err, collection) {
      if (err) {
        mongodb.close();
        return callback(err);
      }
      //通过 tags.tag 查询并返回只含有 name、time、title 键的文档组成的数组
      collection.find({
        "tags.tag": tag
      }, {
        "name": 1,
        "time": 1,
        "title": 1
      }).sort({
        time: -1
      }).toArray(function (err, docs) {
        mongodb.close();
        if (err) {
          return callback(err);
        }
        callback(null, docs);
      });
    });
  });
};
```

修改 `index.js`，在 `app.get('/tags')` 后添加如下代码：

```
app.get('/tags/:tag', function (req, res) {
  Post.getTag(req.params.tag, function (err, posts) {
    if (err) {
```

```
    req.flash('error',err);
    return res.redirect('/');
  }
  res.render('tag', {
    title: 'TAG:' + req.params.tag,
    posts: posts,
    user: req.session.user,
    success: req.flash('success').toString(),
    error: req.flash('error').toString()
  });
});
});
});
```

在 `views` 文件夹下新建 `tag.ejs`，添加如下代码：

```
<%- include header %>
<ul class="archive">
  <% var lastYear = 0 %>
  <% posts.forEach(function (post, index) { %>
    <% if (lastYear != post.time.year) { %>
      <li><h3><%= post.time.year %></h3></li>
    <% lastYear = post.time.year %>
    <li><time><%= post.time.day %></time></li>
    <li><a href="/u/<%= post.name %>/<%= post.time.day %>/<%= post.title %>"><%= post.title %></a></li>
  <% }) %>
</ul>
<%- include footer %>
```

最后，别忘了修改 `edit.ejs`，为了保持和 `post.ejs` 一致。打开 `edit.ejs`，在 `正文:
` 前添加如下代码：

```
标签: <br />
<input type="text" name="tag1" value="<%= post.tags[0].tag %>" disabled="disabled" />
<input type="text" name="tag2" value="<%= post.tags[1].tag %>" disabled="disabled" />
<input type="text" name="tag3" value="<%= post.tags[2].tag %>" disabled="disabled" /><br />
```

这里我们设定了编辑时不能编辑文章的标签。

现在，我们的博客就增加了标签和标签页的功能。

Last edited by nswbmw, a month ago