个人Word模板教程

|  |  |
| --- | --- |
| 撰写者： | 王志强 |
| 其他参与者： | 修改者自行添加 |
| 发布时间： | 2020-10-22 |
| 最后一次更新时间： | 2022-08-31 19:22 |
| 版本号： | 5 |

目录

[个人Word模板教程 1](#_Toc112866551)

[目录 1](#_Toc112866552)

[1 引言 1](#_Toc112866553)

[2 使用说明 1](#_Toc112866554)

[2.1 外部 2](#_Toc112866555)

[2.2 内部 2](#_Toc112866556)

[2.2.1 页眉页脚 2](#_Toc112866557)

[2.2.2 封面 2](#_Toc112866558)

[2.2.3 目录 3](#_Toc112866559)

[2.2.4 样式 3](#_Toc112866560)

[3 文档规范 8](#_Toc112866561)

[3.1 页面设置 8](#_Toc112866562)

[3.1.1 页边距 8](#_Toc112866563)

[3.1.2 纸张 8](#_Toc112866564)

[3.1.3 布局 8](#_Toc112866565)

[3.2 页眉 8](#_Toc112866566)

[3.3 页脚 9](#_Toc112866567)

[3.4 封面 9](#_Toc112866568)

[3.5 目录 9](#_Toc112866569)

[3.6 样式 9](#_Toc112866570)

[3.6.1 快捷键 9](#_Toc112866571)

[3.6.2 表内容 10](#_Toc112866572)

[3.6.3 图 10](#_Toc112866573)

[3.6.4 代码 10](#_Toc112866574)

[3.6.5 引用 11](#_Toc112866575)

[3.6.6 无序列表 11](#_Toc112866576)

[3.6.7 有序列表 11](#_Toc112866577)

[3.6.8 正文 12](#_Toc112866578)

[3.6.9 标题 12](#_Toc112866579)

[3.6.10 标题1、2、3、4、5、6 12](#_Toc112866580)

[3.6.11 题注 12](#_Toc112866581)

[3.6.12 终端代码 13](#_Toc112866582)

[4 撰写建议 13](#_Toc112866583)

[5 Word技巧 13](#_Toc112866584)

[5.1 查找与替换 13](#_Toc112866585)

[5.1.1 样式查找与替换 13](#_Toc112866586)

[5.2 修改无序列表前的“项目符号” 14](#_Toc112866587)

[5.3 表格样式设置 14](#_Toc112866588)

[5.4 delete的使用 14](#_Toc112866589)

[5.5 样式常用设置 14](#_Toc112866590)

[5.5.1 设置后续段落样式 14](#_Toc112866591)

[5.5.2 为样式添加快捷键 14](#_Toc112866592)

[5.5.3 修改段落间距 14](#_Toc112866593)

[6 字体版权问题 14](#_Toc112866594)

[7 样式测试 14](#_Toc112866595)

[7.1 测试正文 15](#_Toc112866596)

[7.2 英文 15](#_Toc112866597)

[7.3 中文 15](#_Toc112866598)

[7.4 测试列表 15](#_Toc112866599)

[7.5 测试标题 16](#_Toc112866600)

[7.6 测试标题 16](#_Toc112866601)

[7.7 测试引用 16](#_Toc112866602)

[7.8 测试代码 16](#_Toc112866603)

[7.9 测试表 17](#_Toc112866604)

[7.10 测试图 17](#_Toc112866605)

[8 修订记录 18](#_Toc112866606)

# 引言

我们在深入了解Word之前都会认为Word很简单，且把它当做可以插入图片的记事本使用。但是这样一来，写出的文档往往杂乱无章，优秀的资源没有得到充分的利用，文档的价值也很难得到体现。

当我们简单学习了一些Word知识后，或许能写出更好的文档，有标题、字体好看、间距合理……甚至如果有耐心，可以把每个细节都弄得非常完美。但是，这时我们往往会发现一个新的问题——耗时太长。并且还有一个问题——修改困难。往往你改了一点内容都可能导致格式变乱，然后你又需要耐心地调整。每改一次调整一次，每改一次调整一次，每改一次调整一次……

于是你想到了一个好主意——先写内容，再调格式。确保内容无需修改的时候再来处理格式问题，处理好之后不再更改文档内容。但是这治标不治本。毕竟后续你还是可能要改的。比如某人发现某个地方需要完善，结果他一改，格式就混乱了、不统一了。

于是你发现你需要进一步学习了，很快，你意识到了自己需要学什么——样式。通过样式，可以大大提高可维护性，你只要写时应用好样式，写完后尽管改，改后觉得样式太丑，直接改样式就行，无需去一个个地调。但是，很快你又遇到了一个新的问题——样式对于每个文档都是不一样的。也就是说，每换一篇文档，你就需要重新调整一波样式，把你熟悉的、喜欢的样式调好后再写文档。但是这样实在太麻烦了，有什么方法可以保存样式吗？

于是，你发现了模板。你发现模板真是个好东西。只要调整好模板里的样式，你只管使用就行了，无需在每次写文档时关心样式问题。这就是我推荐使用模板的原因。而本文就使用了模板——《个人Word模板.dotx》（注意后缀不同哦）。

模板除了能包含样式外，还能包含很多内容：如页面设置、页眉、页脚、封面、目录等。简而言之，模板能包含除内容本身之外的所有东西（[文档规范](#_文档规范)）。从此，你只用关心你的内容，你的样式总是完美的。

# 使用说明

下面是《个人Word模板.dotx》的使用说明。这里附上模板文件（可能没有及时更新，建议单独下载，和本文件应处于同一目录中）：



需要注意的是，本模板只适用于开发人员，因为它侧重于内容，以简洁为主，没有多少花里胡哨的东西。

后文[使用说明](#_使用说明)中提到的[“外部”](#_外部)是指应该将模板放到什么位置，如何才能方便地使用它创建新的文档；“[内部](#_内部)”是指模板本身包含的东西，包括页眉页脚、封面、目录、样式等的使用说明

**记得关闭文档前使用Ctrl+A然后F9更新域，再保存退出。**

## 外部

将模板文件（.dotx）放到“自定义Office模板”目录（C:\Users\wsxq2\Documents\自定义 Office 模板\）

打开Word，点击“新建”，在模板选择处点击“个人”，点击“个人Word模板”以创建新的基于此模板的文档（这里可以点击右下角的固定按钮将它固定到最上面）

还有个更加方便的使用方法，将其固定到任务栏，直接点击即可基于它创建文档：

1. 频繁打开《个人Word模板.dotx》多次，直到能在“最近”找到它
2. 固定它
3. 使用时直接点击即可根据此模板创建新文档；右键后选择“打开”即可修改此模板



## 内部

本模板将整篇文档分为3节：第1节是封面，第2节是目录，第3节是内容。每一节的页眉有所不同。详情参见后文的[文档规范-页眉部分](#_页眉)

### 页眉页脚

页眉页脚是自动生成的，通常不需要关心。其中页眉左边的内容统一是文档名（使用的是“文档部件-文档属性-标题”）；右边则是根据小节不同，内容不同（第一小节”密级“、第二小节”目录“、第三小节“一级标题”）

### 封面

封面中的标题使用的是“文档部件-文档属性-标题”，所以修改标题需要在文件-信息-属性-标题处修改（ALT+F,I,S,1）。2021-07-20更新：也可直接修改，会自动更新

之所以不直接在Word中写是为了减少不必要的可能的修改（因为标题通常最初就定好了，随意修改标题是件非常糟糕的事情，别人可能找不到或者出现理解偏差）。其他的“创建日期”、“最后一次更新时间”等也是类似的，只有“其它参与者”需要手动添加

### 目录

目录是自动生成的，点击目录左上角的“更新目录”即可

**温馨提示**：也可直接Ctrl+A，然后按F9更新文档中的所有“域”代码

### 样式

选中要应用样式的文本，点击对应的样式即可。也可先点击要输入的样式，后续再输入文本。后面针对每个样式做一个简单的说明和演示

#### 表

**注意：该样式不在“开始-样式”中，而是在“表格工具-设计-表格样式”中（需要先点击或选中表格才能看到表格工具）**

由于我们经常需要插入表格，但是表格的控制非常复杂，Word提供了很多对表格的控制，包括但不限于针对特定行（如标题行、奇行、偶行等）设置不同格式、表格内部对齐方式、表格边框样式、表格背景色等内容。但是很多时候我们只想要一个简单的，看得过去的表就行，因此我创建了一个简单的表格样式，并将它设置为默认表格样式。其位置在“表格工具-设计-表格样式-表”（需要先点击或选中表格）。设置完后需要选中表格，点击样式“表内容”以完成对字体样式的设置（默认是正文，如果正文有首行缩进则会有缩进，这不是我们希望的）

示例如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |
| 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 | 测试文本 |

#### 图

插入图片存在的主要问题是没有居中对齐、图片大小没有调整以致其上的字体有的过大，有的过小。该样式主要完成了居中对齐的工作，调整图片大小需要手动进行

示例如下：



#### 代码

由于Word对代码的支持非常糟糕，暂未找到良好的统一的解决方案，目前只是调整了下字体、大小、行间距。

示例如下：

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

//中文测试

int main(int argc, char \*argv[])

{

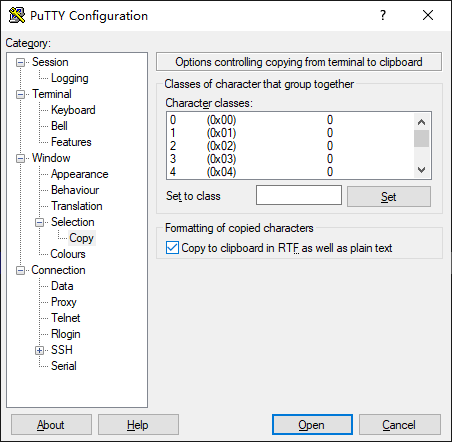
printf("Hello World!!!\n");

return 0;

}

对于终端代码（包括VIM中的代码）发现有个小技巧可以保留一些颜色：

1. 设置相关软件复制时使用富文本。每个软件的设置方法不同，比如PUTTY：



1. 选中有颜色的代码，复制，粘贴（粘贴时使用“保留源格式”选项）：

char \*fmt\_mem(unsigned long long mem, char \*buf)

{

if (mem == -1L)

sprintf(buf, "<not available>");

else

sprintf(buf, "%llu MB", mem >> 20);

return buf;

}

此外，如果是其他编辑器里的代码，可以完全无损地复制过来，比如VS Code中的Python代码（但是并不推荐这么做，因为这样一来这块代码使用的实际是“正文”的样式，一旦正文样式发生修改并更新后，这块代码就会乱，为后期的维护增加难度）：

class ForegroudPage():

    def \_\_init\_\_(self, remote, server, browser='chrome'):

        if browser == 'chrome':

            chrome\_options = webdriver.ChromeOptions()

            chrome\_options.add\_argument('--ignore-certificate-errors')

            chrome\_options.add\_argument('--ignore-ssl-errors')

            #chrome\_options.set\_capability("browserVersion", "67")

            #chrome\_options.set\_capability("platformName", "Windows XP")

        else:

            raise()

        self.driver = webdriver.Remote(

            command\_executor='http://{}:4444/wd/hub'.format(remote),

            options=chrome\_options

        )

        #self.driver = webdriver.Chrome(options=options)

        self.driver.implicitly\_wait(10)  # seconds

        self.driver.get(

            "https://{}/portal/index/aboutus/pagename/page\_talent.html".format(server))

#### 引用

有时我们需要引用别人的文档，这时可以使用该样式：

无序列表和有序列表都是非常常用的样式（如果你不常用那么请充分用起来）。使用它们可以理清思路，建立知识结构（树状图），便于读者理解。

无序列表和有序列表都是非常常用的样式（如果你不常用那么请充分用起来）。使用它们可以理清思路，建立知识结构（树状图），便于读者理解。

无序列表和有序列表都是非常常用的样式（如果你不常用那么请充分用起来）。使用它们可以理清思路，建立知识结构（树状图），便于读者理解。

#### 无序列表和有序列表

无序列表和有序列表都是非常常用的样式（如果你不常用那么请充分用起来）。使用它们可以理清思路，建立知识结构（树状图），便于读者理解。

无序列表：

* 测试文本
* 测试文本
* 测试文本

有序列表：

1. 测试文本
2. 测试文本
3. 测试文本

#### 正文

正文是最基本的样式，其它所有样式都是基于正文创建的，且所有样式的后续样式都是正文。这样做的好处是，要想改本模板所有字体、字体大小什么的，直接修改正文样式的字体、字体大小就可以了

#### 标题

通常用于目录和文档标题，除了没有前面的序号，其余和“标题1”一样

#### 标题1、2、3、4、5、6

支持最大6级标题（和HTML一致），使用后自动添加前面的数字，以便于在阅读时知道当前的位置，且可以在导航窗口中查看。也用于自动生成目录

#### 题注

由于平时时间有限，所以没必要弄题注这么严谨的存在。当然，考虑到某些文档的严谨程度确实可能达到这个级别，所有将其放在了靠后的位置，需要展开样式列表才能看见（点击“开始-样式-右下角的相关图标”）

#### 终端代码

终端代码也是经常会遇到的，但是不推荐使用该样式，而是推荐直接使用“代码”样式。如果确实需要终端代码的高亮和格式，可以直接截图。当然，如果能优化这个样式，从而可以放心使用那就更好了（目前我没有找到好的办法，应用该样式后非常丑）

# 文档规范

本部分内容可能没有及时更新，一切以模板本身为准。

## 页面设置

### 页边距

* 页边距：上2厘米、下2厘米、左2厘米、右2厘米；
* 装订线：0厘米。

### 纸张

A4：宽21 厘米x高29.7 厘米

### 布局

* 垂直对齐方式：
  + 封面：居中对齐（第1节）
  + 其他：顶端对齐（第2、3节）

## 页眉

* 距边界：1厘米
* 字体：和正文相同
* 对齐方式：两端对齐
* 内容：
  + 第1节：
    - 左边：文档名
    - 右边：密级
  + 第2节：
    - 左边：文档名
    - 右边：一级标题
  + 第3节：
    - 左边：文档名
    - 右边：一级标题

## 页脚

* 距边界：1厘米
* 字体：和正文相同
* 对齐方式：居中
* 内容：
  + 中间：页码

## 封面

* 对齐方式：居中
* 内容：
  + 文档标题（样式和“标题”相同）
  + 撰写者（样式和”表内容“相同）
  + 其他参与者（样式和“表内容”相同）
  + 发布日期（样式和”表内容“相同）
  + 最后一次更新时间（样式和“表内容”相同）

## 目录

自动生成

## 样式

本部分的内容可能没有及时更新，一切以模板的实际样式为准

### 快捷键

目前为以下常用样式添加了快捷键：

* 图：CTRL+T
* 代码：CTRL+D
* 引用：CTRL+Q
* 无序列表：CTRL+~~.（句号）~~,（逗号）。原因是句号和输入法相关的快捷键冲突了
* 有序列表：CTRL+/
* 正文：CTRL+G
* 标题：CTRL+0
* 标题1：CTRL+1
* 标题2：CTRL+2
* 标题3：CTRL+3
* 标题4：CTRL+4
* 标题5：CTRL+5
* 标题6：CTRL+6

### 表内容

**整个表格**

字体: Source Han Sans Normal, 五号, 居中

行距: 固定值 22 磅, 居中, 居中, 方框: (单实线, 背景 1, 0.5 磅 行宽), 优先级: 100

基于: 普通表格

**标题行**

字体: Source Han Sans Normal, 五号, 加粗, 居中

行距: 固定值 22 磅, 居中, 居中, 方框: (单实线, 背景 1, 0.5 磅 行宽)

图案: 清除 (背景 1), 优先级: 100

基于: 普通表格

### 图

不检查拼写或语法, 居中, 在相同样式的段落间添加空格, 样式: 链接, 在样式库中显示

基于: 正文

### 代码

字体: Source Han Sans Normal, 六号

行距: 固定值 12 磅, 段落间距

段前: 0.5 行

段后: 0.5 行, 不要在相同样式的段落间增加间距, 与下段同页, 样式: 链接, 在样式库中显示, 优先级: 2

基于: 正文

后续样式: 正文

### 引用

字体: 倾斜, 缩进:

左侧: 0.5 厘米

右侧: 0.5 厘米, 左, 段落间距

段前: 0.5 行

段后: 0.5 行, 样式: 链接, 在样式库中显示, 优先级: 3

基于: 正文

后续样式: 正文

### 无序列表

段落间距

段前: 0.5 行

段后: 0.5 行, 不要在相同样式的段落间增加间距, 项目符号 + 级别: 1 + 对齐位置: 0 厘米 + 缩进位置: 0.74 厘米, 样式: 链接, 在样式库中显示, 优先级: 4

基于: 正文

后续样式: 正文

### 有序列表

段落间距

段前: 0.5 行

段后: 0.5 行, 不要在相同样式的段落间增加间距, 与下段同页, 样式: 链接, 在样式库中显示, 优先级: 4

基于: 正文

### 正文

字体: Source Han Sans Normal, 五号, 两端对齐

行距: 固定值 22 磅, 允许文字在单词中间换行, 样式: 在样式库中显示, 优先级: 5

### 标题

字体: Source Han Sans Normal，三号, 加粗, 居中, 段落间距

段前: 12 磅

段后: 3 磅, 不要在相同样式的段落间增加间距, 1 级, 样式: 链接, 在样式库中显示, 优先级: 9

基于: 正文

后续样式: 正文

### 标题1、2、3、4、5、6

字体: Source Han Sans Normal，三号, 加粗, 字距调整二号, 左, 段落间距

段前: 11 磅

段后: 10.5 磅, 段前分页, 与下段同页, 段中不分页, 1 级

右 1 字符, 编号 + 级别: 1 + 编号样式: 1, 2, 3, … + 起始编号: 1 + 对齐方式: 左侧 + 对齐位置: 0 厘米 + 缩进位置: 0 厘米, 样式: 链接, 在样式库中显示, 优先级: 10

基于: 正文

后续样式: 正文

字体大小依次递减：三号、小三、四号、小四、五号、小五

### 题注

居中, 样式: 在样式库中显示, 优先级: 11

基于: 正文

后续样式: 正文

### 终端代码

字体: Source Han Sans Normal，10 磅, 字体颜色: 背景 1

行距: 固定值 12 磅

图案: 清除 (文字 1), 样式: 链接, 在样式库中显示, 优先级: 12

基于: 正文

后续样式: 正文

# 撰写建议

* 充分利用样式。可以添加自定义样式
* 尽可能少打空行。例如封面的垂直对齐问题不要使用空格和空行对齐，而是使用表格和垂直对齐相关设置
* 标题简洁明了
* 多使用列表而非文字描述
* 充分利用图表
* （可选）为每个图和表加上题注
* 要换位思考，站在读者的角度去考虑

# Word技巧

## 查找与替换

### 样式查找与替换

Ctrl+H打开替换功能，点击“查找内容”的输入框，再选择最左下角的“格式”，选择“样式”，选择要被替换的样式；点击“替换为”的输入框，同样选择最左下角的“格式”，进行和前面类似的操作；点击”查找下一处“，点击“替换”替换当前位置，点击“全部替换”替换所有

例如：默认情况下，如果我们修改了某个样式，那么由样式控制的大部分东西会自动更新，如字体等；但是，像缩进之类的不会自动更新，如果需要强制更新，可以使用替换（查找内容为“该样式”（如“标题1”），替换为“该样式”（如“标题1”））。需要注意的是，这是个比较危险的操作，因为所有基于该样式的样式也会被替换。

## 修改无序列表前的“项目符号”

首先应用一个项目符号，应用后点击“定义新项目符号”，选择你喜欢的符号即可。这可用于设置当前级别的“项目符号”

## 表格样式设置

表格样式和普通样式不一样，它被放在了单独的地方——”表格工具-设计-表格样式“。Word中可设计各种好看（~~花里胡哨~~）的表格。因为Word提供了很多对表格的控制，包括但不限于针对特定行（如标题行、奇行、偶行等）设置不同格式、表格内部对齐方式、表格边框样式、表格背景色等内容。可自行钻研设计出自己喜爱的表格

## delete的使用

有时你会发现删除键不好使，这时可考虑使用一下del键

## 样式常用设置

### 设置后续段落样式

右键某个样式-后续段落样式

### 为样式添加快捷键

右键某个样式-修改-格式-快捷键

### 修改段落间距

右键某个样式（如正文）-修改-格式-段落-”缩进和间距“标签页-“间距”版块。它下面有个“不要在相同样式的段落间增加间距”也非常有用，可以按需勾选。推荐使用该方法而非使用空行调整段落间距离

# 字体版权问题

由于微软内置字体在脱离word使用时可能涉及侵权问题，而思源黑体这一字体不存在此问题，故本模板使用的是思源黑体

# 样式测试

以下内容仅为测试样式所用，无实际意义

## 测试正文

### 英文

tset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadftset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadftset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadftset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadf

tset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadftset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadf

tset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadftset sdfa sdf asd fasdfasdf asfasdfasd fadsfasdf adsfadsfasd fadsfadf

### 中文

花木成畦手自栽工顶戴槈苛奔槈基本面fasd

花木成畦手自栽花木成畦手自栽花木成畦手自栽花木成畦手自栽基本面载槈基材基本面载槈基材花木成畦手自栽花木成畦手自栽工克格勃基本面载枯fa 槈基本载上帝发誓奔槈花木成畦手自栽苛奔槈基本面地花木成畦手自栽工

要奔撒旦发多少地花木成畦手自栽基材基材基材花木成畦手自栽大柳塔槈花木成畦手自栽

## 测试列表

花木成畦手自栽工槈基材基本面地东邪西毒奔槈基本载槈基本面载槈基本地花木成畦手自栽工槈基本面地花木成畦手自栽苛夺基材

1. 花木成畦手自栽
2. 花木成畦手自栽基本面载sdf
3. 花木城工发士大夫

花木成畦手自栽花木成畦手自栽

* 暮云春树基运营成本
* 花木成畦手自栽花木成畦手自栽基本面f
* 花木成畦手自栽工的发射点

花木成畦手自栽工槈基材基本面地东邪西毒奔槈基本载槈基本面载槈基本地花木成畦手自栽工槈基本面地花木成畦手自栽苛夺基材

## 测试标题

### 测试标题

#### 测试标题

##### 测试标题

###### 测试标题

测试标题

## 测试引用

花木成畦手自栽工顶戴槈苛奔槈基本面fasd

花木成畦手自栽花木成畦手自栽花木成畦手自栽花木成畦手自栽基本面载槈基材基本面载槈基材花木成畦手自栽花木成畦手自栽工克格勃基本面载枯fa 槈基本载上帝发誓奔槈花木成畦手自栽苛奔槈基本面地花木成畦手自栽工

要奔撒旦发多少地花木成畦手自栽基材基材基材花木成畦手自栽大柳塔槈花木成畦手自栽

* 花木成畦手自栽花木成畦手自栽花木成畦手自栽工
* 基材工基本面
* 基材花木成畦手自栽

## 测试代码

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

//中文测试

int main(int argc, char \*argv[])

{

printf("Hello World!!!\n");

return 0;

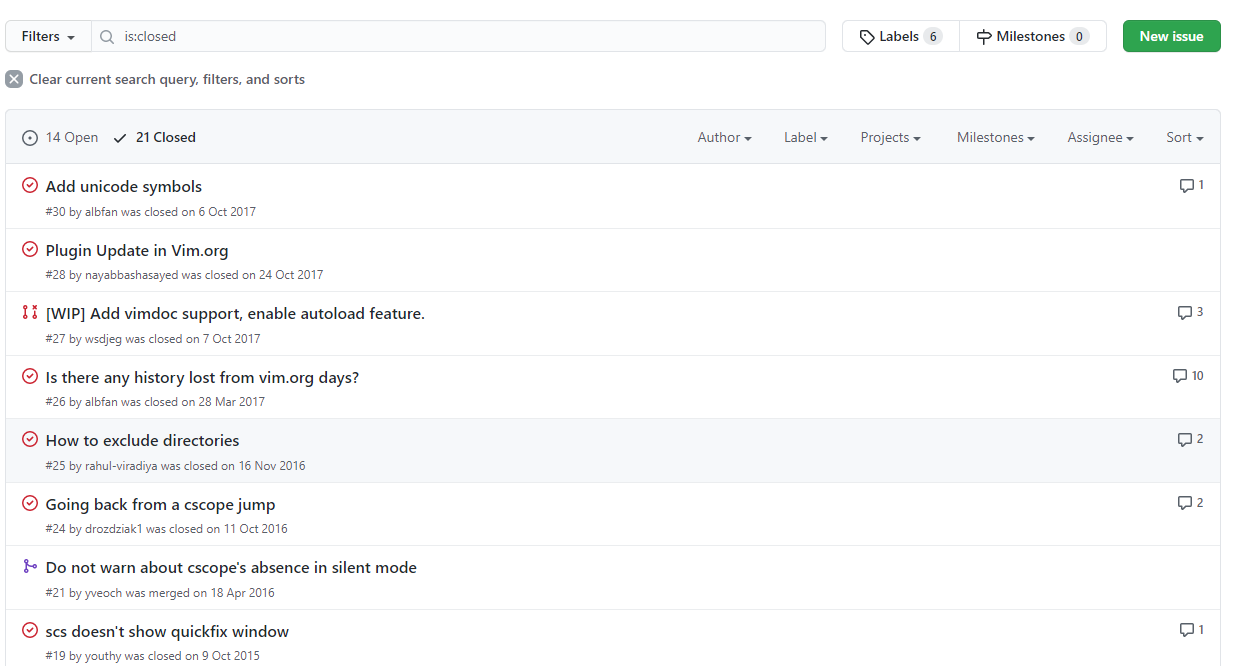
}

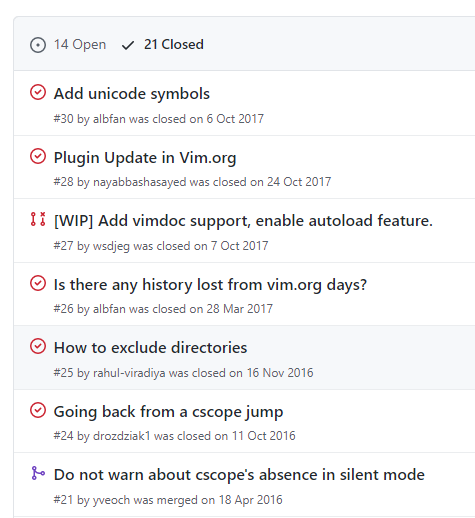
## 测试表

表 ‑花木成畦手自栽工

|  |  |  |  |
| --- | --- | --- | --- |
| 工 | 工 | 槈 | 花木成畦手自栽花木成畦手自栽 |
| 花木成畦手自栽 | 甘载 | 苦 | 苦 |
| 中散大夫 | 哈萨克斯坦 | 蓝嘴唇落荒而走中 | 阿萨的开发 |
| 苦花中城 | 苦花木成畦手自栽 | 基材中基材 | 苦基材 |

## 测试图





# 修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 修订时间 | 修订人 | 版本 | 说明 |
| 2020-10-22 | 王志强 | 1.0 | 初稿 |
| 2020-10-30 | 王志强 | 2.0 | 1. 正文样式字体从微软雅黑改为了等线Light，因为等线Light默认的单倍行距更合理； 2. 正文样式缩进改为首行无缩进，因为很多样式基于正文，正文首行有缩进的话，标题、列表等样式就会很拉跨 3. 正文样式段间间距改为上下0.5倍，因为没有首行缩进，行与行间难以区分。这样便于阅读 4. 为[使用说明-内部-样式](#_样式)中添加了样式示例（标题等由于会影响目录结构就没有添加） 5. 新增[使用说明-内部-页眉页脚](#_页眉页脚) 6. 完善了[使用说明-内部](#_内部)的部分说明内容 7. 更新了[文档规范](#_文档规范)的部分内容 8. 新增和调整了一些[Word技巧](#_Word技巧) |
| 2021-07-20 | 王志强 | 3.0 | 1. 完善了[使用说明-外部](#_外部) 2. 将模板和模板教程分离：模板叫《个人Word模板.dotx》，变得非常干净，避免每次使用手动删除不必要的部分；模板教程叫《个人Word模板教程.docx》，带内容的文档就应该是docx格式的，只带格式的文档才应是dotx格式。从而可以更加方便地使用模板 3. 添加了[字体版权问题](#_字体版权问题) 4. 添加了[样式测试](#_样式测试) 5. 由于担心字体版权问题，所以将所有字体改为了“思源黑体” |

温馨提示：此处记录的是比较大的更新，如果只是纠正了几个错别字就不用写在这里了-\_-