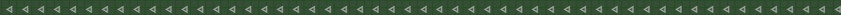


# Beamer 演示学习笔记



ZOHOOO@YEAH.NET



2016 年 5 月 26 日

1. 基本使用

2. 主题选用

3. 主题定制

4. 各种设置

1. 基本使用

2. 主题选用

3. 主题定制

4. 各种设置

- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 简要介绍

Beamer 是一个用于制作演示文稿的 LaTeX 文档类，由 Till Tantau 编写。

## 简要介绍

Beamer 是一个用于制作演示文稿的 LaTeX 文档类，由 Till Tantau 编写。相对于其它同类工具，Beamer 有如下这些优点：

## 简要介绍

Beamer 是一个用于制作演示文稿的 LaTeX 文档类，由 Till Tantau 编写。相对于其它同类工具，Beamer 有如下这些优点：

- ▶ 功能强大，各种侧栏、顶栏、底栏，导航栏一应俱全。
- ▶ 定制灵活，可以单独改变任何元素的结构，颜色和字体。
- ▶ 效果多样，支持各种各样的过渡效果并可以精确控制。
- ▶ 使用方便，可以用 latex, pdflatex 及 xelatex 编译。

## 简要介绍

Beamer 是一个用于制作演示文稿的 LaTeX 文档类，由 Till Tantau 编写。相对于其它同类工具，Beamer 有如下这些优点：

- ▶ 功能强大，各种侧栏、顶栏、底栏，导航栏一应俱全。
- ▶ 定制灵活，可以单独改变任何元素的结构，颜色和字体。
- ▶ 效果多样，支持各种各样的过渡效果并可以精确控制。
- ▶ 使用方便，可以用 latex, pdflatex 及 xelatex 编译。

由于这些优点，Beamer 出现之后，很快成为最流行的 LaTeX 演示制作工具。



- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 英文文档

最简单的 Beamer 英文文档如下：

```
\documentclass{beamer}
\begin{document}
  \begin{frame}
    Hello Beamer!
  \end{frame}
\end{document}
```

在这里约定一下，我们用“演示文稿”来表示整个 Beamer 文档，用“幻灯片”来表示 Beamer 演示的其中一张，即上面代码中的 frame 环境。

## 英文文档

最简单的 Beamer 英文文档如下：

```
\documentclass{beamer}
\begin{document}
  \begin{frame}
    Hello Beamer!
  \end{frame}
\end{document}
```

在这里约定一下，我们用“演示文稿”来表示整个 Beamer 文档，用“幻灯片”来表示 Beamer 演示的其中一张，即上面代码中的 `frame` 环境。从这个例子可以看出，Beamer 中每张幻灯片的内容都是放置在一个 `frame` 环境里面的。

## 中文文档

如果要使用中文，可以用 `ctex` 宏包，例如：

```
\documentclass{beamer}
\usepackage[UTF8]{ctex}
\begin{document}
  \begin{frame}
    你好 Beamer!
  \end{frame}
\end{document}
```

对于中文文档，建议用 UTF8 编码，然后用 `xelatex` 程序编译。另外，可以在载入 `ctex` 宏包时加上 `noindent` 选项，以取消段落的缩进。

## 中文文档

在 CTeX 套装 2.9 自带的 WinEdt 编辑器里面，对中文默认不是 UTF8 编码的。我们可以按照下列步骤建立 UTF8 编码的中文文档：

1. 新建一个空白文件；
2. 在第一行写上这一句：  

```
% -*- coding: utf-8 -*-
```
3. 保存，关闭，再打开文件。

## 幻灯片标题

在每张幻灯片中，可以添加标题和副标题，例如：

```
\begin{frame}{幻灯片标题}{我是一个副标题}  
Hello Beamer!  
\end{frame}
```

## 幻灯片标题

在每张幻灯片中，可以添加标题和副标题，例如：

```
\begin{frame}{幻灯片标题}{我是一个副标题}  
Hello Beamer!  
\end{frame}
```

或者也可以分开来写，如下：

```
\begin{frame}  
\frametitle{幻灯片标题}  
\framesubtitle{我是一个副标题}  
Hello Beamer!  
\end{frame}
```

## 竖直位置

在 Beamer 的每张幻灯片中，正文内容（不包括幻灯片标题）默认都是竖直居中的。这是一种很好的功能，但也许你就喜欢正文竖直居上。



## 竖直位置

在 Beamer 的每张幻灯片中，正文内容（不包括幻灯片标题）默认都是竖直居中的。这是一种很好的功能，但也许你就喜欢正文竖直居上。没问题，只要在文档类中加上 `t` 选项就可以了，如下：

```
\documentclass[t]{beamer}
```

## 竖直位置

在 Beamer 的每张幻灯片中，正文内容（不包括幻灯片标题）默认都是竖直居中的。这是一种很好的功能，但也许你就喜欢正文竖直居上。没问题，只要在文档类中加上 `t` 选项就可以了，如下：

```
\documentclass[t]{beamer}
```

如果你只需要让某张幻灯片的正文内容竖直居上、居中或者居下，可以在 `frame` 环境中分别加上 `t`、`c` 或者 `b` 选项。例如：

```
\begin{frame}[b]  
Hello Beamer from the Bottom!  
\end{frame}
```

- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 标题页面

在幻灯片中用 `\titlepage` 命令可以生成标题页，一般这是第一张幻灯片。例如：

```
\title{Beamer演示学习笔记}  
\author{zoho@bbs.ctex.org}  
\date{2011年12月6日}  
\begin{frame}[plain]  
\titlepage  
\end{frame}
```

其中的 `plain` 选项表示不显示顶栏侧栏底栏等外部元素。

## 文档结构

在 Beamer 文档中，可以用 `\part`、`\section`、`\subsection` 等结构命令，但是不能用 `\chapter`。例如：

## 文档结构

在 Beamer 文档中，可以用 `\part`、`\section`、`\subsection` 等结构命令，但是不能用 `\chapter`。例如：

```
\documentclass{beamer}
\begin{document}
\section{One Section}
\begin{frame}First Frame\end{frame}
\begin{frame}Second Frame\end{frame}
\section{The Other Section}
\begin{frame}Third Frame\end{frame}
\end{document}
```

## 文档结构

在 Beamer 文档中，可以用 `\part`、`\section`、`\subsection` 等结构命令，但是不能用 `\chapter`。例如：

```
\documentclass{beamer}
\begin{document}
\section{One Section}
\begin{frame}First Frame\end{frame}
\begin{frame}Second Frame\end{frame}
\section{The Other Section}
\begin{frame}Third Frame\end{frame}
\end{document}
```

注意这些结构命令必须放置在各个 `frame` 环境之间，放在里面会有负面效果。

## 目录页面

类似于标题页面，我们可以在幻灯片中用 `\tableofcontents` 命令生成目录页。例如：



## 目录页面

类似于标题页面，我们可以在幻灯片中用 `\tableofcontents` 命令生成目录页。例如：

```
\begin{frame}  
\tableofcontents[hideallsubsections]  
\end{frame}
```

其中 `hideallsubsections` 选项表示不显示小节标题。

- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 有序列表

在 Beamer 中可以如常使用列表环境，例如：

## 有序列表

在 Beamer 中可以如常使用列表环境，例如：

```
\begin{enumerate}  
\item 我是第一项  
\item 我是第二项  
\item 我是第三项  
\end{enumerate}
```

得到下面的结果：

1. 我是第一项
2. 我是第二项
3. 我是第三项

## 无序列表

再看看 Beamer 中的无序列表环境，例如：

## 无序列表

再看看 Beamer 中的无序列表环境，例如：

```
\begin{itemize}
\item 红色 -- red
\item 绿色 -- green
\item 蓝色 -- blue
\end{itemize}
```

得到下面的结果：

- ▶ 红色 – red
- ▶ 绿色 – green
- ▶ 蓝色 – blue

## 描述列表

再看看 Beamer 中的描述列表环境，例如：

## 描述列表

再看看 Beamer 中的描述列表环境，例如：

```
\begin{description}
\item[红色] 热情、活泼、温暖、幸福
\item[绿色] 新鲜、平静、安逸、柔和
\item[蓝色] 深远、永恒、沉静、寒冷
\end{description}
```

得到下面的结果：

红色 热情、活泼、温暖、幸福

绿色 新鲜、平静、安逸、柔和

蓝色 深远、永恒、沉静、寒冷



- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 区块环境

Beamer 里面定义了一个区块环境，可以用于显示重要的内容。例如下面的代码

```
\begin{block}{重要内容}  
2012 年 12 月 21 日是世界末日。  
\end{block}
```

得到的结果为

**重要内容**

2012 年 12 月 21 日是世界末日。

## 提醒环境

与区块环境类似地还有一个提醒环境和例子环境。先看看提醒环境。例如下面的代码

```
\begin{alertblock}{重要提醒}  
2012 年 12 月 21 日是世界末日。  
\end{alertblock}
```

得到的结果为

**重要提醒**

2012 年 12 月 21 日是世界末日。

## 例子环境

再来看看例子环境。例如下面的代码

```
\begin{exampleblock}{重要例子}  
2012 年 12 月 21 日是世界末日。  
\end{exampleblock}
```

得到的结果为

**重要例子**

2012 年 12 月 21 日是世界末日。

- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 定理环境

Beamer 中也定义了各种定理环境，而且默认是用区块环境的样式来显示的。例如下面的代码

```
\begin{theorem}
微积分基本公式： $\int_a^b f(x)\mathrm{d}x=F(b)-F(a)$ 。
\end{theorem}
```

得到如下的结果：

### 定理

微积分基本公式： $\int_a^b f(x)dx = F(b) - F(a)$ 。

## 定理环境

各种可用的定理类环境有这些：theorem、corollary、definition、definitions、fact、example 和 examples。

在 Beamer 中定理名默认是英文显示的，如果要改为中文显示，可以在文档开头用类似下面的代码：

```
\documentclass[notheorems]{beamer}
\usepackage[UTF8,noindent]{ctex}
\newtheorem{theorem}{定理}
\newtheorem{example}[theorem]{例子}
\newtheorem*{theorem*}{定理}
\newtheorem*{example*}{例子}
```

其中的 notheorems 选项表示不使用默认的定理类环境。

## 证明环境

Beamer 中也定义了证明环境。例如下面的代码

```
\begin{proof}
```

令  $g(x) = e^x - x - 1$ 。则当  $x > 1$  时，有  $g'(x) = e^x - 1 > 0$ ，  
因此  $g(x) > g(1) = 0$ 。即有  $x > 1$  时  $e^x > 1 + x$ 。

```
\end{proof}
```

得到如下的结果：

**证明.**

令  $g(x) = e^x - x - 1$ 。则当  $x > 1$  时，有  $g'(x) = e^x - 1 > 0$ ，  
因此  $g(x) > g(1) = 0$ 。即有  $x > 1$  时  $e^x > 1 + x$ 。  $\square$



## 证明环境

类似于定理类环境，Beamer 的证明环境中默认也用英文的“Proof”。

下面的代码

```
\renewcommand{\proofname}{证明}
```

可以将它改为中文的“证明”二字。

- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 分栏环境

Beamer 中提供了 `columns` 环境，用于分栏显示。例如：

```
\begin{columns}
```

```
\column{.5\textwidth}
```

左栏的内容，占用一半宽度。

```
\column{.5\textwidth}
```

右栏的内容，占用一半宽度。

```
\end{columns}
```

左栏的内容，占用一半宽度。

右栏的内容，占用一半宽度。

## 分栏对齐

Beamer 的 `columns` 环境，将会自动在各栏之间留出空隙。

## 分栏对齐

Beamer 的 `columns` 环境，将会自动在各栏之间留出空隙。

在前面的例子中，各栏总宽度正好等于内容区宽度，加上空隙后就超出内容区宽度。这样，分栏环境就和正文参差不齐。

## 分栏对齐

Beamer 的 `columns` 环境，将会自动在各栏之间留出空隙。

在前面的例子中，各栏总宽度正好等于内容区宽度，加上空隙后就超出内容区宽度。这样，分栏环境就和正文参差不齐。

解决的方法是指定 `onlytextwidth` 选项。

```
\begin{columns}[onlytextwidth]
```

```
\column{.5\textwidth}
```

左栏的内容，占用一半宽度。

```
\column{.5\textwidth}
```

右栏的内容，占用一半宽度。

```
\end{columns}
```

左栏的内容，占用一半宽度。      右栏的内容，占用一半宽度。

- 1.1 简要介绍
- 1.2 基本内容
- 1.3 目录小节
- 1.4 列表环境
- 1.5 区块环境
- 1.6 数学环境
- 1.7 分栏显示
- 1.8 创建覆盖

## 覆盖命令

利用 Beamer 中的覆盖 (overlay) 命令, 我们可以逐步显示幻灯片的内容。常用的覆盖命令有如下这些:

- ▶ `\pause`
- ▶ `\onslide`
- ▶ `\onslide+`
- ▶ `\onslide*`
- ▶ `\uncover`
- ▶ `\visible`
- ▶ `\only`

接下来我们将逐个介绍它们的用法以及区别。



## 覆盖命令之一

最简单的覆盖命令是 `\pause`。例如：

一个 `\pause` 两个 `\pause` 三个

得到的结果是：

## 覆盖命令之一

最简单的覆盖命令是 `\pause`。例如：

一个 `\pause` 两个 `\pause` 三个

得到的结果是：一个

## 覆盖命令之一

最简单的覆盖命令是 `\pause`。例如：

一个 `\pause` 两个 `\pause` 三个

得到的结果是：一个 两个

## 覆盖命令之一

最简单的覆盖命令是 `\pause`。例如：

一个 `\pause` 两个 `\pause` 三个

得到的结果是：一个 两个 三个。

## 覆盖命令之一

最简单的覆盖命令是 `\pause`。例如：

一个 `\pause` 两个 `\pause` 三个

得到的结果是：一个 两个 三个。

这些 `\pause` 命令将把一帧幻灯片分为多页显示。

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：



## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是： 一个



## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：

一个	三个
----	----

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：一个 两个 三个

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：一个 两个 三个。

这些 `\onslide` 命令同样把一帧幻灯片分为多页显示。

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：一个 两个 三个。

这些 `\onslide` 命令同样把一帧幻灯片分为多页显示。但是用 `\onslide` 我们需要指定在哪几页显示。

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是： 一个 两个 三个。

这些 `\onslide` 命令同样把一帧幻灯片分为多页显示。但是用 `\onslide` 我们需要指定在哪几页显示。例如：

- ▶ `<1->` 表示从第 1 页开始显示；
- ▶ `<2-4>` 表示在第 2 到 4 页显示；
- ▶ `<-5>` 表示在前 5 页显示；
- ▶ `<1,3-5>` 表示在第 1 和第 3 到 5 页显示。

## 覆盖命令之二

比 `\pause` 高级点的覆盖命令是 `\onslide`。例如：

`\onslide<1->` 一个 `\onslide<3->` 两个 `\onslide<2->` 三个

得到的结果是：一个 两个 三个。

这些 `\onslide` 命令同样把一帧幻灯片分为多页显示。但是用 `\onslide` 我们需要指定在哪几页显示。例如：

- ▶ `<1->` 表示从第 1 页开始显示；
- ▶ `<2-4>` 表示在第 2 到 4 页显示；
- ▶ `<-5>` 表示在前 5 页显示；
- ▶ `<1,3-5>` 表示在第 1 和第 3 到 5 页显示。

在同一个 frame 中，最好不要混合使用 `\pause` 和 `\onslide`，否则覆盖次序可能不是你想要的。

## 覆盖命令之二

如果我们先设置好 `\setbeamercovered` 命令如下：

```
\setbeamercovered{transparent}
```

就可以让 `\pause` 或者 `\onslide` 设定的未显示内容用半透明显示。

此时

```
\onslide<1-> 一个 \onslide<3-> 两个 \onslide<2-> 三个
```

得到的结果是：一个 两个 三个。

## 覆盖命令之二

如果我们先设置好 `\setbeamercovered` 命令如下：

```
\setbeamercovered{transparent}
```

就可以让 `\pause` 或者 `\onslide` 设定的未显示内容用半透明显示。

此时

```
\onslide<1-> 一个 \onslide<3-> 两个 \onslide<2-> 三个
```

得到的结果是：一个 两个 三个。



## 覆盖命令之二

如果我们先设置好 `\setbeamercovered` 命令如下：

```
\setbeamercovered{transparent}
```

就可以让 `\pause` 或者 `\onslide` 设定的未显示内容用半透明显示。

此时

```
\onslide<1-> 一个 \onslide<3-> 两个 \onslide<2-> 三个
```

得到的结果是：一个 两个 三个。

## 覆盖命令之二

如果我们先设置好 `\setbeamercovered` 命令如下：

```
\setbeamercovered{transparent}
```

就可以让 `\pause` 或者 `\onslide` 设定的未显示内容用半透明显示。

此时

```
\onslide<1-> 一个 \onslide<3-> 两个 \onslide<2-> 三个
```

得到的结果是：一个 两个 三个。

## 覆盖命令之二

这里用到的 `\setbeamercovered` 命令，有如下这些选项：

- ▶ `invisible`：默认值，不显示后面页面的内容
- ▶ `transparent`：以半透明样式显示后面页面的内容
- ▶ `dynamic`：同上，但后面页面的内容透明度会逐步变化
- ▶ `highly dynamic`：同上，但效果更明显

用 `dynamic` 的效果见这里：

一个 两个 三个 四个 五个 六个

## 覆盖命令之二

这里用到的 `\setbeamercovered` 命令，有如下这些选项：

- ▶ `invisible`：默认值，不显示后面页面的内容
- ▶ `transparent`：以半透明样式显示后面页面的内容
- ▶ `dynamic`：同上，但后面页面的内容透明度会逐步变化
- ▶ `highly dynamic`：同上，但效果更明显

用 `dynamic` 的效果见这里：一个 两个 三个 四个 五个 六个

## 覆盖命令之二

这里用到的 `\setbeamercovered` 命令，有如下这些选项：

- ▶ `invisible`：默认值，不显示后面页面的内容
- ▶ `transparent`：以半透明样式显示后面页面的内容
- ▶ `dynamic`：同上，但后面页面的内容透明度会逐步变化
- ▶ `highly dynamic`：同上，但效果更明显

用 `dynamic` 的效果见这里：一个 两个 三个 四个 五个 六个

## 覆盖命令之二

这里用到的 `\setbeamercovered` 命令，有如下这些选项：

- ▶ `invisible`：默认值，不显示后面页面的内容
- ▶ `transparent`：以半透明样式显示后面页面的内容
- ▶ `dynamic`：同上，但后面页面的内容透明度会逐步变化
- ▶ `highly dynamic`：同上，但效果更明显

用 `dynamic` 的效果见这里：一个 两个 三个 四个 五个 六个

## 覆盖命令之二

这里用到的 `\setbeamercovered` 命令，有如下这些选项：

- ▶ `invisible`：默认值，不显示后面页面的内容
- ▶ `transparent`：以半透明样式显示后面页面的内容
- ▶ `dynamic`：同上，但后面页面的内容透明度会逐步变化
- ▶ `highly dynamic`：同上，但效果更明显

用 `dynamic` 的效果见这里：一个 两个 三个 四个 五个 六个

## 覆盖命令之二

这里用到的 `\setbeamercovered` 命令，有如下这些选项：

- ▶ `invisible`：默认值，不显示后面页面的内容
- ▶ `transparent`：以半透明样式显示后面页面的内容
- ▶ `dynamic`：同上，但后面页面的内容透明度会逐步变化
- ▶ `highly dynamic`：同上，但效果更明显

用 `dynamic` 的效果见这里：一个 两个 三个 四个 五个 六个



## 覆盖命令之三

与 `\onslide` 类似的还有 `\onslide+` 命令。这两者的区别在于 `\onslide+` 不受 `\setbeamercovered` 命令影响。用 `\onslide+` 设定的未显示内容，永远不可能用半透明显示。此时

`\onslide+<1->`一个`\onslide+<3->`两个`\onslide+<2->`三个

得到的结果是：。

## 覆盖命令之三

与 `\onslide` 类似的还有 `\onslide+` 命令。这两者的区别在于 `\onslide+` 不受 `\setbeamercovered` 命令影响。用 `\onslide+` 设定的未显示内容，永远不可能用半透明显示。此时

`\onslide+<1->`一个`\onslide+<3->`两个`\onslide+<2->`三个

得到的结果是：。

## 覆盖命令之三

与 `\onslide` 类似的还有 `\onslide+` 命令。这两者的区别在于 `\onslide+` 不受 `\setbeamercovered` 命令影响。用 `\onslide+` 设定的未显示内容，永远不可能用半透明显示。此时

`\onslide+<1->`一个 `\onslide+<3->`两个 `\onslide+<2->`三个

得到的结果是：

一个	三个
----	----

。

## 覆盖命令之三

与 `\onslide` 类似的还有 `\onslide+` 命令。这两者的区别在于 `\onslide+` 不受 `\setbeamercovered` 命令影响。用 `\onslide+` 设定的未显示内容，永远不可能用半透明显示。此时

`\onslide+<1->`一个`\onslide+<3->`两个`\onslide+<2->`三个

得到的结果是：一个 两个 三个。

## 覆盖命令之四

与 `\onslide` 和 `\onslide+` 类似的还有 `\onslide*` 命令。用 `\onslide*` 设定的未显示内容，在幻灯片上不占用空间。此时

```
\onslide*<1->{壹}\onslide*<3->{贰}\onslide*<2->{叁}
```

得到的结果是：□。

## 覆盖命令之四

与 `\onslide` 和 `\onslide+` 类似的还有 `\onslide*` 命令。用 `\onslide*` 设定的未显示内容，在幻灯片上不占用空间。此时

```
\onslide*<1->{壹}\onslide*<3->{贰}\onslide*<2->{叁}
```

得到的结果是： 壹。

## 覆盖命令之四

与 `\onslide` 和 `\onslide+` 类似的还有 `\onslide*` 命令。用 `\onslide*` 设定的未显示内容，在幻灯片上不占用空间。此时

```
\onslide*<1->{壹}\onslide*<3->{贰}\onslide*<2->{叁}
```

得到的结果是：壹叁。

## 覆盖命令之四

与 `\onslide` 和 `\onslide+` 类似的还有 `\onslide*` 命令。用 `\onslide*` 设定的未显示内容，在幻灯片上不占用空间。此时

```
\onslide*<1->{壹}\onslide*<3->{贰}\onslide*<2->{叁}
```

得到的结果是：壹贰叁。



## 覆盖命令之四

与 `\onslide` 和 `\onslide+` 类似的还有 `\onslide*` 命令。用 `\onslide*` 设定的未显示内容，在幻灯片上不占用空间。此时

```
\onslide*<1->{壹}\onslide*<3->{贰}\onslide*<2->{叁}
```

得到的结果是：壹贰叁。

### 注意

用 `\onslide*` 命令设定的内容**必须**放在分组括号中。用 `\onslide` 和 `\onslide+` 设定的内容虽然可以不放在分组括号中，但建议最好也放在括号中。

## 覆盖命令之五

与 `\onslide`、`\onslide+` 和 `\onslide*` 命令对应的还有另外三个覆盖命令：`\uncover`、`\visible` 和 `\only`。

1. `\uncover<1-2>{text}` 等同于 `\onslide<1-2>{text}`
2. `\visible<1-2>{text}` 等同于 `\onslide+<1-2>{text}`
3. `\only<1-2>{text}` 等同于 `\onslide*<1-2>{text}`

### 注意

使用覆盖命令 `\uncover`、`\visible` 和 `\only` 时，都**必须**将内容放在分组括号中。

1. 基本使用

2. 主题选用

3. 主题定制

4. 各种设置

2.1 整体主题

2.2 细分主题

## 整体主题

Beamer 的整体主题包含了结构、颜色、字体各方面的设置。我们可以用命令 `\usetheme{主题名}` 来选择整体主题。

## 整体主题

Beamer 的整体主题包含了结构、颜色、字体各方面的设置。我们可以用命令 `\usetheme{主题名}` 来选择整体主题。其中主题名有如下这些选择：

**无导航栏** default、boxes、Bergen、Pittsburgh 和 Rochester。

**带顶栏** Antibes、Darmstadt、Frankfurt、JuanLesPins、Montpellier 和 Singapore。

**带底栏** Boadilla 和 Madrid。

**带顶栏底栏** AnnArbor、Berlin、CambridgeUS、Copenhagen、Dresden、Ilmenau、Luebeck、Malmoe、Szeged 和 Warsaw。

**带侧栏** Berkeley、Goettingen、Hannover、Marburg 和 PaloAlto。

2.1 整体主题

2.2 细分主题

## 细分主题

Beamer 的每个演示主题实际上都是由外部主题、内部主题、颜色主题和字体主题这四种细分主题组合而成的。如果要对演示主题作更加细致地选择，可以按照下面这四种细分主题自由组合：

1. 外部主题，用 `\usebeameroutertheme` 命令；
2. 内部主题，用 `\usebeamerinnertheme` 命令；
3. 颜色主题，用 `\usebeamercolortheme` 命令；
4. 字体主题，用 `\usebeamerfonttheme` 命令。

你可以通过这四种细分主题的选择得到一个新的整体主题。



## 细分主题 1 — 外部主题

外部主题设定演示文稿是否有顶栏、底栏和侧栏，以及它们的结构，可以用 `\useoutertheme{主题名}` 来选择，其中主题名有如下这些选择：

- ▶ default
- ▶ infolines
- ▶ miniframes
- ▶ sidebar
- ▶ smoothbars
- ▶ split
- ▶ shadow
- ▶ tree
- ▶ smoothtree

## 细分主题 2 — 内部主题

内部主题设定演示文稿正文内容（例如标题、列表、定理等）的样式，可以用 `\useinnertheme{主题名}` 来选择，其中主题名有如下这些选择：

- ▶ default
- ▶ circles
- ▶ rectangles
- ▶ rounded

## 细分主题 3 — 颜色主题

颜色主题设定演示文稿的各部分各结构各元素的配色，可以用 `\usecolortheme{主题名}` 来选择，其中主题名有这些选择：

**基本颜色** default、sidebartab、structure；

**完整颜色** albatross（信天翁）、beaver（海狸）、beetle（甲壳虫）、crane（鹤）、dove（鸽子）、fly（苍蝇）、seagull（海鸥）、wolverine（狼獾）；

**内部颜色** lily（百合）、orchid（兰花）、rose（玫瑰）；

**外部颜色** dolphin（海豚）、seahorse（海马）、whale（鲸鱼）。

## 细分主题 4 — 字体主题

字体主题设定演示文稿的字体，可以用 `\usefonttheme{主题名}` 命令来选择，其中主题名有如下这些选择：

- ▶ default
- ▶ serif
- ▶ structurebold
- ▶ structureitalicserif
- ▶ structuresmallcapserif

## 主题画廊

Beamer 的整体主题太多了，一个个尝试太费时。你可以在下面这些网址直观地比较这些主题（同时也包含了不同的颜色主题的搭配）：

- ▶ [http://deic.uab.es/~iblanes/beamer\\_gallery/](http://deic.uab.es/~iblanes/beamer_gallery/)
- ▶ <http://www.hartwork.org/beamer-theme-matrix/>

Beamer 自带的各种主题的配色很多都不怎么好看，不过配色可以自己定制，虽然麻烦了点，至少也是可以满足要求的。

1. 基本使用

2. 主题选用

3. 主题定制

4. 各种设置

3.1 三个方面

3.2 内容定制

3.3 外部定制

## 三个方面

Beamer 的各部分的内容都可以自己定制和修改，和主题的划分类似，可以从如下这三个方面来定制自己的主题：

1. 定制模板，用 `\setbeamertheme` 命令
2. 定制颜色，用 `\setbeamercolor` 命令
3. 定制字体，用 `\setbeamerfont` 命令



3.1 三个方面

3.2 内容定制

3.3 外部定制

## 定制背景色

例如，下面的代码修改了演示文稿的渐变背景颜色：

```
\definecolor{bottomcolor}{rgb}{0.32,0.3,0.38}  
\definecolor{middlecolor}{rgb}{0.08,0.08,0.16}  
\setbeamertemplate{background canvas}[vertical shading]  
[bottom=bottomcolor, middle=middlecolor, top=black]
```

其中 `\definecolor` 命令设定了两种颜色，而最后一行设定背景颜色在底部、中部和顶部这三种颜色中渐变。

## 定制标题页

例如，下面的代码修改了文档标题的字体和颜色：

```
\setbeamerfont{title}{size=\LARGE}  
\setbeamercolor{title}{fg=yellow,bg=gray}
```

其中 fg 和 bg 分别表示文字颜色和背景颜色，某一个不指定就表示用默认颜色。

## 定制目录页

例如，下面的代码修改了目录页中节标题的模板和颜色：

```
\setbeamertemplate{section in toc}[sections numbered]  
\setbeamercolor{section in toc}{fg=yellow!80!gray}
```

其中第一行设定显示节标题的编号，第二行将节标题的颜色设为 yellow!80!gray（表示 80% 黄色和 20% 灰色的混合）。

## 定制幻灯片标题

例如，下面的代码修改了每个幻灯片的标题样式：

```
\setbeamertemplate{frametitle}{  
  \noindent\insertframetitle\par  
  \noindent\insertframesubtitle\par}  
\setbeamerfont{frametitle}{size=\large}  
\setbeamercolor{frametitle}{fg=yellow!70!gray}
```

其中第一行的设定使得幻灯片标题和正文对齐，这样看起来更加整齐美观。

## 定制正文字体

例如，下面的代码修改了正文字体的样式：

```
\setbeamercolor{normal text}{fg=white,bg=black}
```

其中黑底白字是看起来比较明显的一种颜色搭配。

## 定制无序列表

无序列表项的样式可以用下面的代码来设定：

```
\setbeamertemplate{itemize items}[样式名]
```

其中样式名一共有如下四种选择（default 和 triangle 一样）：

- ▶ default
- ▶ triangle
- circle
- square
- ball

你可以从上面几种样式中任选一种。

## 定制有序列表

有序列表项的样式可以用下面的代码来设定：

```
\setbeamertemplate{enumerate items}[样式名]
```

其中样式名一共有如下四种选择：

1. default

① circle

① square

① ball

你可以从上面几种样式中任选一种。



## 定制区块环境

例如，下面的代码修改了区块环境的样式：

```
\setbeamertemplate{blocks}[rounded][shadow=true]  
\setbeamercolor{block title}{fg=yellow,bg=gray!50!black}  
\setbeamercolor{block body}{bg=gray}
```

### 重要内容

2012 年 12 月 21 日是世界末日。

其中第一行设定区块环境用圆角带阴影的矩形来表示。

3.1 三个方面

3.2 内容定制

3.3 外部定制

## 定制底栏

例如，下面的代码修改了导航栏和底栏的样式：

```
\setbeamertemplate{navigation symbols}{}  
\setbeamertemplate{footline}[frame number]
```

其中最后一行设定不显示导航栏，而第二行设定底栏只显示页码。

1. 基本使用

2. 主题选用

3. 主题定制

4. 各种设置

## 字号大小

Beamer 演示中全部可以使用的字号如下：8pt、9pt、10pt、11pt、12pt、14pt、17pt、20pt，默认为 11pt。

## 字号大小

Beamer 演示中全部可以使用的字号如下：8pt、9pt、10pt、11pt、12pt、14pt、17pt、20pt，默认为 11pt。建议在较大的场合演示时使用大号的字体，例如：

```
\documentclass [14pt] {beamer}
```

## 字号大小

Beamer 演示中全部可以使用的字号如下：8pt、9pt、10pt、11pt、12pt、14pt、17pt、20pt，默认为 11pt。建议在较大的场合演示时使用大号的字体，例如：

```
\documentclass [14pt] {beamer}
```

Beamer 中的设置的页面大小比较小，仅为 128 毫米乘以 96 毫米。在全屏放映时 PDF 浏览器会自动放大字体，因此，同样的大小看起来要比 article 的情形大很多。

## 抄录环境

在 Beamer 演示中使用 `\verb` 抄录命令或者 `verbatim` 抄录环境时，必须在该 `frame` 中加上 `fragile` 选项，例如：

```
\begin{frame}[fragile]{抄录环境}
```

```
这是一段抄录代码：\verb!\frame{hello beamer}!。
```

```
\end{frame}
```



## 抄录环境

在 Beamer 演示中使用 `\verb` 抄录命令或者 `verbatim` 抄录环境时，必须在该 `frame` 中加上 `fragile` 选项，例如：

```
\begin{frame}[fragile]{抄录环境}
这是一段抄录代码：\verb!\frame{hello beamer}!。
\end{frame}
```

这个选项将导致 Beamer 将该 `frame` 环境的全部内容先写在一个名为 `filename.vrb` 的临时文件里再处理。