

# Perl语言高级编程专题

## Lesson 12

周晓方

[courses@xfzhou.homeftp.org](mailto:courses@xfzhou.homeftp.org)

# Perl Tk

- 最小TK程序

```
use Tk;
```

```
my $mw=MainWindow->new()->MainLoop;
```

- 安装情况

- activeperl 5.6和5.8预装了Perl/Tk
- activeperl 5.5需要手工安装Perl/Tk
- 用这个命令查看是否按照了Perl/Tk模块:  
`perl -e "use Tk;"`

- 参考

- <http://www.lns.cornell.edu/~pvhp/ptk/ptkTOC.html>
- Perl的在线文档C:/Perl/html/site/lib/Tk.html
- 《即时应用Perl模块》(人邮2001年)第6章,创建图形用户界面
- 《Perl编程经典》(清华大学出版社2002年)第21章,图形化与Tk
- Learning Perl Tk, O'Reilly 1999
- Maturing Perl/Tk, O'Reilly 2002



# 如果没有预装Tk怎么办?

- 上CPAN,找到Tk,并且下载一个合适的版本
- 本地安装要用到make工具和C编译器  
记得用编译/安装本地Perl的make和C编译器

The screenshot shows the CPAN website interface. The browser's address bar displays the URL `http://search.cpan.org/~srezic/Tk-804.029/`. The CPAN logo is prominently displayed at the top, along with navigation links: Home, Authors, Recent, News, Mirrors, FAQ, and Feedback. Below the logo is a search bar and a dropdown menu set to 'All'. The main content area is titled 'Slaven Rezić > Tk-804.029' and includes a 'permalink' link. The release details for 'Tk-804.029' are shown, including a 'Download' link, a 'Browse' link, and the release date '27 May 2010'. A table lists other releases, with 'Tk-804.028\_503' selected. The sidebar on the left contains links for 'This Release', 'Other Releases', 'Links', 'CPAN Testers', 'Rating', 'License', 'Special Files', and 'Modules'. The 'Modules' section lists 'PreLoad' and 'Tk800.015'. The footer of the browser window shows the system status '同步被禁用' and a zoom level of '100%'.

Other Releases	Links
Tk-804.028_503 -- 12 May 2010	[Download] [Browse]
Tk-804.028_502 -- 30 Jan 2010	[Goto]
Tk-804.028_501 -- 04 Nov 2008	[64] [Dependencies] [Other Tools]
Tk-804.028_500 -- 01 Oct 2008	[Perl/Platform Version Matrix]
Tk-804.028 -- 18 Dec 2007	[License that is not approved by]
Tk-804.027_502 -- 04 Dec 2007	[distribution without restrictions]
Tk-804.027_501 -- 21 Sep 2007	[README]
Tk-804.027_500 -- 11 Feb 2007	[PL] [ToDo]
Tk-804.027 -- 11 Apr 2004	
Tk-804.026 -- 18 Mar 2004	
Tk800.025 -- 07 Sep 2003	
Tk-800.024 -- 17 Mar 2002	
Tk800.015 -- 28 Jul 1999	

# Perl Tk version "*Hello World!*"

- Perl Tk程序的基本结构

```
use Tk;
```

```
#创建主窗口
```

```
my $main=new MainWindow;
```

```
...构造窗体, ...
```

```
.....添加组件...
```

```
#启动事件循环
```

```
$main->MainLoop;
```

```
...主程序结束...
```

```
...定义子程序...
```

```
1;
```



```
#!/usr/bin/perl -w
use strict;
use Tk;
my $main = new MainWindow;
$main->title('^_^');
$main->Label(-text=>'Hello World!')->pack;
$main->Label(-text=>'This is Perl/TK!')
    ->pack;
$main->Button(-text => 'Quit',
    -command => sub{exit}
)->pack;
$main->MainLoop;
1;
```

# 向组件传递参数的一般方法

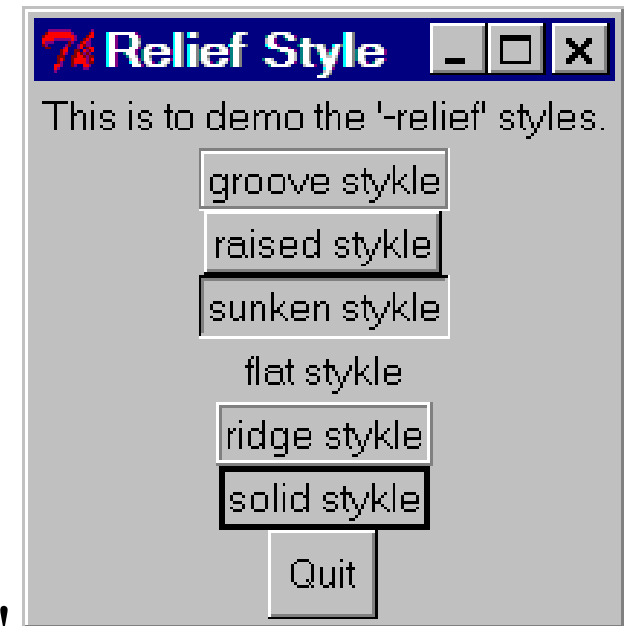
- 创建时传递参数
  - `$T = $main->Label(-text => '#!^', -o2 => V2, -o3 => V3 ...)`
- 取回参数
  - `$V2 = $T->cget('-o2');`
- 修改参数
  - `$T->configure(-option => new_value);`
- 了解参数情况
  - `@a = $T->configure('-background');` 返回:(background, background, Background, SystemButtonFace, SystemButtonFace)
  - *(optionName, Name, Class, Default Value, Current Value)*
  - `@a = $T->configure;` 返回所有参数情况(二维数组)

# 组件的常用参数(I)

- 文本相对组件的位置(按钮、标签等)
  - anchor => 'center' 'n' 'ne' 'e' 'se' 's' 'sw' 'w' 'nw'
- 背景颜色, 前景颜色
  - background => 'red'                      表示颜色的英文
  - foreground => '#00ff00'                  表示颜色的16进制RGB
- 显示的位图或图标(按钮, 菜单按钮等)
  - bitmap=>'question'
  - bitmap=>'@bitmap\_filename'
- 边框宽度, 高度, 宽度
  - 单位可以是c厘米, I英寸, m毫米, p像素
  - borderwidth => '4p' (缺省是2像素)
  - height => 20 (缺省是0)
  - width => 100 (缺省是0)

# 组件的常用参数(II)

- 边框样式
  - relief => 'groove' 'raised' 'sunken' 'flat' 'ridge' 'solid'
- 响应状态
  - state => 'disabled' 'normal' 'active'
- 文本
  - text => 'Hello world!'
- 变量文本(随时改变变量的值, 随时更新显示)
  - my \$var = "AAA";
  - textvariable => \ \$var



# 主窗体， 标签窗口

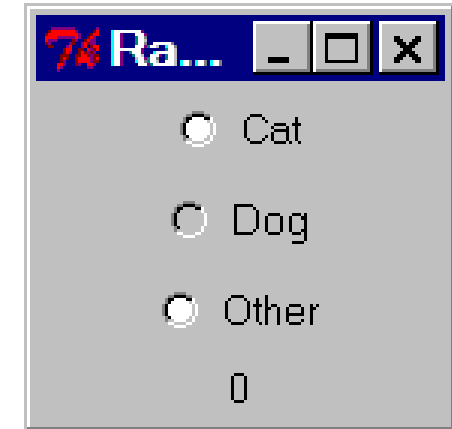
- 主窗口 (MainWindow)
  - 创建 `my $main = new MainWindow;`
  - 改标题 `$main -> title("标题");`
  - 取回标题 `$main->cget('-title');`
  - 父窗口 `$main -> parent` # always returns *undef*
  - 启动事件循环 `$main->MainLoop; #通常是最后一条命令`
- 标签窗口 (Label)
  - 静态文本 `-text => '....'`
  - 动态文本 `-textvariable => \ $var`
  - 显示样式 `-releaf => '....'`
  - `$main->Label(-text=>'Hello')->pack();`



# 单选按钮 (多选一)

- Radiobutton()

- 文本: `-text => '??'`
- 对应值: `-value => ??`
- 对应变量: `-variable => \ $V`
- 同一组单选按钮共用同一个变量。



```
use Tk;
my $pet = 0;
my $main = new MainWindow;
$main->Radiobutton(-text => 'Cat',
    -value => 1, -variable => \ $pet)->pack();
$main->Radiobutton(-text => 'Dog',
    -value => 2, -variable => \ $pet)->pack();
$main->Radiobutton(-text => 'Other',
    -value => 3, -variable => \ $pet)->pack();
$main->Label(-textvariable => \ $pet
    ->pack();
$main->MainLoop;
1;
```

# 多选按钮

- Checkbutton()

- 文本

- text => '??'

- 对应值

- onvalue => ??

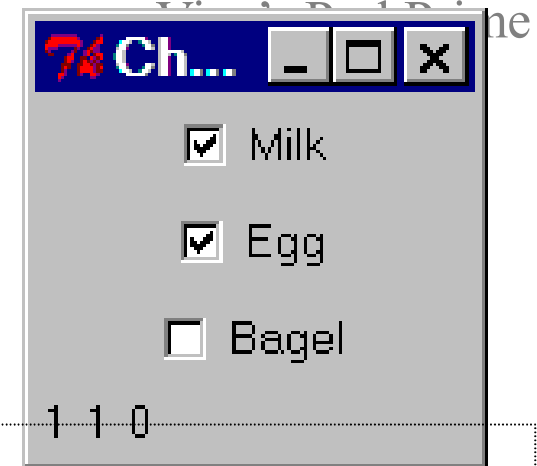
- offvalue=> ??

- 默认值分别是  
1和0

- 对应变量

- variable => \ \$V

- 不同的选项  
使用不同的  
变量.



```
use Tk;
my ($milk, $egg, $bagel) = ('-', '-', '-');
my $main = new MainWindow;
$main->Checkbutton(-text => 'Milk', -
onvalue => 1, -offvalue => 0,
    -variable => \$milk)->pack();
$main->Checkbutton(-text => 'Egg',
    -variable => \$egg)->pack();
$main->Checkbutton(-text => 'Bagel',
    -variable => \$bagel)->pack();
$main->Label(-textvariable => \$milk)
    ->pack(-side => 'left');
$main->Label(-textvariable => \$egg)
    ->pack(-side => 'left');
$main->Label(-textvariable => \$bagel)
    ->pack(-side => 'left');
$main->MainLoop;
1;
```

# 输入项

- Entry()

- 输入一行文字
- 输入密码时，采用-show选项
- 用get()命令取结果，也可以使用-textvariable选项将输入保存到指定变量中

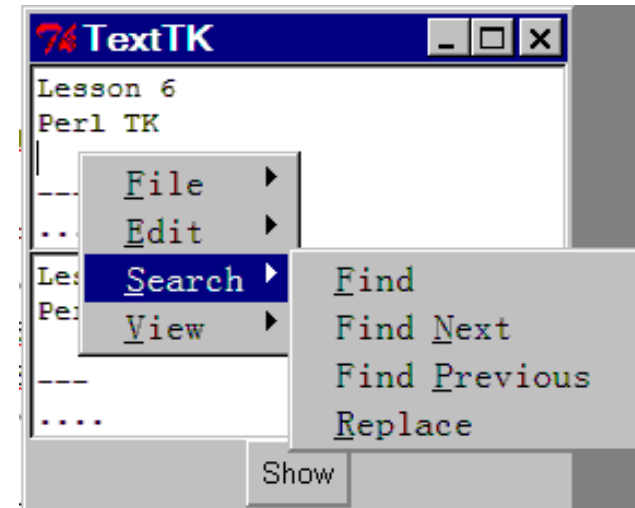


```
use Tk;
my ($name, $pin);
my $main = new MainWindow;
$name = $main->Entry;
$pin = $main->Entry(-show => '*');
$name -> pack(); $pin -> pack();
$main -> Button(-text => 'Ok',
               -command => sub {
                           print $name->get(), "\n";
                           print $pin ->get(), "\n";
                           exit;
                           })->pack();
$main->MainLoop;
1;
```

- Text()

# (多行)文本

- 长宽: -width, -height
- 状态: -state => 'normal' 'disabled'
- 插入方法 insert(from, string)
- 删除方法 delete(from, to)
- 读取方法 get(from, to)



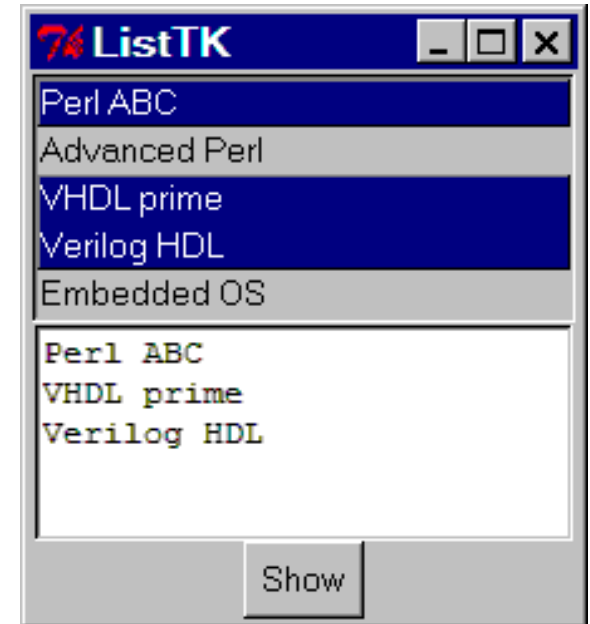
- 指定位置  
'行.列'
- 文本末端  
'end'

```
use Tk;
my ($text, $out);
my $main = new MainWindow;
$text = $main->Text(-width => 30, -height => 5)->pack;
$out = $main->Text(-width => 30, -height => 5,
                  -state => 'disabled')->pack;
$main -> Button(-text => 'Show',
               -command => sub {
                   $out->configure(-state => 'normal');
                   $out->delete('1.0', 'end');
                   $out->insert('end', $text->get('1.0', 'end'));
                   $out->configure(-state => 'disabled');
               })->pack();
$main->MainLoop;
1;
```

# 列表框

- 替代单选、多选，节约空间

```
use Tk;
my ($list, $out);
my $main = new MainWindow;
$list = $main->Listbox(-width => 30, -height => 5,
                      -selectmode => 'multiple')->pack();
$list->insert('end', 'Perl ABC', 'Advanced Perl',
             'VHDL prime', 'Verilog HDL',
             'Embedded OS', 'Multimedia');
$list->bind('<Double-1>', \&get_item);
$out = $main->Text(-width => 26, -height => 5)
      ->pack();
$main->Button(-text => 'Show',
             -command => \&get_item)->pack();
$main->MainLoop;
1;
sub get_item {
    $out->delete('1.0', 'end');
    my @sel = $list->Getselected();
    for (1..@sel) {
        $out->insert("$_.0", "$sel[$_]\n");
    }
}
```

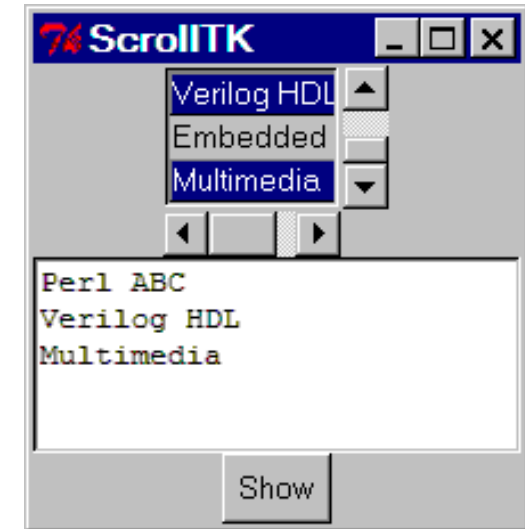


- -selectmode
  - single
  - multiple
- get(index)返回串
- insert(位置,...)
  - delete(from, to)
    - n, from 0
    - 'active'
    - 'end'
- Getselected()
- SelectionSet(index)

# 滚动条

- 带滚动条的列表框 `Scrolled()`

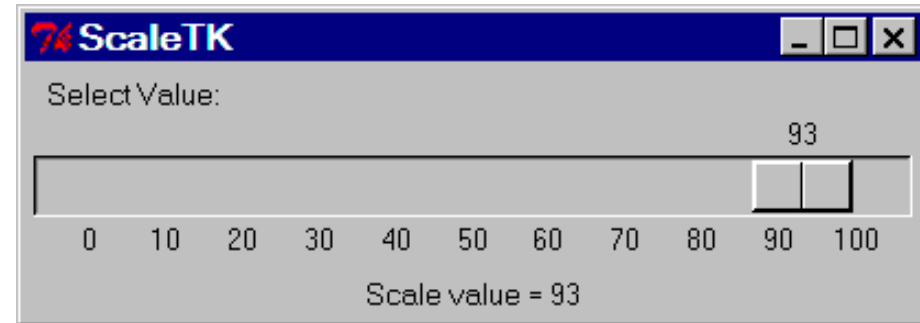
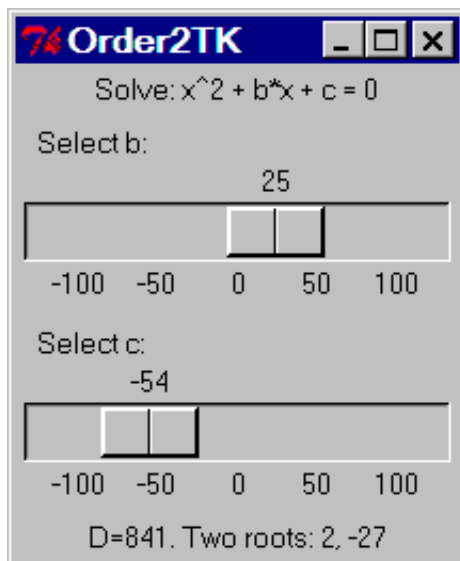
```
use Tk;
my ($list, $out);
my $main = new MainWindow;
$list = $main->Scrolled('Listbox', -width => 10,
    -height => 3, -selectmode => 'multiple',
    -scrollbars => 'se')->pack();
$list->insert('end', 'Perl ABC', 'Advanced Perl',
    'VHDL prime', 'Verilog HDL',
    'Embedded OS', 'Multimedia');
$list->bind('<Double-1>', \&get_item);
$out = $main->Text(-width => 26, -height => 5)
    ->pack();
$main -> Button(-text => 'Show',
    -command => \&get_item)->pack();
$main->MainLoop;
1;
sub get_item {
    $out->delete('1.0', 'end');
    my @sel = $list->Getselected();
    for (1..@sel) {
        $out->insert("$_.0", "$sel[$_]\n");
    }
}
```



- `Scrolled('类型', ...)`
- `-scrollbars => 方位`
  - s,e,n,w

# 刻度标尺

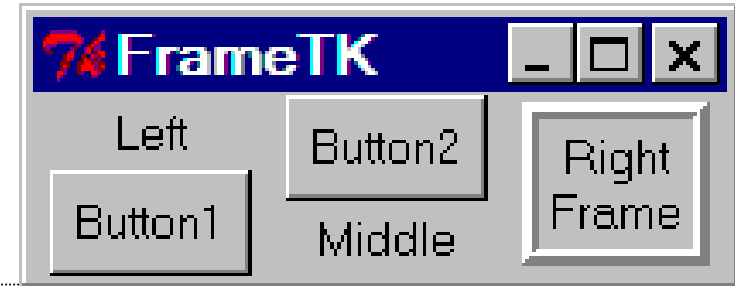
- Scale()
  - orient 水平或垂直
  - from, –to 区间范围
  - tickinterval 刻度间隔
  - label
  - length 屏幕显示宽度
  - variable 更新的变量
  - command 回调函数



```
#!/usr/bin/perl -w
use strict;
use Tk;
my ($scale, $text, $value);
my $main = new MainWindow;
$scale = $main->Scale(-orient => 'horizontal',
    -from => 0, -to => 100, -tickinterval => 10,
    -label => 'Select Value:', -length => 400,
    -variable => \$value, -command => \&show)
    ->pack();
$main -> Label(-textvariable => \$text)->pack();
$main->MainLoop;
1;

sub show {
    $text = "Scale value = $value";
}
```

# 框架



- 将窗口分组的容器Frame()

- borderwidth
- relief
- height
- width

- pack(-side=>..)

- top
- left
- right
- bottom

```
#!/usr/bin/perl -w
use strict;
use Tk;
my ($f1, $f2, $f3);
my $main = new MainWindow;
$f1 = $main->Frame()
    ->pack(-side => 'left', -padx => 5);
$f1->Label(-text=>'Left')->pack();
$f1->Button(-text=>'Button1')->pack();

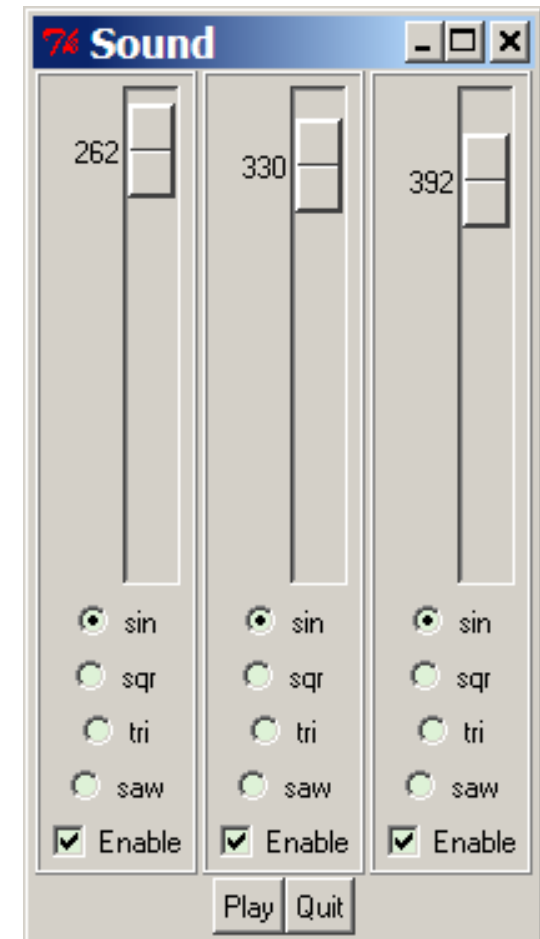
$f2 = $main->Frame()
    ->pack(-side => 'left', -padx => 5);
$f2->Button(-text=>'Button2')->pack();
$f2->Label(-text=>'Middle')->pack();

$f3 = $main->Frame(-relief=>'ridge',
    -borderwidth=>5)
    ->pack(-side => 'left', -padx => 5);
$f3->Label(-text=>"Right\nFrame")->pack();
$main->MainLoop;
1;
```



# 回家作业（不做了）

- 做一个Tk程序，控件及其含义：  
    – 刻度标尺——产生频率200Hz-2000Hz的波形  
    – 单选按钮——波形种类有正弦波、方波、三角波、锯齿波  
    – 多选按钮——使能3组频率和种类控件  
    – PLAY按钮——用Win32::Sound播放3秒钟,用44100高保真波形吧  
    – QUIT按钮——退出  
    – Frame框架 ——协助合理布局



# 菜单条

## • Menubutton()

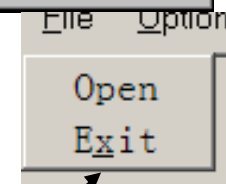
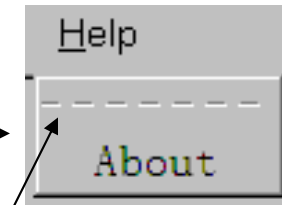
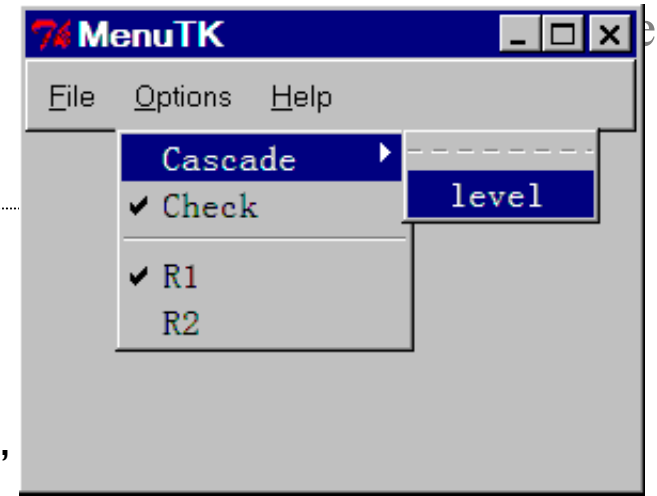
```
use Tk;
my($ck, $r);
my $main = new MainWindow;
my $menubar = $main->Frame(-relief => 'raised',
    -borderwidth => 2)->pack(-side => 'top', -fill => 'x');
my $menufile = $menubar->Menubutton(-text => 'File',
    -underline => 0, -tearoff => 0) -> pack(-side => 'left');
my $menuopts = $menubar->Menubutton(-text => 'Options',
    -underline => 0, -tearoff => 0) -> pack(-side => 'left');
my $menuhelp = $menubar->Menubutton(-text => 'Help',
    -underline => 0, -tearoff => 1) -> pack(-side => 'left');

$menufile->command(-label => 'Open');
$menufile->command(-label => 'Exit', -command => sub{exit}, -underline => 1);

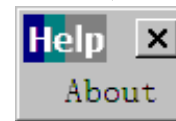
my $menucasc = $menuopts->menu->Menu();
$menuopts->add('command', -label => 'level');
$menuopts->cascade(-label => 'Cascade');
$menuopts->entryconfigure('Cascade', -menu => $menucasc);
$menuopts->checkboxbutton(-label => 'Check', -variable => \$ck);
$menuopts->separator();
$menuopts->radiobutton(-label => 'R1', -variable => \$r);
$menuopts->radiobutton(-label => 'R2', -variable => \$r);

$menuhelp->command(-label => 'About');
$main->Label(-text => "Hello\nworld!", -height => 10, -width => 40)->pack;
$main->MainLoop;
1;
```

'x'水平方向填满



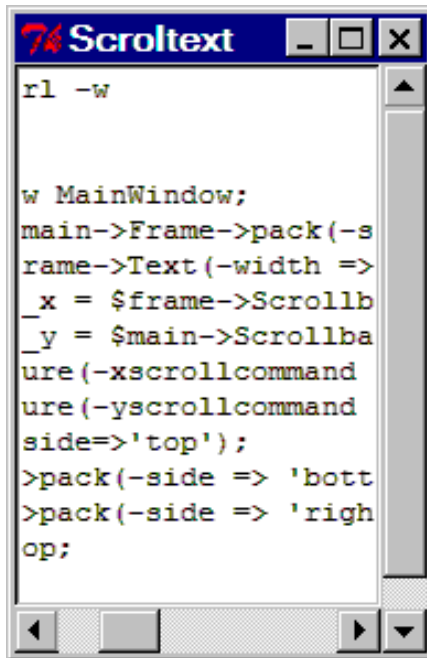
E~xit



单击Tearoff后  
形成独立窗口

# 滚动条窗口

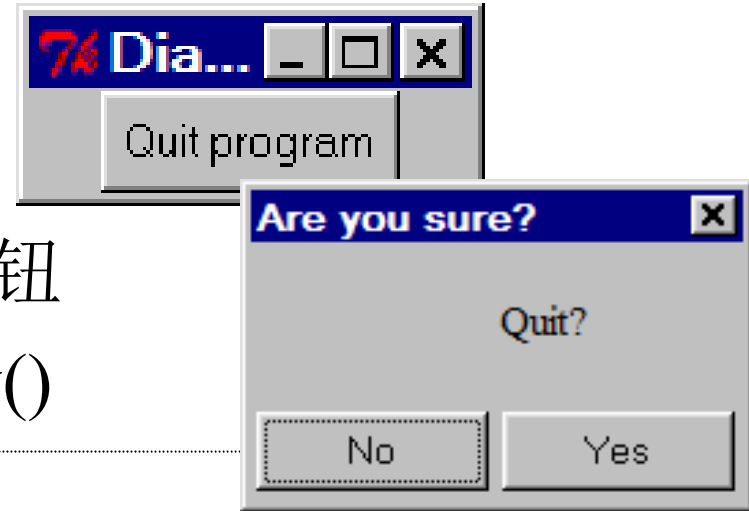
- Scrollbar()
  - command=>['xview', \$wedge] 或 ['yview', \$wedge]
  - orient=>'vertical' 或 'horizontal'
- \$wedge:
  - xscrollcommand=>['set', \$scrollbar\_x]
  - yscrollcommand=>['set', \$scrollbar\_y]



```
use Tk;
my $main = new MainWindow;
my $frame = $main->Frame-> pack(-side=>'left');
my $text = $frame->Text(-width => 30,
                        -height => 4)->pack(-side=>'left');
my $scrollbar_x = $frame->Scrollbar(-command =>
    ['xview', $text], -orient => 'horizontal');
my $scrollbar_y = $main->Scrollbar(-command =>
    ['yview', $text]);
$text->configure(-xscrollcommand => ['set', $scrollbar_x]);
$text->configure(-yscrollcommand => ['set', $scrollbar_y]);
$text->pack(-side=>'top');
$scrollbar_x->pack(-side => 'bottom', -fill => 'x');
$scrollbar_y->pack(-side => 'right', -fill => 'y');
$main->MainLoop;
1;
```

# 简单对话框

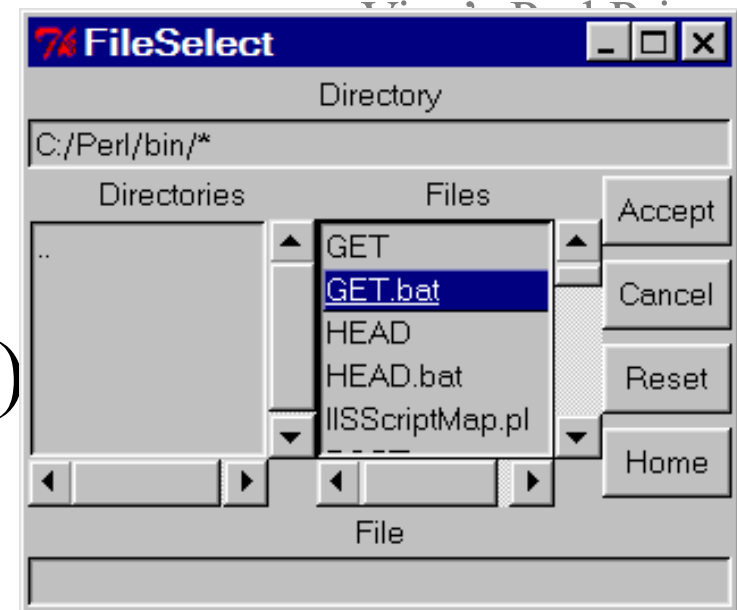
- Dialog()
  - use Tk::Dialog;
  - 只包含文本提示和一组按钮
  - 调用方法: \$dialog->Show()



```
use Tk;
use Tk::Dialog;
my $main = new MainWindow;
my $dialog = $main->Dialog(
    -title      => 'Are you sure?', -text => 'Quit?',
    -default_button => 'No',      -buttons => ['No', 'Yes']
);
$main->Button(
    -text      => 'Quit program',
    -command=> sub{
        exit if $dialog->Show eq 'Yes';
    }
)->pack;
$main->MainLoop;
1;
```

# 文件选择对话框

- use Tk::FileSelect;
- FileSelect(-directory=>目录)



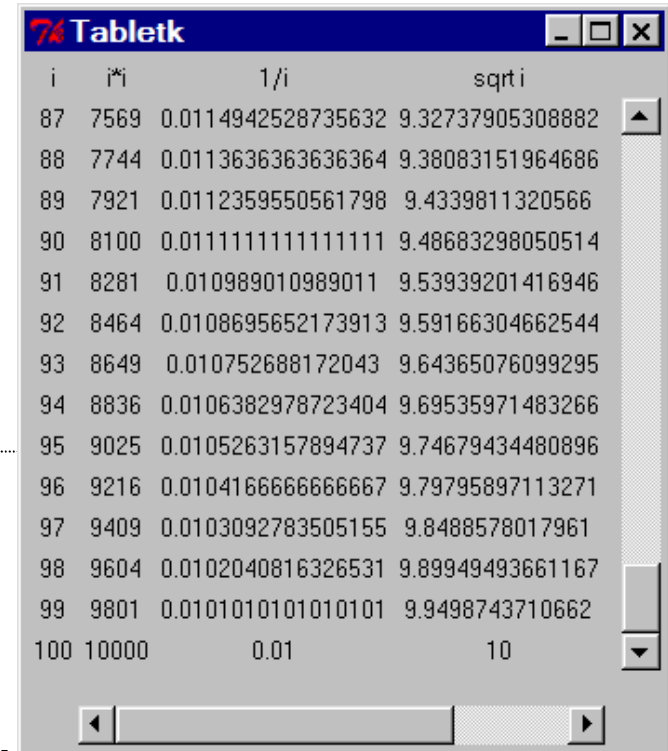
```
use Tk::FileSelect;
my $main = new MainWindow;
my $fs = $main->FileSelect(-directory=>'c:');
my $file = "";
$main->Label(-textvariable => \$file)->pack;
$main->Button(
    -text    => 'Select File',
    -command=> sub{
        $file = $fs->Show;
    }
)->pack;
$main->MainLoop;
1;
```

# 表格

- use Tk::Table;
- 可滚动的表格

```
use Tk;
use Tk::Table;
my $main = new MainWindow;
my $tab = $main->Table(
    -height => 10, -rows => 100,
    -columns => 6, -scrollbars => 'se',
    -fixedrows => 1, -fixedcolumns => 1,
    -takefocus => 1)->pack;
$tab->put(0, 0, 'i');      $tab->put(0, 1, 'i*i');  $tab->put(0, 2, '1 / i');
$tab->put(0, 3, 'sqrt i');$tab->put(0, 4, 'ln i');

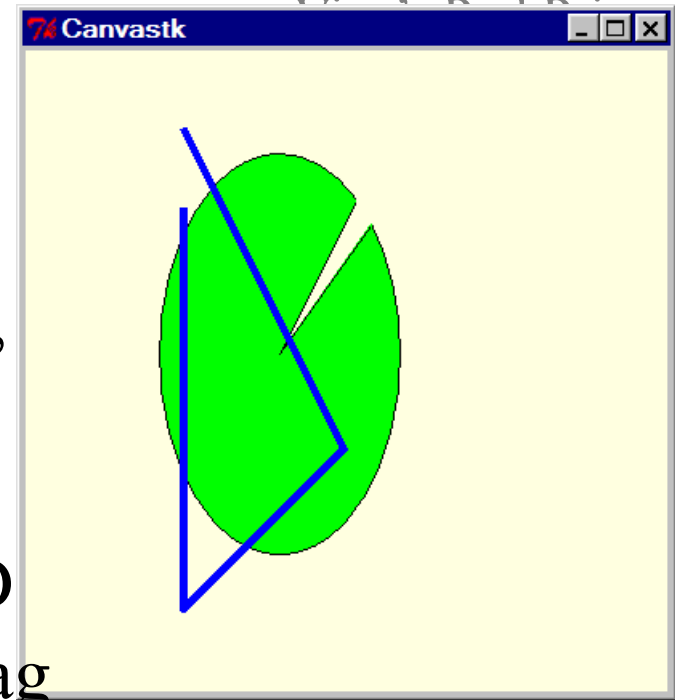
foreach (1..100) {
    $tab->put($_, 0, $_);
    $tab->put($_, 1, $_*$_);
    $tab->put($_, 2, 1/$_);
    $tab->put($_, 3, sqrt $_);
    $tab->put($_, 4, log $_);
}
$main->MainLoop;
1;
```



i	$i^2$	$1/i$	$\sqrt{i}$	$\ln i$
87	7569	0.0114942528735632	9.32737905308882	
88	7744	0.0113636363636364	9.38083151964686	
89	7921	0.0112359550561798	9.4339811320566	
90	8100	0.0111111111111111	9.48683298050514	
91	8281	0.010989010989011	9.53939201416946	
92	8464	0.0108695652173913	9.59166304662544	
93	8649	0.010752688172043	9.64365076099295	
94	8836	0.0106382978723404	9.69535971483266	
95	9025	0.0105263157894737	9.74679434480896	
96	9216	0.0104166666666667	9.79795897113271	
97	9409	0.0103092783505155	9.8488578017961	
98	9604	0.0102040816326531	9.89949493661167	
99	9801	0.0101010101010101	9.9498743710662	
100	10000	0.01	10	

# 画布Canvas

- Canvas()
  - Arc, Line, Bitmap, Image, Oval, Polygon, Rectangle, Text, ...
- Item Tags and IDs
  - 画布上每个零件都有独立的ID
  - 一个或一组零件可用有一个Tag
  - bind方法，给每个零件定义回调函数
  - move方法，改变零件的位置
  - lower和raise方法，改变零件的Z-顺序
  - 画布可用保存为postscript文件
  - ..lots more features...

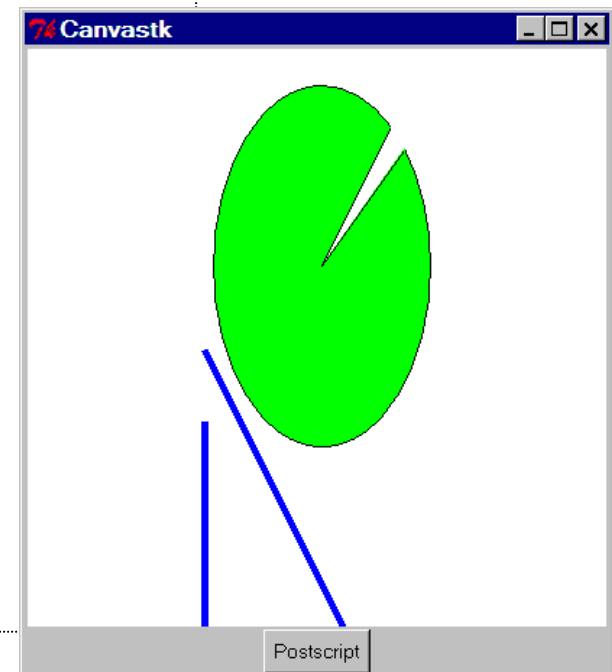
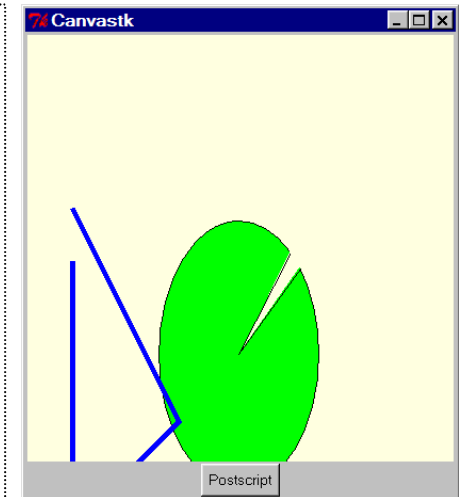


转.ps文件为.pdf/.png/.jpeg的模块 **PostScript::Convert**  
记得先安装**Ghostscript**，并确保**gswin32c.exe**目录加入路径

# 画布实例——会移动的圆弧、线条

```
#!/usr/bin/perl -w
use strict;
use Tk;
use Tk::Canvas;
my $main = new MainWindow;
my $canvas = $main->Canvas(
    -background => 'LightYellow',
    -height      => 400,
    -width       => 400
)->pack;
my $arc = $canvas->createArc(100,100,250,350, -extent => 350,
    -outline=>'black', -start=>50, -fill=>'green');
my $line = $canvas->createLine(100,100,100,350,200,250,100,50,
    -fill=>'blue',-width=>5);

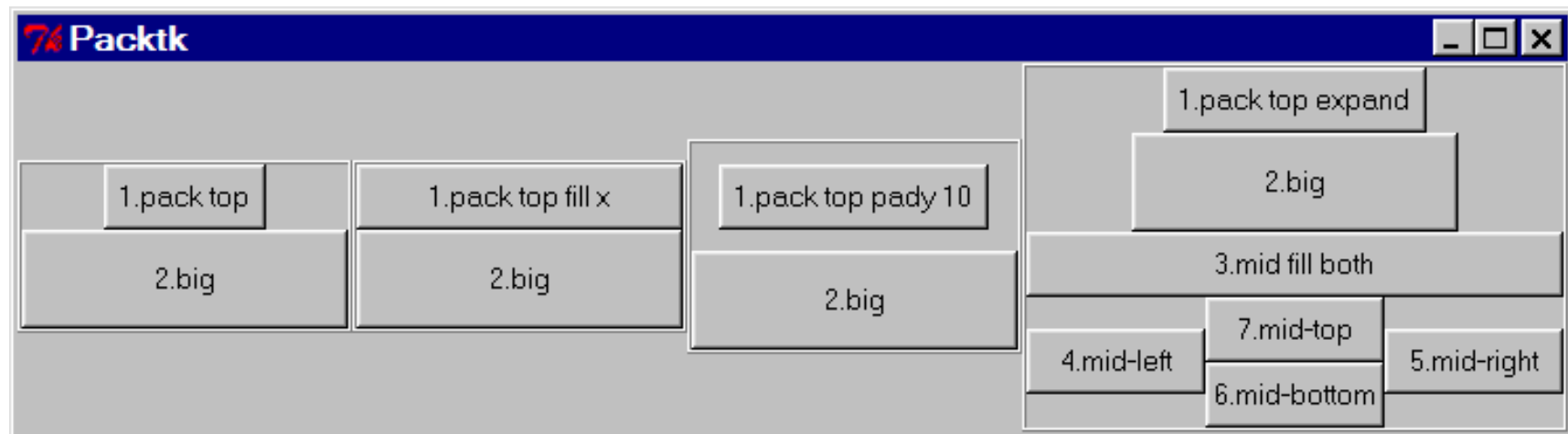
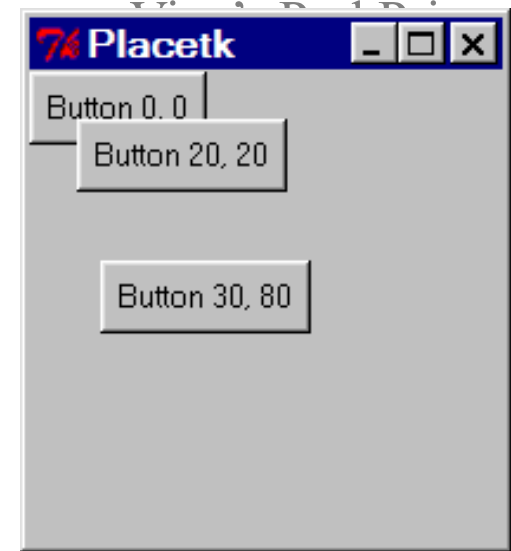
$canvas->bind($arc, '<Button-1>', [ \&jump, $arc ]);
$canvas->bind($line, '<Double-1>', [ 'jump', $line]);
$main->Button(-text=>'Postscript', -command=>sub {
    $canvas->postscript(-file=>'canvas.ps')
})->pack;
$main->MainLoop;
1;
sub jump{
    my($canvas) = shift;
    my($id) = shift;
    $canvas->move($id, rand(100) - 50, rand(100) - 50);
}
```





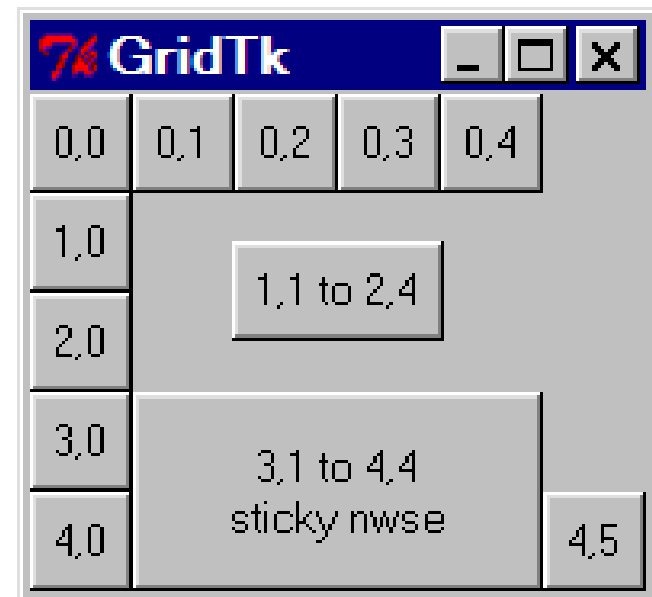
# place和pack

- `place(-x=>??, -y=>??)`
  - 指定组件的象素位置，组件可重叠
- `pack()`
  - `side=>'top' 'left' 'right' 'bottom'`
  - `expand=>0或1`，是否充满余下的窗口空间
  - `fill=>'none' 'x' 'y' 'both'`在指定方向上填充
  - `padx=>数量, pady=>数量`，给定填充的大小



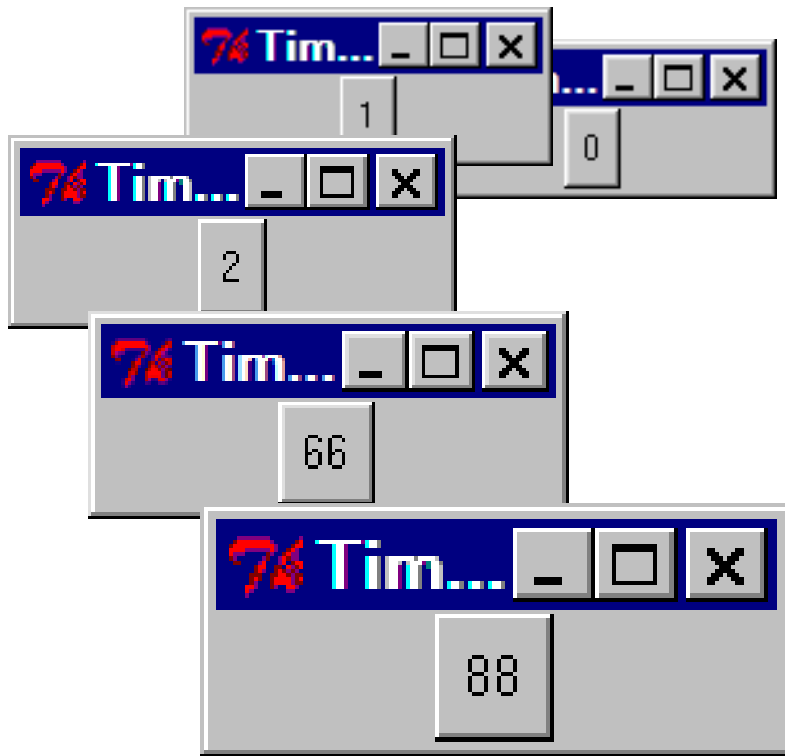
# Grid几何图形管理

- 除了pack外，grid()也是常用的图形管理方法
  - -row/-column指定组件所在行/列，从0开始
  - -rowspan/-columnspan组件可以跨多个行/列
  - -sticky=>方位，组件在给定矩形中的相对位置，可以是n/s/w/e的组合，如果nwse，就充满空间。
  - 和pack()类似，Grid安排的组件之间不会重叠，除非在一个格点中安排多个组件。



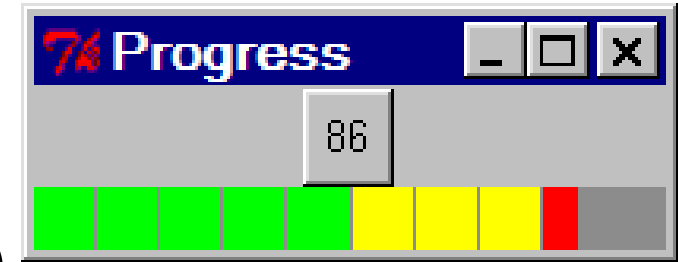
# Tk的非抢先式定时回调功能

- `$id = $widget->after(ms, cmd)`定时并执行
- `$id = $widget->repeat(ms, cmd)`重复定时执行
- `$widget->afterCancel($id)`或`$id->cancel`, 取消



```
use Tk;
my($t) = 0;
my $main=MainWindow->new();
my $id = $main->Button(
    -textvariable=>\$t,
    -command=>\&cancel)->pack()
    ->repeat(1000,\&count);
$main->MainLoop;
1;
sub count {
    $t++;
}
sub cancel{
    $id->cancel;
}
```

# 进度条

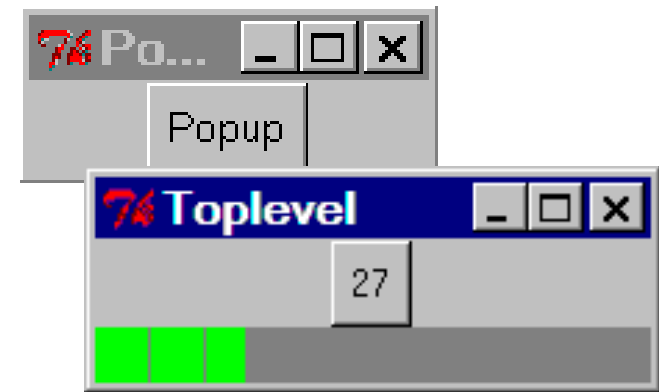


- Tk::ProgressBar, ProgressBar()
  - anchor 水平条 'e' 'w' 垂直条 'n' 's'
  - blocks 分块大小, -gap 块之间的距离
  - colors => [位置, 颜色, 位置, 颜色, ...]
  - from, -to, 区间
  - length, -width, 尺寸
  - variable => \ \$变量,  
建议通过 value()  
方法设置进度

```
.....同上例.....  
my $progress = $main->ProgressBar  
(  
    -width => 20,  
    -length => 200,  
    -anchor => 'w',  
    -from => 0,  
    -to => 100,  
    -blocks => 10,  
    -colors=> [0, 'green', 50, '  
                yellow', 80, 'red'],  
    -variable => \ $t  
)->pack;  
$main->MainLoop;  
.....
```

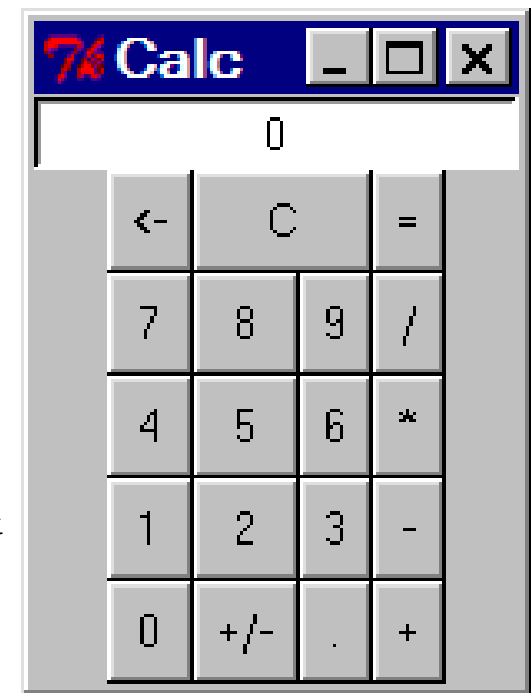
# 多窗口

- `$schild = $parent->Toplevel();`  
创建一个新的窗口，然后在  
`$schild`上填写各种窗口组件
- `$schild->destroy;`关闭`$schild`窗口
- 参考popup.pl



## 回家作业 11

- 用Tk界面编写一个简单的计算器
- 学号-11.pl
- 参考老式计算器，消息 $\leftrightarrow$ 状态✓，过程控制流✗



# Perl/Tk的工具

- 尝试Alexander Adam的*Perl/Tk Desinger*
- 类似Vb的界面，但缺少说明文档或软件更新

