```
# SpecTcl, by S. A. Uhler and Ken Corey
# Copyright (c) 1994−1995 Sun Microsystems, Inc.
#
# See the file "license.txt" for information on usage and redistribution
# of this file, and for a DISCLAIMER OF ALL WARRANTIES.
#
# pick a color menu item
# This is a placeholder for now

if {![info exists Colors]} {
    set Colors {black white bisque grey red green blue magenta yellow}
}

# pop−up a color menu
# return selected color, updating color list as needed (in global Color)
#  win: the toplevel (created if needed)
#  colors: The name of the variable containing the current colors
```

setup_colors

```
proc setup_colors {win colors {cmd puts}} {
    global Color P
    upvar $colors list

    # create the top level (if needed)

    catch "destroy $win"
    toplevel $win −bd 2 −relief ridge  −cursor [cursor menu]
    wm overrideredirect $win 1
    set parent [winfo parent $win]
    foreach i {rootx rooty height} {
    set $i [winfo $i $parent]
    }
    wm geometry $win +$rootx+[expr $rooty + $height]

    # pack in existing colors, set bindings

    set index 0
    foreach i $list {
    frame $win.${i}$index −width 14 −height 14 \
        −bd 2 −relief raised −bg $i −cursor [cursor item]
    grid $win.${i}$index −row [expr $index/4] −column [expr $index%4]
    bind $win.${i}$index <ButtonRelease−1> "$cmd $i"
    bind $win.${i}$index <Enter> "%W configure -relief sunken"
    bind $win.${i}$index <Leave> "%W configure -relief raised"
    incr index
    }
    label $win.more −text "more..." −padx 1 −pady 1 −bd 2 −relief raised
    grid $win.more −row [expr ($index−1)/4+1] −column 0 −columnspan 4 −sticky {e
w}
    bind $win.more <Enter> "%W configure -relief sunken"
    bind $win.more <Leave> "%W configure -relief raised"
    bind $win.more <ButtonRelease−1> "more_colors %W Colors"
    bind $win <ButtonRelease−1>  \
    {catch {destroy [winfo toplevel %W]}}
```

```
    tkwait visibility $win
    grab $win
    set P(default:foreground) {}
}

# handle additions/changes to color palette
```

more_colors

```
proc more_colors {win array} {
    global Colors
    dputs "more colors <$win> <$Colors>"

    if {[set new [tk_chooseColor]] != ""} {
    if {[lsearch $Colors $new] == -1} {
        lappend Colors $new
    }
    }
}
```