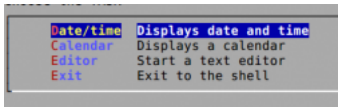
# A menu box



- A **menu box** display a list of choices to the user in the form of a menu.
- Each menu is made of a *tag* string and an *item* string. In this example, a tag (e.g., Calendar) is on left side and an item (e.g., "Displays a calendar") is on right side:

```
Date/time "Displays date and time" \
Calendar "Displays a calendar" \
Editor "Start a text editor" \
Exit "Exit to the shell"
```

- The *tag* gives the entry a name to distinguish it from the other entries in the menu. Use the *tag* to make decision using if statement or case..esac statement.
- The item is nothing but a short description of the option that the entry represents.
- All choices (menus) are displayed in the order given.
- On exit the *tag* of the chosen menu entry will be printed on dialog's output. This can be redirected to the file using the following syntax:



A sample menu output

```
> /tmp/menu.output
```

- If the "--help-button" option is given, the corresponding help text will be printed if the user selects the help button.

## Example

- Create a shell script called utilitymenu.sh:

```
#!/bin/bash
# utilitymenu.sh - A sample shell script to display menus on screen
# Store menu options selected by the user
INPUT=/tmp/menu.sh.$$

# Storage file for displaying cal and date command output
OUTPUT=/tmp/output.sh.$$

# get text editor or fall back to vi_editor
vi_editor=${EDITOR-vi}

# trap and delete temp files
trap "rm $OUTPUT; rm $INPUT; exit" SIGHUP SIGINT SIGTERM

#
# Purpose - display output using msgbox
#  $1 -> set msgbox height
#  $2 -> set msgbox width
#  $3 -> set msgbox title
#
function display_output(){
    local h=${1-10}         # box height default 10
    local w=${2-41}         # box width default 41
    local t=${3-Output}     # box title
    dialog --backtitle "Linux Shell Script Tutorial" --title "${t}" --clear --msgbox "$(<$OUTPUT)" ${h} ${w}
}
#
# Purpose - display current system date & time
#
function show_date(){
    echo "Today is $(date) @ $(hostname -f)." >$OUTPUT
    display_output 6 60 "Date and Time"
}
#
# Purpose - display a calendar
#
function show_calendar(){
    cal >$OUTPUT
    display_output 13 25 "Calendar"
}
#
# set infinite loop
#
while true
do

### display main menu ###
dialog --clear  --help-button --backtitle "Linux Shell Script Tutorial" \
--title "[ M A I N - M E N U ]" \
--menu "You can use the UP/DOWN arrow keys, the first \n\
letter of the choice as a hot key, or the \n\
number keys 1-9 to choose an option.\n\
Choose the TASK" 15 50 4 \
Date/time "Displays date and time" \
Calendar "Displays a calendar" \
Editor "Start a text editor" \
Exit "Exit to the shell" 2>"${INPUT}"

menuitem=$(<"${INPUT}")


# make decsion
case $menuitem in
    Date/time) show_date;;
    Calendar) show_calendar;;
    Editor) $vi_editor;;
    Exit) echo "Bye"; break;;
esac

done

# if temp files found, delete em
```

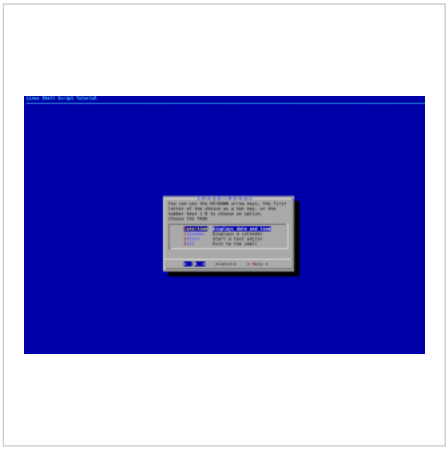
```
[ -f $OUTPUT ] && rm $OUTPUT
[ -f $INPUT ] && rm $INPUT
```

Save and close the file. Run it as follows:

```
chmod +x utilitymenu.sh
./utilitymenu.sh
```

Sample outputs:

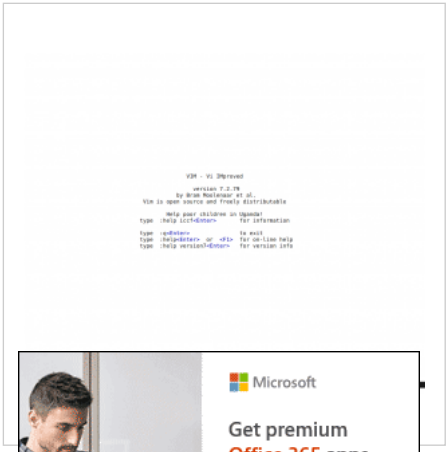
**utilitymenu.sh shell script output (dialog command with menus)**



A menu based dialog box


Messagebox displaying date & time


Messagebox displaying a calendar



V