

```
#!/bin/bash
```

```
#Funcion que muestra los menus
```

```
function mostrar_menu()
```

```
{
```

```
    #Crea una variable llamada salida que contiene un archivo temporal para almacenar salida del menu
```

```
    salida=/tmp/menu.sh
```

```
    ar=() #Array en el que guardamos el texto de cada linea
```

```
    cod=() #Array que guarda el codigo tambien almacenado en el archivo que usamos para hacer un
```

```
return.
```

```
    titulo="" #Almacena el titulo que se le dara al menu
```

```
    option="" #Almacena us subtítulo
```

```
    aceptar="" #Almacena el mensaje del boton aceptar
```

```
    salir="" #Almacena el mensaje del boton salir
```

```
    cont=0
```

```
#Bucle para almacenar las lineas del archivo txt/$idioma/$1
```

```
while read n ; do
```

```
    if [ $cont -eq 0 ]; then
```

```
        titulo=$n;
```

```
    elif [ $cont -eq 1 ]; then
```

```
        option=$n;
```

```
    elif [ $cont -eq 2 ]; then
```

```
        aceptar=$n;
```

```
    elif [ $cont -eq 3 ]; then
```

```
        salir=$n;
```

```
    else
```

```
        #Guarda en el array ar la linea hasta el ";", mensaje que tiene que mostrar en
```

```
el menu
```

```
        ar+=(`echo $n | cut "-d;" -f1`);
```

```
        #Guarda en el array cod la linea después del ";", codigo que tiene que usara
```

```
para el return
```

```
        cod+=(`echo $n | cut "-d;" -f2`);
```

```
    fi
```

```
    let cont=$cont+1
```

```
done < txt/$idioma/$1 #archivo que se le pasa al bucle
```

```
#Crea el menu pasandole el array ar
```

```
#Guarda la salida en $salida
```

```
$salir" \
```

```
    --title "$titulo" \
```

```
    --menu "$option" 17 50 12 "${ar[@]}" \
```

```
2>"${salida}"
```

```
#La variable menuitem recibe como entrada el contenido de $salida que contiene un archivo temporal
```

```
menuitem=$(<"${salida}");
```

```
#Devuelve el codigo del array "cod" en la posicion correspondiente.
```

```
return ${cod[$menuitem-1]};
```

```
}
```

```
function menu_parametros()
```

```
{
```

```
    #Crea una variable llamada salida que contiene un archivo temporal para almacenar salida del menu
```

```
    salida="/tmp/menu.para.sh"
```

```
    ar=() #Array en el que guardamos el texto de cada linea
```

```
    opt=() #Array en el que guardamos lo que le pasamos a dialog
```

```
    pass=()
```

```
    var=()
```

```
    parametros=() #Array que guarda los parametros que le pasaremos a la funcion
```

```

titulo="" #Almacena el titulo que se le dara al menu
aceptar="" #Almacena el mensaje del boton aceptar
salir="" #Almacena el mensaje del boton salir
cont=0

#Bucle para almacenar las lineas del archivo txt/$idioma/$1
while read n ; do
    if [ $cont -eq 0 ]; then
        titulo=$n;
    elif [ $cont -eq 1 ]; then
        aceptar=$n
    elif [ $cont -eq 2 ]; then
        salir=$n
    else
        #Guarda en el array ar la linea hasta el ";", mensaje que tiene que mostrar en
        el formulario

        ar+=(`echo $n | cut "-d;" -f1`);
        var+=(`echo $n | cut "-d;" -f3`);
        pass+=(`echo $n | cut "-d;" -f2`);
    fi
    let cont=$cont+1
done < txt/$idioma/$1 #archivo que se le pasa al bucle

cont=0
cont2=0
#bucle que recorre el array ar y almacena en opt lo que le pasaremos al dialog
for i in "${ar[@]}";
do
    de=`echo ${rellenar[$cont2]}`;
    let cont=$cont+1;
    if [ ${var[$cont2]} != "n" ]; then
        if [ ${var[$cont2]} == "1" ]; then
            opt+=($i $cont 1 "$home" $cont 25 50 35 ${pass[$cont2]});
        elif [ ${var[$cont2]} == "2" ]; then
            opt+=($i $cont 1 "$shell" $cont 25 50 35 ${pass[$cont2]});
        elif [ ${var[$cont2]} == "3" ]; then
            opt+=($i $cont 1 "N" $cont 25 50 35 ${pass[$cont2]});
        fi
    else
        opt+=($i $cont 1 "" $cont 25 50 35 ${pass[$cont2]});
    fi
    let cont2=$cont2+1;
done

#Crea un formulario pasandole el array opt
dialog --backtitle "BePiRo" --insecure --ok-label "$aceptar" --cancel-label "$salir" --title
"$titulo" \
--mixedform "" 15 50 8 \
"${opt[@]}" \
2>"${salida}"
#Si el dialog devuelve 1 (Boton salir o cancelar), hacemos return 4
if [ $? == 1 ]; then
    return 4
fi

#Guarda lo que el usuario introduce en parametros
while read n ; do
    parametros+=("$n");
done < $salida #archivo que se le pasa al bucle

#Guarda el numero de lineas quitando las lineas en blanco.
nl=`cat /tmp/menu.para.sh | sed '/^\s*$/d' | wc -l`
}

function mostrar_mensaje()
{
    #Busca el codigo en txt/$idioma/codigos.txt para imprimir el mensaje de ese codigo
    msg=`grep -w $1 txt/$idioma/codigos.txt | cut -d: -f2`
    dialog --msgbox "$msg" 6 50
}

```

```
function mostrar_lista()  
{  
    dialog --msgbox "$1" 10 80;  
}
```