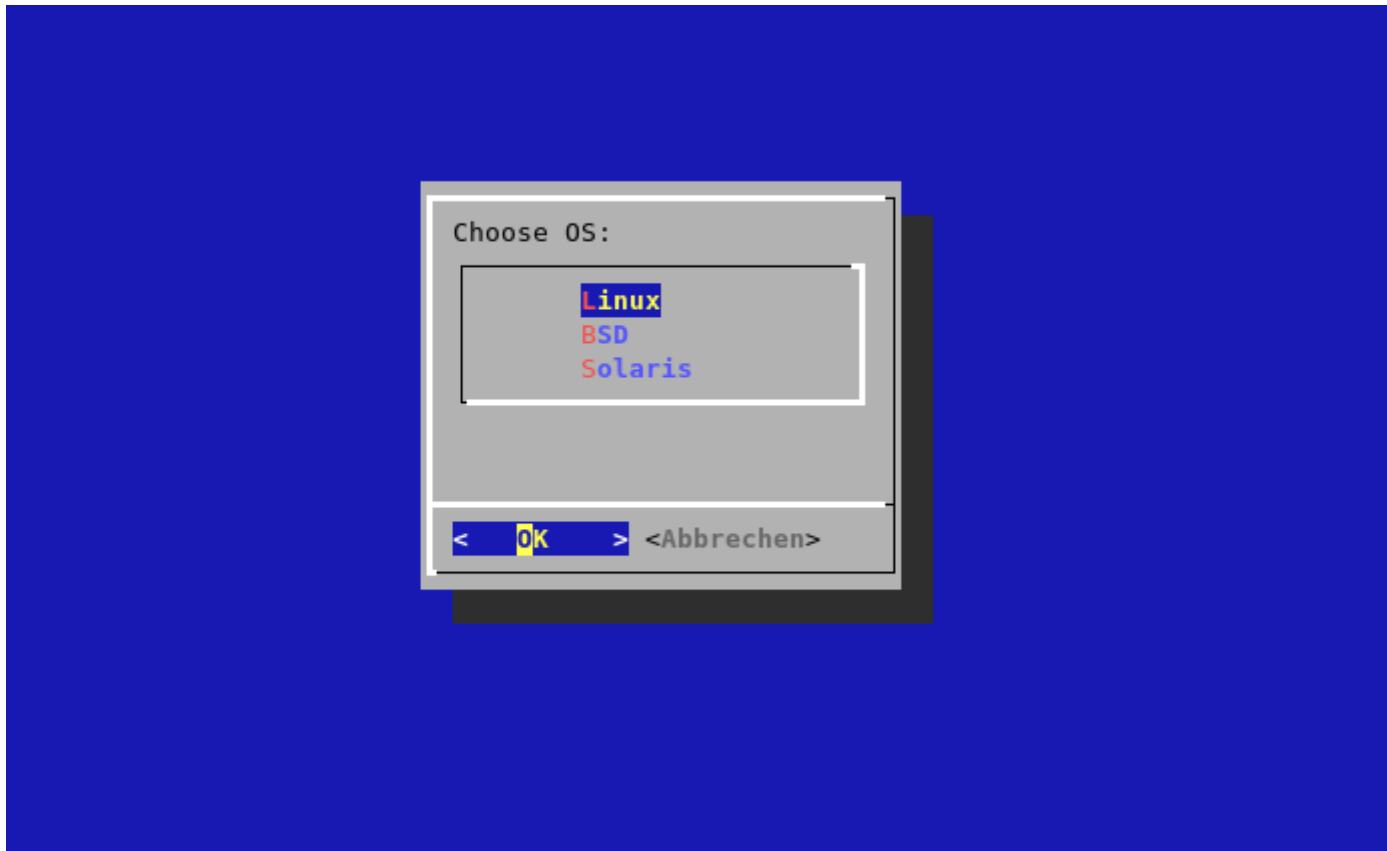


# Shellscripting (Text-User-Interface)



# Dialog

## Dialog & Xdialog:

- Mit **dialog** und **Xdialog** können Sie grafische (bzw. semigrafische) Dialoge in Ihre Shellscripts mit einbauen
- Die Tools dienen zur einfachen Darstellung (halb-)grafischer Dialogfenster auf dem Bildschirm, sodass Sie Benutzerabfragen in Scripts anschaulicher und einfacher gestalten können.
- Die Rückgabewerte der Dialoge entscheiden dann über den weiteren Verlauf des Shellscripts.
- Um **dialog** & **Xdialog** zu benutzen, muss man sie zuerst installieren

# Dialog

## Dialog & Xdialog:

- **Syntax:**

*dialog [Option] [Text] [Höhe] [Breite]*

*Xdialog [Option] [Text] [Höhe] [Breite]*

- **Beispiel:**

*dialog --msgbox „Hallo“ 5 20*

*Xdialog --msgbox „Hallo“ 10 20*

# Dialog

## msgbox:

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

```
dialog --backtitle "msgbox" --title "Message" --msgbox "Hallo" 5 20
```

```
dialog --clear
```

```
clear
```



# Dialog

## yesno box:

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

```
Xdialog --yesno "Want you to quit?" 10 50
```

```
answer=$?
```

```
Xdialog --clear
```

```
if [ $answer -eq 0 ]
```

```
then
```

```
    Xdialog --msgbox "Your answer was YES." 10 50
```

```
else
```

```
    Xdialog --msgbox "Your answer was NO. But I don't care XD" 10 50
```

```
fi
```

```
clear
```

# Dialog

## Filedescriptors:

- Funktionsaufrufe zum Öffnen oder Erstellen von Dateien
- Die Standard-Datenströme **stdin**, **stdout** und **stderr** enthalten dann als Handle die Werte 0, 1 beziehungsweise 2
- Die Werte von 3-9 kann man selber definieren
- Dialog gibt standartmäßig den **stderr**-Wert zurück, aber man kann die Ausgabe umleiten, um an andere Werte ranzukommen
- z.B.: `3>&1 1>&2 2>&3`
- `3>&1` im Terminal erstellt einen Filedescriptor 3 und leitet ihn zu 1 um was **stdout** ist. `1>&2` leitet den file descriptor 1 zu **stderr** um und `2>&3` leitet den Filedescriptor 2 zu 3 was **stdout** ist.

# Dialog

## radiobox:

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

```
drink=$(Xdialog --backtitle "radiobox" --title "Drinks" \
```

```
--radiolist "Select drinks:" 0 0 3 \
```

```
Coca-Cola "" off \
```

```
Fanta "" off \
```

```
Sprite "" off 3>&1 1>&2 2>&3)
```

```
Xdialog --backtitle "radiobox" --title "Drinks" \
```

```
--msgbox "Your choice: $drink" 10 40
```

```
Xdialog --clear
```

```
clear
```

# Dialog

## infobox:

```
#!/bin/bash
dialog --yesno "Want you delete all?" 0 0
answer=$?
dialog --clear
if [ $answer -eq 0 ]
then
    dialog --infobox "This can take a while" 5 50
    sleep 2
    dialog --clear
    dialog --msgbox "Done!" 5 50
fi
clear
```



# Dialog

## inputbox:

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

```
name=$(dialog --inputbox "What is your name?" 0 0 "" \  
3>&1 1>&2 2>&3)
```

```
dialog --clear
```

```
dialog --msgbox "Hello $name, Welcome on $HOSTNAME!" 5 50  
clear
```

# Dialog

## textbox:

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

```
dialog --textbox "$0" 0 0
```

```
dialog --clear
```

```
clear
```

# Dialog

## menubox:

```
#!/bin/bash
function oschoose ()
{
    os=$(dialog --menu "Choose OS:" 0 0 0 \
        "Linux" "" "BSD" "" "Solaris" "" 3>&1 1>&2 2>&3)
}
until [ $answer -eq 0 ]
do
    oschoose
    quit=$?
    if [ $quit -eq 1 ]
    then
        exit
    fi
    dialog --yesno "Confirm your choose: $os" 0 0
    answer=$?
    dialog --clear
done
dialog --clear
clear
```

# Dialog

## checkbox:

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

```
pizza=$(dialog --checklist "Pizza with ..." 0 0 4 \
```

```
mozzarella "" on \
```

```
salami "" off \
```

```
ham "" off \
```

```
tuna "" off 3>&1 1>&2 2>&3)
```

```
dialog --clear
```

```
dialog --title "Pizza" --msgbox "Your order: Pizza with $pizza" 5 50
```

```
dialog --clear
```

```
clear
```

# Dialog

## gauge:

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

```
for i in $(seq 0 10 100)
```

```
do
```

```
    sleep 1
```

```
    echo $i | dialog --gauge "Please wait" 10 70 0
```

```
done
```

# Dialog

## gauge:

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

```
DIALOG=dialog
```

```
(echo "10" ; sleep 1
```

```
echo "XXX" ; echo "All data will be saved"; echo "XXX"
```

```
echo "20" ; sleep 1
```

```
echo "50" ; sleep 1
```

```
echo "XXX" ; echo "All data will be archived"; echo "XXX"
```

```
echo "75" ; sleep 1
```

```
echo "XXX" ; echo "Data will be uploaded"; echo "XXX"
```

```
echo "100" ; sleep 3) |
```

```
$DIALOG --title "Progress" --gauge "Start backup script" 8 30
```

```
$DIALOG --clear
```

```
$DIALOG --msgbox "Job completed" 5 20
```

```
$DIALOG --clear
```

```
clear
```

# Dialog

## gauge:

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

```
counter=0
```

```
(while [ $counter -le 100 ]
```

```
do
```

```
cat <<EOF
```

```
XXX
```

```
$counter
```

```
Disk copy /dev/dvd to /home/data ($counter%):
```

```
XXX
```

```
EOF
```

```
((counter+=10))
```

```
sleep 1
```

```
done) | dialog --title "File Copy" --gauge "Please wait" 7 70
```