



UPMC
SORBONNE UNIVERSITÉS



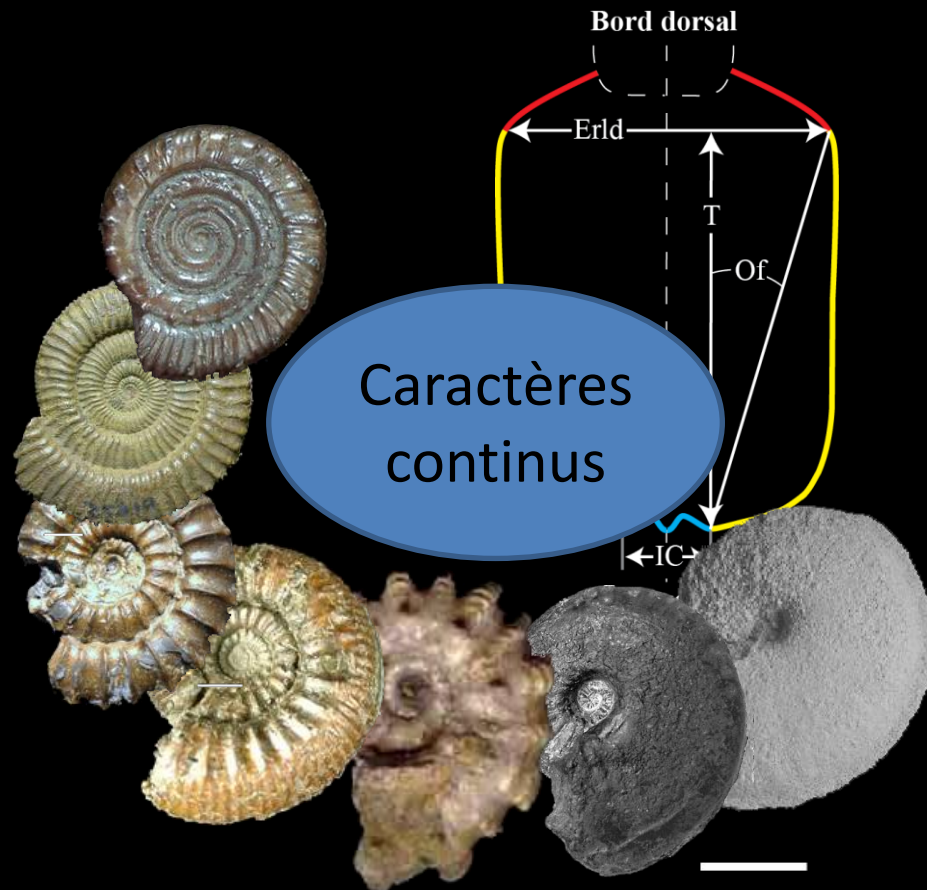
Package Tcl/tk : Comment un Rophobe peut utiliser un script R sans angoisse

Jérémie Bardin
MNHN / UPMC
jbardin@mnhn.fr

Analyse cladistique

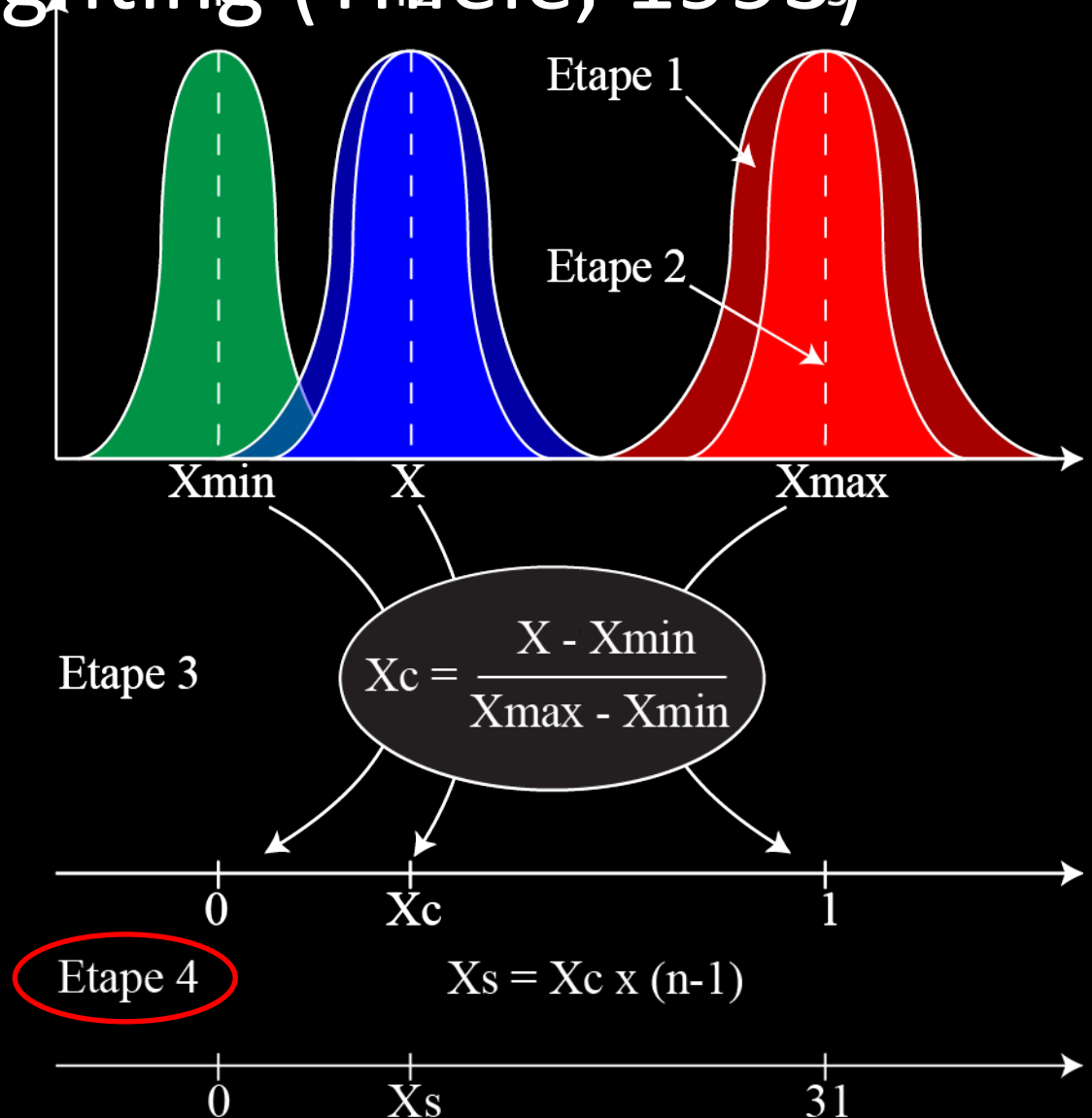


Analyse cladistique



Discrétisation des caractères continus : le Gap Weighting (Thiele, 1993)

1. Standardisation des variances
2. Calcul des moyennes
3. Standardisation entre 0 et 1
4. Multiplication par $n - 1$
= Utilisation du maximum
d'états codables



La fonction GapW



Jeu de données

GAP WEIGHTING

Matrices avec N_s états utilisés pour discrétiser les caractères continus

$N_s = 2$ - - - - - $1\ 001$ - - - $65\ 001$

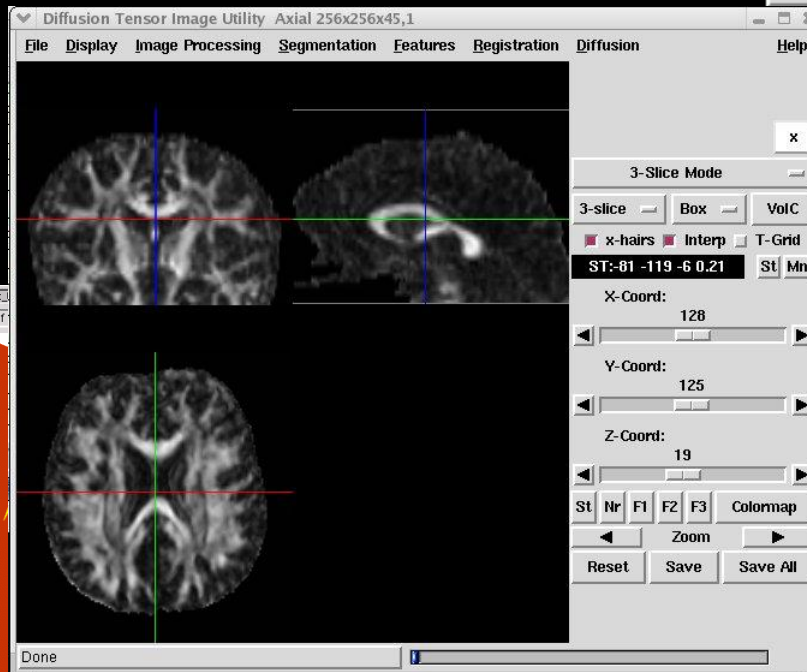
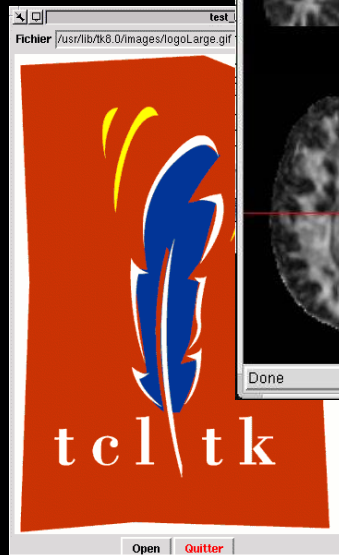


ANALYSES EXHAUSTIVES (PAUP & TNT)

Arbres phylogénétiques

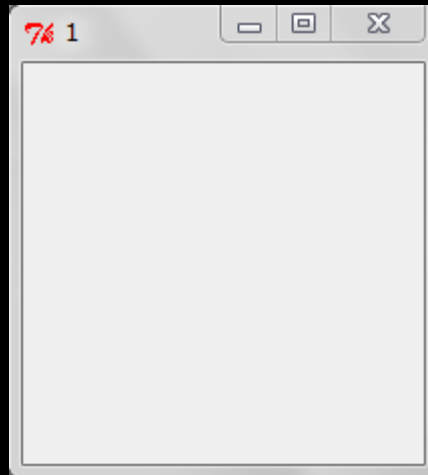
Tcl/tk : généralités

- Tool Command Language
- Tk



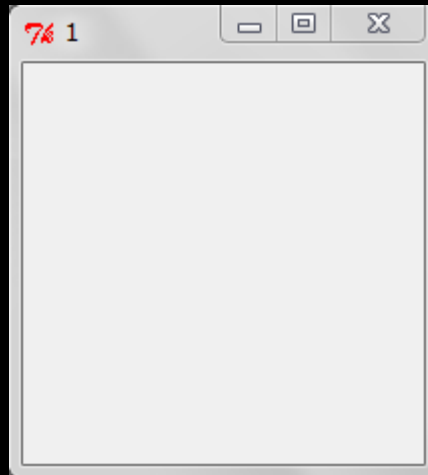
Le principe de base

- Créer une fenêtre
- `tt<-tktoplevel()`



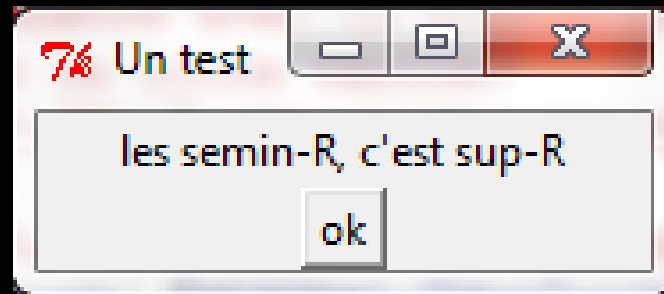
Le principe de base

- Créer une fenêtre
- `tt<-tktoplevel()`
- Créer des objets
- `label<-tklabel(tt, text="les semin-R, c'est sup-R")`
- `bouton<-tkbutton(tt, text="ok")`



Le principe de base

- Créer une fenêtre
 - `tt<-tktoplevel()`
- Créer des objets
 - `label<-tklabel(tt, text="les semin-R, c'est sup-R")`
 - `bouton<-tkbutton(tt, text="ok")`
- Afficher les objets
 - `tkpack(label, bouton) OU tkgrid(label, bouton)`

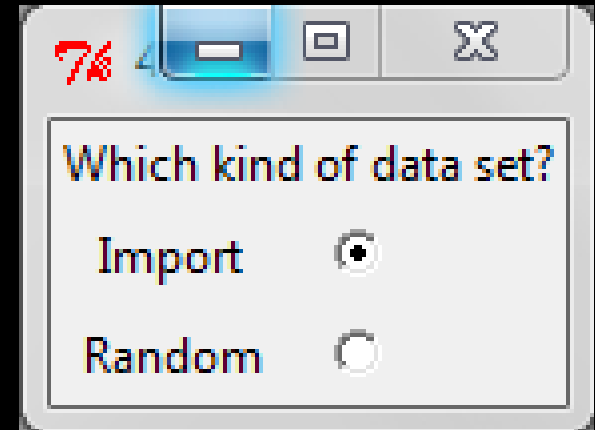


Le principe de base

- Créer une fenêtre
 - `tt<-tktoplevel()`
- Créer des objets
 - `label<-tklabel(tt, text="les semin-R, c'est super")`
 - `bouton<-tkbutton(tt, text="ok")`
- Afficher les objets
 - `tkpack(label, bouton) OU tkgrid(label, bouton)`
- Détruire la fenêtre
 - `tkdestroy (tt)`

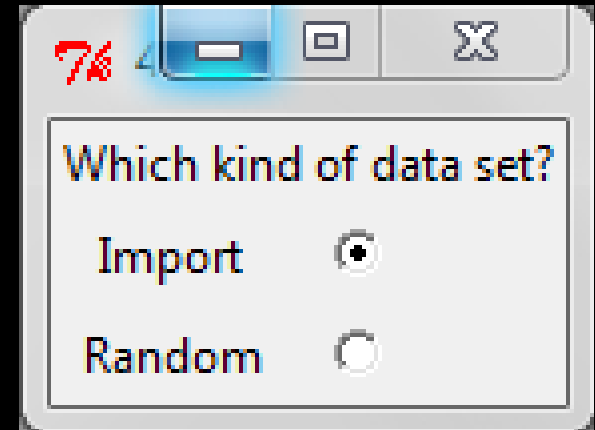
Les variables

- Créer une fenêtre
- Créer des objets
- `tt <- tktoplevel()`
- `rb1 <- tkradiobutton(tt)`
- `rb2 <- tkradiobutton(tt)`

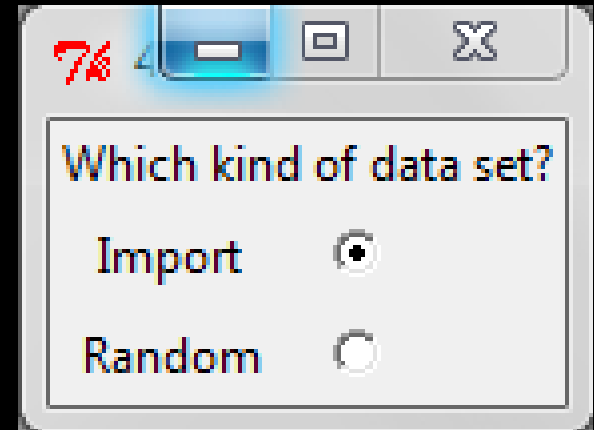


Les variables

- Créer une fenêtre
- Créer des objets
- `tt <- tkoplevel()`
- `rb1 <- tkradiobutton(tt)`
- `rb2 <- tkradiobutton(tt)`
- `rbValue <- tclVar("import")`
- `tkconfigure(rb1,variable=rbValue,value="import")`
- `tkconfigure(rb2,variable=rbValue,value="random")`



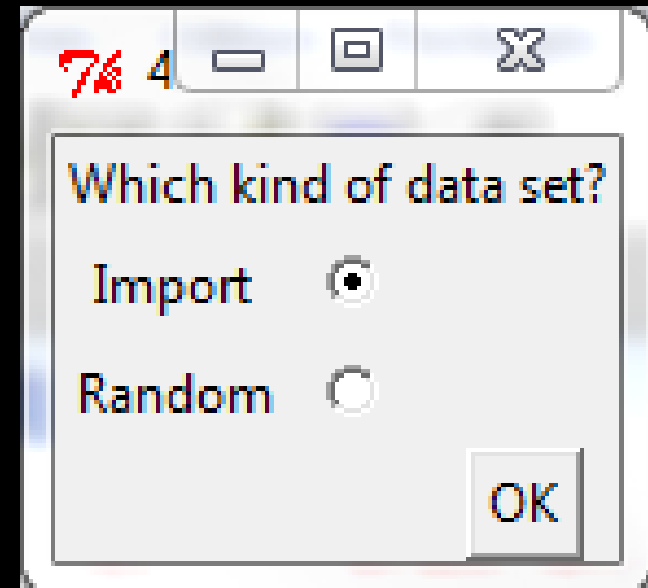
Les variables



- Créer une fenêtre
 - Créer des objets
 - Afficher les objets
 - Détruire la fenêtre
- `tt <- tkoplevel()`
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
 - `rbValue <- tclVar("import")`
 - `tkconfigure(rb1,variable=rbValue,value="import")`
 - `tkconfigure(rb2,variable=rbValue,value="random")`
 - `tkgrid(tklabel(tt,text="Which kind of data set?"),`
`columnspan=5)`
 - `tkgrid(tklabel(tt,text="Import"),rb1)`
 - `tkgrid(tklabel(tt,text="Random"),rb2)`
 - `Data <- tclvalue(rbValue)`
 - `tkdestroy (tt)`

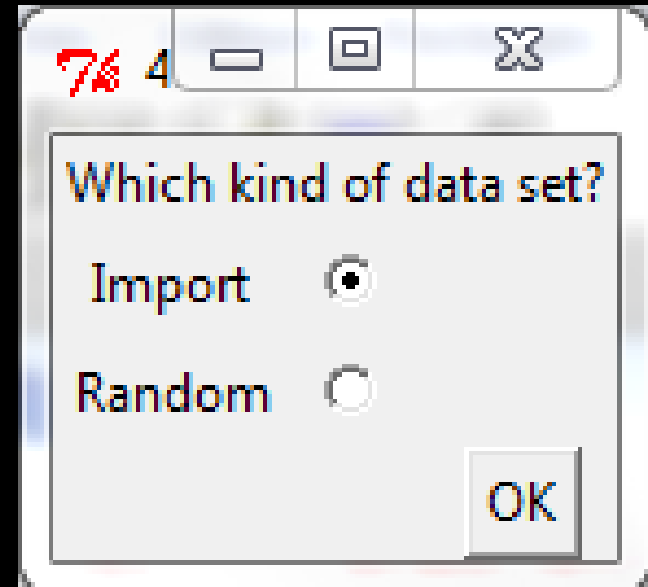
Les variables

- `tt <- tkoplevel()`
- `rb1 <- tkradiobutton(tt)`
- `rb2 <- tkradiobutton(tt)`
- `rbValue <- tclVar("import")`
- `tkconfigure(rb1,variable=rbValue,value="import")`
- `tkconfigure(rb2,variable=rbValue,value="random")`
- `tkgrid(tklabel(tt,text="Which kind of data set?"),
columnspan=5)`
- `tkgrid(tklabel(tt,text="Import"),rb1)`
- `tkgrid(tklabel(tt,text="Random"),rb2)`



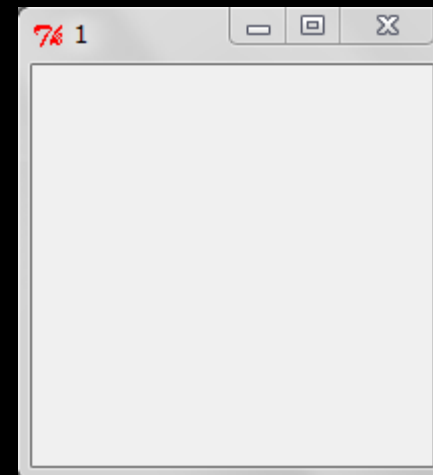
Les variables

- `tt <- tkoplevel()`
- `rb1 <- tkradiobutton(tt)`
- `rb2 <- tkradiobutton(tt)`
- `rbValue <- tclVar("import")`
- `tkconfigure(rb1,variable=rbValue,value="import")`
- `tkconfigure(rb2,variable=rbValue,value="random")`
- `tkgrid(tklabel(tt,text="Which kind of data set?"),
columnspan=5)`
- `tkgrid(tklabel(tt,text="Import"),rb1)`
- `tkgrid(tklabel(tt,text="Random"),rb2)`
- `done <- tclVar(0)`
- `OnOK <- function()`
 - {
 - `Data <- as.character(tclvalue(rbValue))`
 - `tclvalue(done)<-1`
 - `tkdestroy(tt)`
 - }
- `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- `tkgrid(OK.but,row=4, column=3)`
- `tkwait.variable(done)`



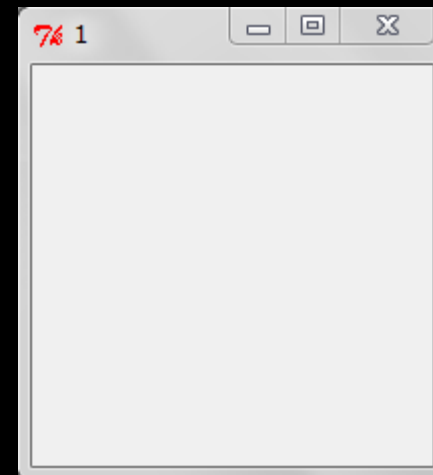
Création de fenêtre

- Créer
- `tt<-tktoplevel()`
- Redimensionner
- `tkwm.minsize(tt, 200, 100)`
- `tkwm.resizable(tt, FALSE, TRUE)`
- `tkwm.geometry(tt, "800x600")`
- Subdiviser
- `one <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `two <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `three <- tkframe(tt, borderwidth = 2,relief = "sunken")`
- Style
- `tkconfigure(tt, borderwidth = 20, width="700",color= "red")`
- `titlepolice <- tkfont.create(size=10, weight="bold")`
- `title1<-tklabel(tt,,text=" Please enter the number of : ")`
- `tkconfigure(title1,font=titlepolice)`
- `tkconfigure(tt, bg="yellow")`

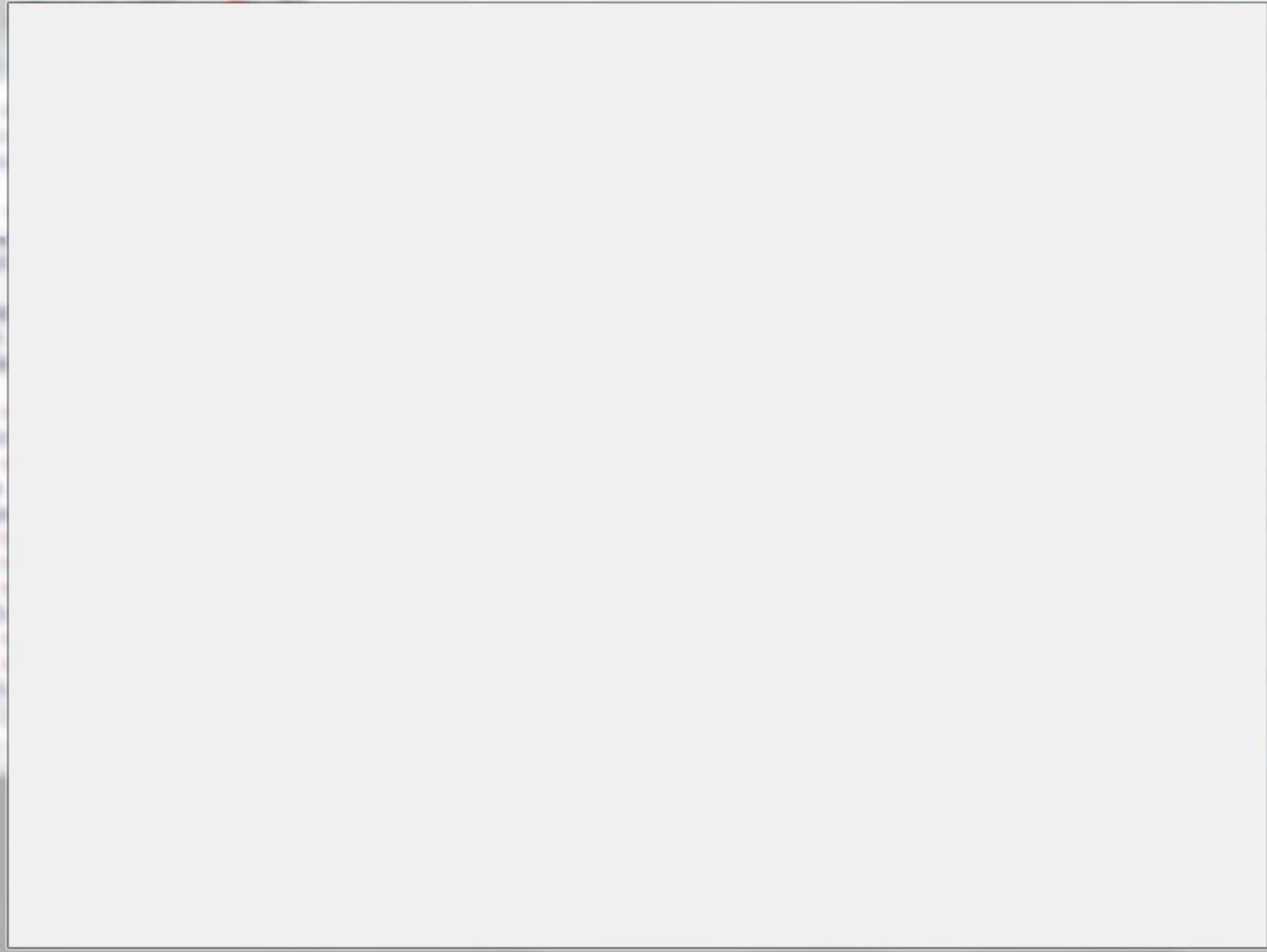


Création de fenêtre

- Créer
- `tt<-tktoplevel()`
- Redimensionner
- `tkwm.minsize(tt, 200, 100)`
- `tkwm.resizable(tt, FALSE, TRUE)`
- `tkwm.geometry(tt, "800x600")`
- Subdiviser
- `one <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `two <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `three <- tkframe(tt, borderwidth = 2,relief = "sunken")`
- Style
- `tkconfigure(tt, borderwidth = 20, width="700",color= "red")`
- `titlepolice <- tkfont.create(size=10, weight="bold")`
- `title1<-tklabel(tt,,text=" Please enter the number of : ")`
- `tkconfigure(title1,font=titlepolice)`
- `tkconfigure(tt, bg="yellow")`

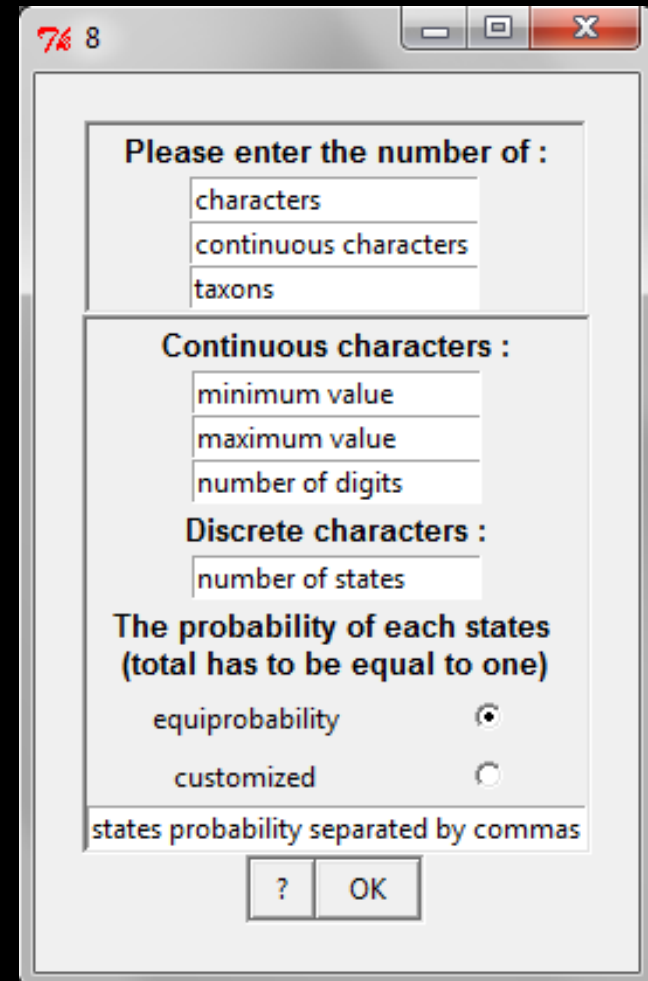


- Cré
- tt<-
- Rec
- tkw
- tkw
- tkw
- Sub
- one
- 2, re
- two
- 2, re
- thre
- "su
- Styl
- tkcc
- title
- title
- of :
- tkcc
- tkcc



Création de fenêtre

- Créer
- `tt<-tkoplevel()`
- Redimensionner
- `tkwm.minsize(tt, 200, 100)`
- `tkwm.resizable(tt, FALSE, TRUE)`
- `tkwm.geometry(tt, "800x600")`
- Subdiviser
- `one <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `two <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `three <- tkframe(tt, borderwidth = 2,relief = "sunken")`
- Style
- `tkconfigure(tt, borderwidth = 20, width="700",color= "red")`
- `titlepolice <- ttfont.create(size=10, weight="bold")`
- `title1<-tklabel(tt,,text=" Please enter the number of : ")`
- `tkconfigure(title1,font=titlepolice)`
- `tkconfigure(tt, bg="yellow")`



Création de fenêtre

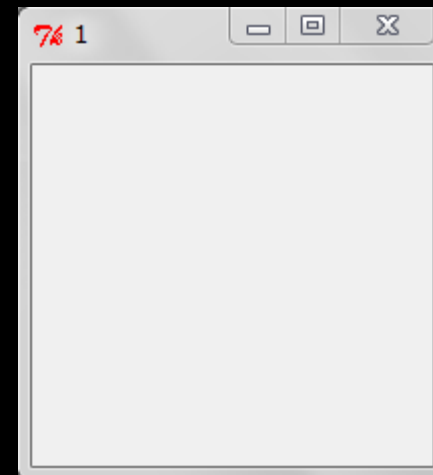
- Créer
- `tt<-tktoplevel()`

- Redimensionner
- `tkwm.minsize(tt, 200, 100)`
- `tkwm.resizable(tt, FALSE, TRUE)`
- `tkwm.geometry(tt, "800x600")`

- Subdiviser
- `one <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `two <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `three <- tkframe(tt, borderwidth = 2,relief = "sunken")`

- Style
- `tkconfigure(tt, borderwidth = 20, width="700",color= "red")`

- `titlepolice <- ttfont.create(size=10, weight="bold")`
- `title1<-tklabel(tt,,text=" Please enter the number of : ")`
- `tkconfigure(title1,font=titlepolice)`
- `tkconfigure(tt, bg="yellow")`



Création de fenêtre

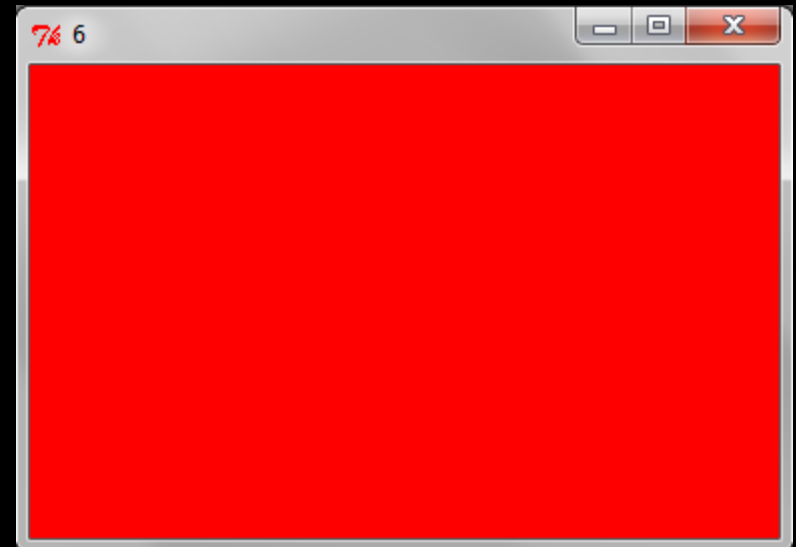
- Créer
- `tt<-tktoplevel()`

- Redimensionner
- `tkwm.minsize(tt, 200, 100)`
- `tkwm.resizable(tt, FALSE, TRUE)`
- `tkwm.geometry(tt, "800x600")`

- Subdiviser
- `one <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `two <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
- `three <- tkframe(tt, borderwidth = 2,relief = "sunken")`

- Style
- `tkconfigure(tt, borderwidth = 20, width="700",color= "red")`

- `titlepolice <- ttfont.create(size=10, weight="bold")`
- `title1<-tklabel(tt,,text=" Please enter the number of : ")`
- `tkconfigure(title1,font=titlepolice)`
- `tkconfigure(tt, bg="yellow")`

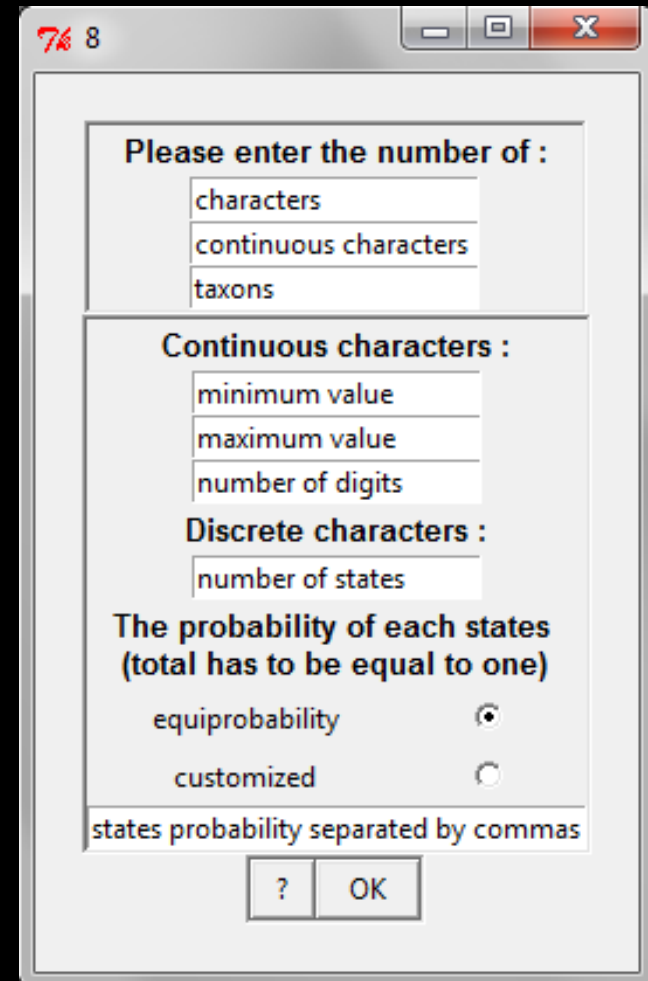


Création de fenêtre

- Créer
 - `tt<-tkoplevel()`

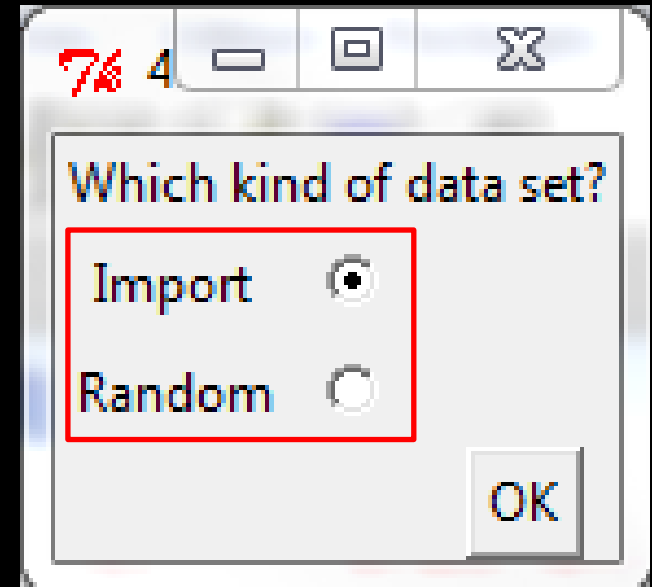
 - Redimensionner
 - `tkwm.minsize(tt, 200, 100)`
 - `tkwm.resizable(tt, FALSE, TRUE)`
 - `tkwm.geometry(tt, "800x600")`

 - Subdiviser
 - `one <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
 - `two <- tkframe(tt,width = "70", borderwidth = 2,relief = "sunken")`
 - `three <- tkframe(tt, borderwidth = 2,relief = "sunken")`
- Style
 - `tkconfigure(tt, borderwidth = 20, width="700",color= "red")`
 - `titlepolice <- ttfont.create(size=10, weight="bold")`
 - `title1<-tklabel(tt,,text=" Please enter the number of : ")`
 - `tkconfigure(title1,font=titlepolice)`
 - `tkconfigure(tt, bg="yellow")`



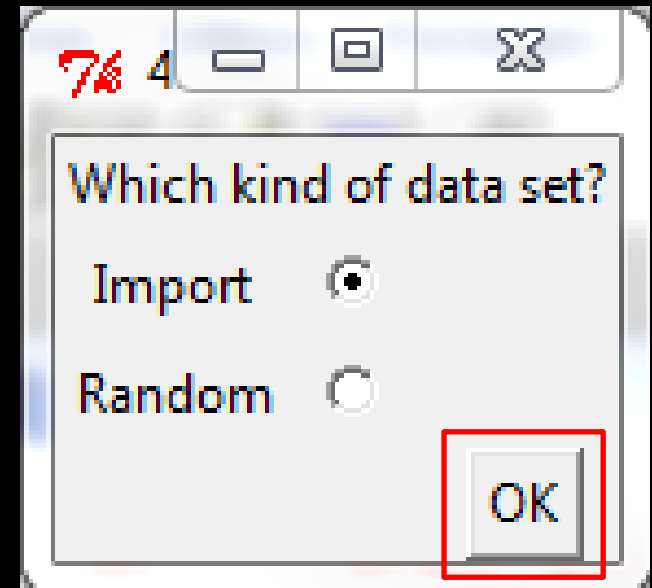
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



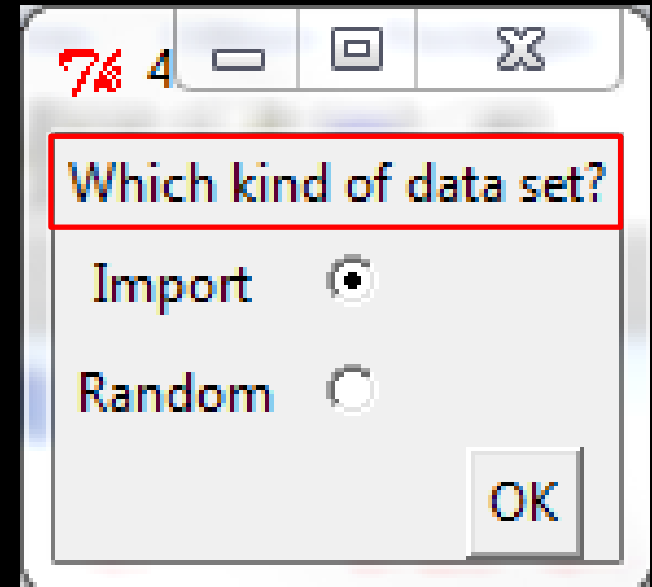
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



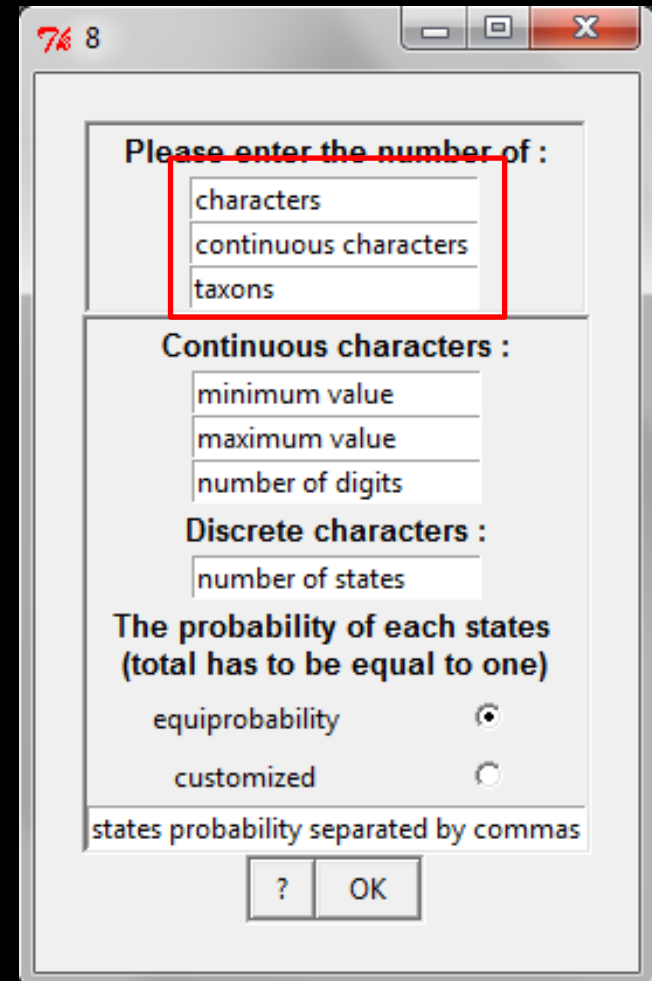
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



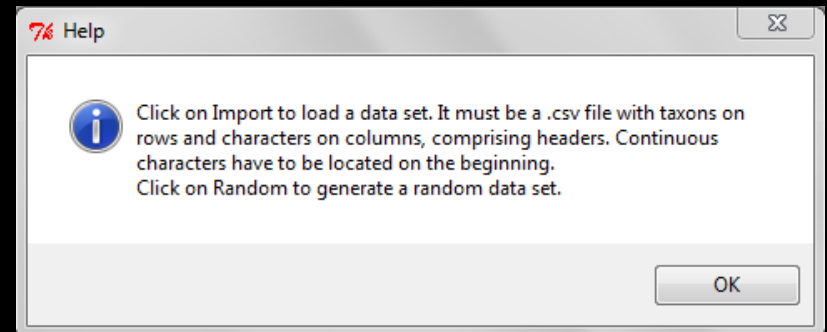
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



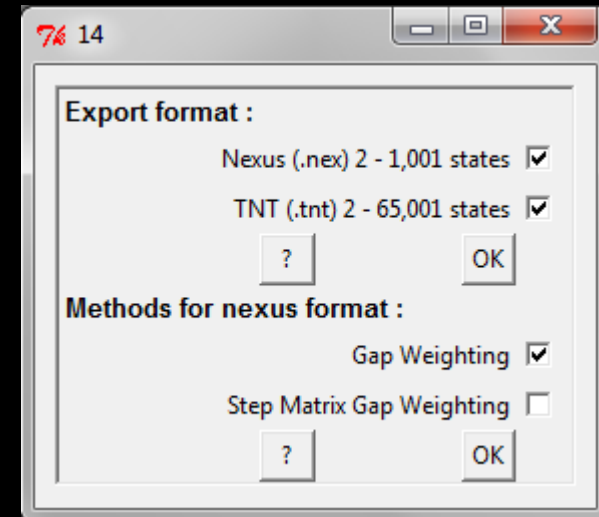
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



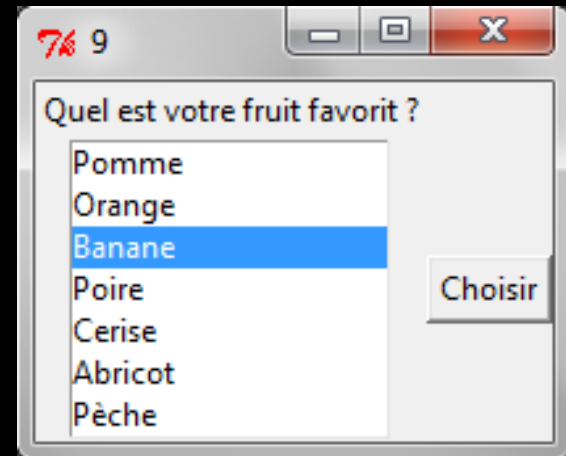
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



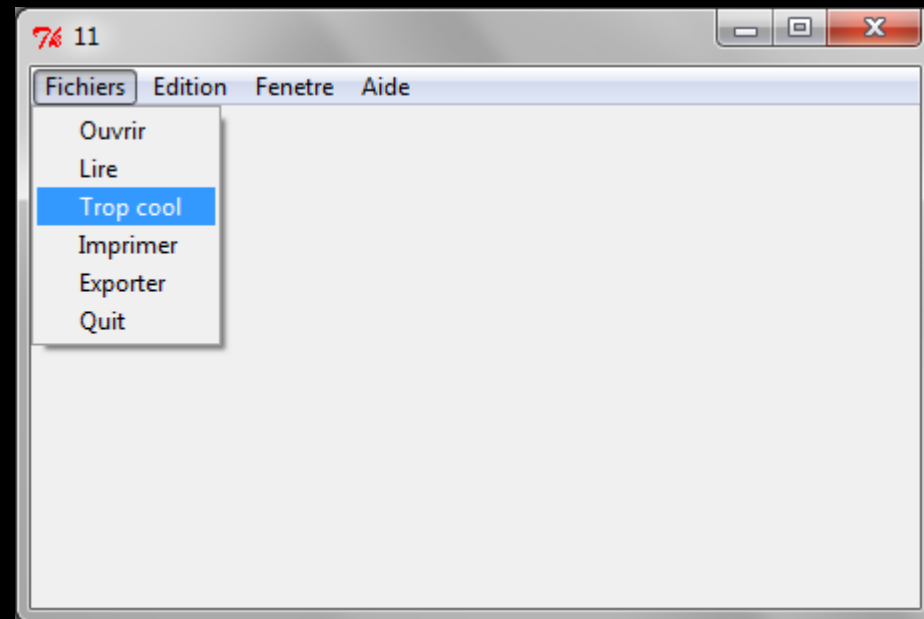
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



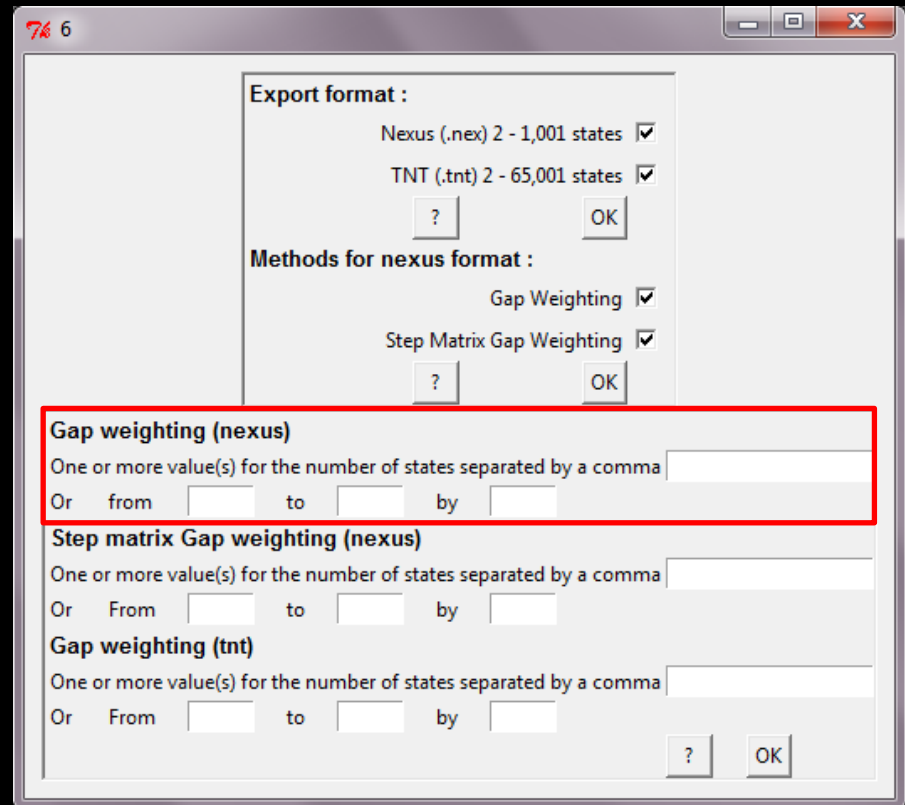
Création d'objets

- Bouton radio
 - `rb1 <- tkradiobutton(tt)`
 - `rb2 <- tkradiobutton(tt)`
- Bouton
 - `OK.but <- tkbutton(tt,text="OK",command=OnOK)`
- Texte
 - `lablistvectproba<-tklabel(two, text="The probability of each states (total has to be equal to one)")`
- Zone de texte
 - `entry.Ncont <-tkentry(one,width="20",textvariable=Ncont)`
- Fenêtre avec un message
 - `ReturnVal <- tkmessageBox(title = "Help", message = Help5, icon = "info", type = "ok")`
- Case à cocher
 - `Bnex <- tkcheckboxbutton(one)`
- Liste
 - `maliste<-tklistbox(tt, height=7, selectmode="single", background="white")`
- Menu
 - `topMenu <- tkmenu(tt)`
 - `fileMenu <- tkmenu(topMenu, tearoff=FALSE)`
 - `tkadd(topMenu,"cascade", label="Fichiers", menu=fileMenu)`



Afficher les objets : grid manager

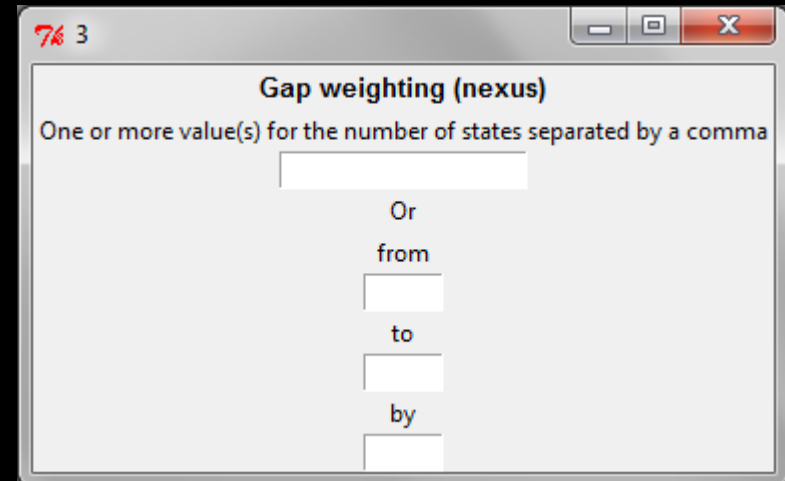
- `tt <- tktoplevel()`
 - `titlepolice <- tkfont.create(size=10, weight="bold")`
 - `titlegw <- tklabel(tt, text="Gap weighting (nexus)")`
 - `tkconfigure(titlegw, font=titlepolice)`
 - `lablistgw <- tklabel(tt, text="One or more value(s) for the number of states separated by a comma")`
 - `OR <- tklabel(tt, text="Or")`
 - `labseqgwF <- tklabel(tt, text="from")`
 - `labseqgwT <- tklabel(tt, text="to")`
 - `labseqgwB <- tklabel(tt, text="by")`
 - `listgw <- tkentry(tt, width=20)`
 - `gwF <- tkentry(tt, width=6)`
 - `gwT <- tkentry(tt, width=6)`
 - `gwB <- tkentry(tt, width=6)`
-
- `tkgrid(titlegw)`
 - `tkgrid(lablistgw)`
 - `tkgrid(OR)`
 - `tkgrid(labseqgwF)`
 - `tkgrid(gwF)`
 - `tkgrid(labseqgwT)`
 - `tkgrid(gwT)`
 - `tkgrid(labseqgwB)`
 - `tkgrid(gwB)`



Afficher les objets : grid manager

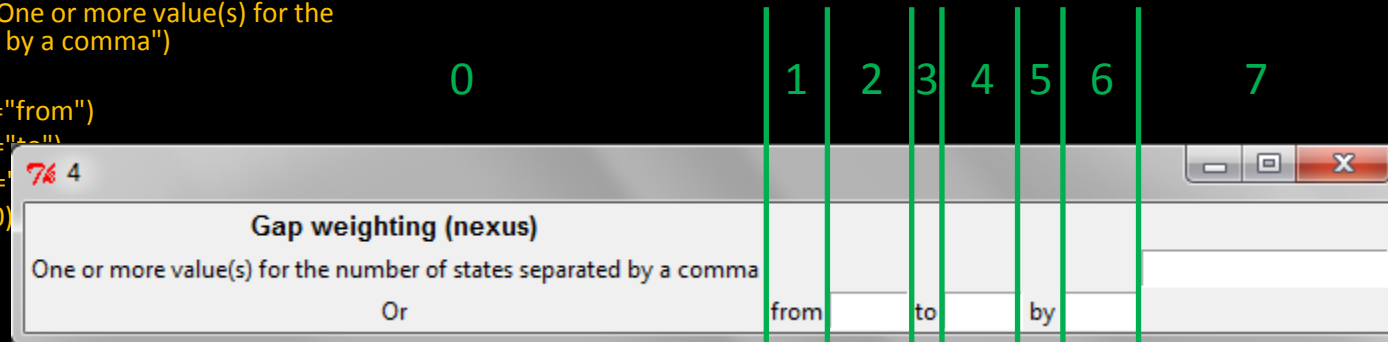
- `tt <- tktoplevel()`
- `titlepolice <- tkfont.create(size=10, weight="bold")`
- `titlegw<-tklabel(tt, text="Gap weighting (nexus)")`
- `tkconfigure(titlegw,font=titlepolice)`
- `lablistgw<-tklabel(tt, text="One or more value(s) for the number of states separated by a comma")`
- `OR<-tklabel(tt,text="Or")`
- `labseqgwF<-tklabel(tt, text="from")`
- `labseqgwT<-tklabel(tt, text="to")`
- `labseqgwB<-tklabel(tt, text="by")`
- `listgw<-tkentry(tt, width=20)`
- `gwF<-tkentry(tt, width=6)`
- `gwT<-tkentry(tt, width=6)`
- `gwB<-tkentry(tt, width=6)`

- `tkgrid(titlegw)`
- `tkgrid(lablistgw)`
- `tkgrid(listgw)`
- `tkgrid(OR)`
- `tkgrid(labseqgwF)`
- `tkgrid(gwF)`
- `tkgrid(labseqgwT)`
- `tkgrid(gwT)`
- `tkgrid(labseqgwB)`
- `tkgrid(gwB)`



Afficher les objets : grid manager

- `tt <- tkoplevel()`
- `titlepolice <- tkfont.create(size=10, weight="bold")`
- `titlegw<-tklabel(tt, text="Gap weighting (nexus)")`
- `tkconfigure(titlegw,font=titlepolice)`
- `lablistgw<-tklabel(tt, text="One or more value(s) for the number of states separated by a comma")`
- `OR<-tklabel(tt,text="Or")`
- `labseqgwF<-tklabel(tt, text="from")`
- `labseqgwT<-tklabel(tt, text="to")`
- `labseqgwB<-tklabel(tt, text="by")`
- `listgw<-tkentry(tt, width=20)`
- `gwF<-tkentry(tt, width=6)`
- `gwT<-tkentry(tt, width=6)`
- `gwB<-tkentry(tt, width=6)`

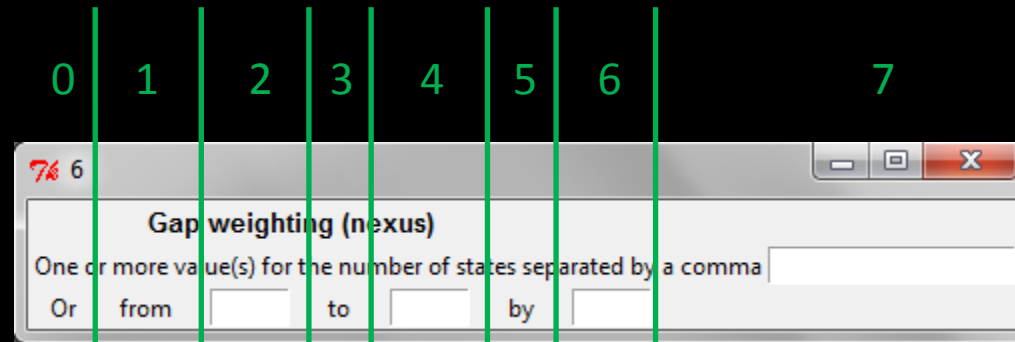


- `tkgrid(titlegw, row=0)`
- `tkgrid(lablistgw, row=1, column=0)`
- `tkgrid(listgw, row=1, column=7)`
- `tkgrid(OR,row=2, column=0)`
- `tkgrid(labseqgwF, row=2, column=1)`
- `tkgrid(gwF, row=2, column=2)`
- `tkgrid(labseqgwT, row=2, column=3)`
- `tkgrid(gwT, row=2, column=4)`
- `tkgrid(labseqgwB, row=2, column=5)`
- `tkgrid(gwB, row=2, column=6)`

Afficher les objets : grid manager

- `tt <- tkoplevel()`
- `titlepolice <- tkfont.create(size=10, weight="bold")`
- `titlegw<-tklabel(tt, text="Gap weighting (nexus)")`
- `tkconfigure(titlegw,font=titlepolice)`
- `lablistgw<-tklabel(tt, text="One or more value(s) for the number of states separated by a comma")`
- `OR<-tklabel(tt,text="Or")`
- `labseqgwF<-tklabel(tt, text="from")`
- `labseqgwT<-tklabel(tt, text="to")`
- `labseqgwB<-tklabel(tt, text="by")`
- `listgw<-tkentry(tt, width=20)`
- `gwF<-tkentry(tt, width=6)`
- `gwT<-tkentry(tt, width=6)`
- `gwB<-tkentry(tt, width=6)`

- `tkgrid(titlegw, row=0, columnspan=6)`
- `tkgrid(lablistgw, row=1, column=0, columnspan=10)`
- `tkgrid(listgw, row=1, column=11)`
- `tkgrid(OR,row=2, column=0)`
- `tkgrid(labseqgwF, row=2, column=1)`
- `tkgrid(gwF, row=2, column=2)`
- `tkgrid(labseqgwT, row=2, column=3)`
- `tkgrid(gwT, row=2, column=4)`
- `tkgrid(labseqgwB, row=2, column=5)`
- `tkgrid(gwB, row=2, column=6)`



Afficher les objets : grid manager

- `tt <- tktoplevel()`
- `titlepolice <- tkfont.create(size=10, weight="bold")`
- `titlegw<-tklabel(tt, text="Gap weighting (nexus)")`
- `tkconfigure(titlegw,font=titlepolice)`
- `lablistgw<-tklabel(tt, text="One or more value(s) for the number of states separated by a comma")`
- `OR<-tklabel(tt,text="Or")`
- `labseqgwF<-tklabel(tt, text="from")`
- `labseqgwT<-tklabel(tt, text="to")`
- `labseqgwB<-tklabel(tt, text="by")`
- `listgw<-tkentry(tt, width=20)`
- `gwF<-tkentry(tt, width=6)`
- `gwT<-tkentry(tt, width=6)`
- `gwB<-tkentry(tt, width=6)`

- `tkgrid(titlegw, row=0, columnspan=6,sticky="w")`
- `tkgrid(lablistgw, row=1, column=0, sticky="w",columnspan=10)`
- `tkgrid(listgw, row=1, column=11, sticky="w")`
- `tkgrid(OR,row=2, column=0, sticky="w")`
- `tkgrid(labseqgwF, row=2, column=1, sticky="w")`
- `tkgrid(gwF, row=2, column=2, sticky="w")`
- `tkgrid(labseqgwT, row=2, column=3, sticky="w")`
- `tkgrid(gwT, row=2, column=4, sticky="w")`
- `tkgrid(labseqgwB, row=2, column=5, sticky="w")`
- `tkgrid(gwB, row=2, column=6, sticky="w")`

