Tkinter funkciju dokumentācija

## Window

Izveido galveno (Tk()) vai papildu (Toplevel()) logu, kas kalpo kā pamats visiem citiem GUI elementiem. Loga izveide nodrošina galveno notikumu cilpu (event loop), kas uztur aplikācijas darbību un reaģē uz lietotāja darbībām.

* Klase/funkcija: Tk() vai Toplevel()
* Svarīgākie parametri:
* screenName – izvēles iespēja noteikt ekrāna nosaukumu vairākmonitoru vidē
* baseName – programmas bāzes nosaukums, kas var parādīties loga nosaukumā
* className – loga klase, ko operētājsistēma izmanto loga dekorācijās
* useTk – vai izmantot Tk režīmu (boolean)
* Galvenās metodes:
* title(text) – iestata loga virsrakstu, kas parādās loga augšējā joslā
* geometry("WxH+X+Y") – definē loga sākotnējos izmērus un pozīciju ekrānā
* resizable(x, y) – nosaka, vai logs ir maināma lieluma horizontāli un/vai vertikāli
* mainloop() – startē notikumu cilpu, uzturot logu aktīvu un reaģējot uz lietotāja darbībām

## Grid

Ģeometrijas pārvaldnieks, kas izvieto widgetus režģveida struktūrā pēc rindu un kolonnu indeksiem. Atļauj precīzi kontrolēt atstarpes, apvienot šūnas un piesaistīt widgetus režģa šūnu malām.

* Funkcija: widget.grid(...)
* Svarīgākie parametri:
* row (int) – rindas indekss (no 0)
* column (int) – kolonnas indekss (no 0)
* padx, pady (int) – horizontālās un vertikālās atstarpes pikseļos
* sticky (str) – piesaiste malām (N, S, E, W vai to kombinācijas)
* rowspan, columnspan (int) – cik rindu vai kolonnu šūnu apvienot

## Button

Interaktīvs GUI komponents, kas reaģē uz lietotāja klikšķiem, izsaucot norādīto funkciju. Pogas stāvokli var konfigurēt un pielāgot izskatu ar tekstu, attēliem un krāsām.

* Klase/funkcija: Button(master, ...)
* Svarīgākie parametri:
* text (str) – uz pogas redzamais teksts
* command (callable) – funkcija, ko izsauc klikšķa brīdī
* width, height (int) – pogas izmēri rakstzīmju vienībās
* bg, fg (str) – fona un teksta krāsa
* state (str) – pogas stāvoklis (NORMAL, DISABLED, ACTIVE)

## Label

Neklikšķināms teksts vai attēls GUI logā, kas kalpo kā statisks paziņojums vai vietturis. Var dinamiski atjaunināt ar metodi config() vai saistīt StringVar mainīgo.

* Klase/funkcija: Label(master, ...)
* Svarīgākie parametri:
* text (str) – parādāmais teksts
* font (tuple) – fonta nosaukums un izmērs
* fg, bg (str) – teksta un fona krāsa
* image – attēla objekts (PhotoImage vai BitmapImage)
* anchor (str) – teksta novietojums šūnā (N, S, E, W, CENTER)

## Listbox

Widget, kas attēlo elementu sarakstu un ļauj lietotājam atlasīt vienu vai vairākus ierakstus. Var pievienot ritjoslu, lai pārvaldītu garākus sarakstus.

* Klase/funkcija: Listbox(master, ...)
* Svarīgākie parametri:
* height (int) – redzamās rindu skaits vienlaikus
* selectmode (str) – atlases režīms (SINGLE, BROWSE, MULTIPLE, EXTENDED)
* activestyle (str) – aktīvās rindas vizuālais izcelšanas veids (dotbox, underline, none)
* exportselection (bool) – vai atlasītais ieraksts tiek kopēts starpliktuvē

## Entry

Vienas rindas teksta ievades lauks, kur lietotājs var rakstīt un rediģēt tekstu. Var slēpt ievades simbolus (piem., paroles) un saistīt ar StringVar, lai sekotu izmaiņām.

* Klase/funkcija: Entry(master, ...)
* Svarīgākie parametri:
* width (int) – lauka platums rakstzīmēs
* show (str) – aizvietojošs simbols privātai ievadei (parole)
* font (tuple) – fonta nosaukums un izmērs
* textvariable – StringVar objekts, saistot lauka saturu ar mainīgo
* state (str) – stāvoklis (NORMAL, DISABLED, READONLY)