

Ohjelmoinnin peruskurssi Y1

CS-A1111

Graafinen käyttöliittymä: kuvioiden piirtäminen ja valmiiden kuvatiedostojen lisääminen graafiseen käyttöliittymään

Oppimistavoitteet: tämän luennon jälkeen

- ▶ Osaat luokan Canvas ja sen metodien avulla ohjelmoida ohjelman piirtämään käyttöliittymään erilaisia kuvioita.
- ▶ Osaat käyttää ohjelmassa valmiita kuvatiedostoja.

Piirtäminen

- ▶ Luokan `Canvas` avulla voidaan luoda piirtoalusta, jolle voidaan piirtää erilaisia kuvioita, viivoja, tekstejä jne.
- ▶ Näin voidaan saada ohjelma tulostamaan erilaisia kaavioita tai kuvaajia.
- ▶ Piirtoalustan koordinaatteja lasketaan pikseleiden avulla. Vasemman yläkulman koordinaatit ovat $(0, 0)$
- ▶ Joitakin luokan `Canvas` metodeita. Mahdollisia parametreja on paljon erilaisia, mutta ensimmäiset parametrit kertovat ne koordinaatit, mihin komponentti piirretään:
 - ▶ `create_line` piirtää suoran viivan
 - ▶ `create_polygon` piirtää murtoviivan
 - ▶ `create_rectangle` piirtää suorakulmion
 - ▶ `create_oval` piirtää ympyrän tai ellipsin
 - ▶ `create_text` piirtää tekstin

Esimerkki piirtävästä ohjelmasta

```
from tkinter import *

class Piirtoikkuna:
    def __init__(self):
        self.__paaikkuna = Tk()
        self.__paaikkuna.title("Piirtoesimerkki")
        self.__alusta = Canvas(self.__paaikkuna, \
                                width = 200, \
                                height = 300, \
                                background = "lightblue")
        self.__alusta.pack()
        self.__alusta.create_line(0, 0, 100, 100, \
                                   fill = "red")
        self.__alusta.create_rectangle(50, 50, 150, 150, \
                                       fill = "blue")
```

Esimerkki piirtävästä ohjelmasta jatkuu

```
self.__alusta.create_oval(100, 100, 200, 200, \  
                           fill = "yellow")  
self.__alusta.create_line(70, 30, 95, 180, \  
                           fill = "green")  
self.__alusta.create_text(100, 250, \  
                           text = "Hurraa!", \  
                           fill = "brown")
```

```
mainloop()
```

```
oma_ikkuna = Piirtoikkuna()
```

Valmiiden kuvatiedostojen lisääminen ikkunaan

- ▶ Luokan `PhotoImage` avulla voidaan `.gif`-, `.pgm`- ja `.ppm`-tyyppisistä (uusissa `tkinter`-versioissa myös `.png`) kuvatiedostoista luoda kuvaolioita, joita voidaan liittää esimerkiksi `Canvas`-tyyppiselle piirtoalustalle ja erilaisiin komponentteihin, kuten painikkeisiin.
- ▶ Muuntotyypisiä kuvatiedostoja (esim. `.jpg`) ei pystytä käsittelemään suoraan `PhotoImage`-luokan avulla. Näitä tiedostoja voi käsitellä esim. `Pillow`-moduulin luokkien avulla.
- ▶ Seuraavilla kalvoilla on kaksi esimerkkiä, joista ensimmäisessä kuvatiedoston sisältämä kuva on liitetty piirtoalustalle ja toisessa kuva on liitetty painikkeeseen.

Kuvaesimerkki 1

```
from tkinter import *

class Kuvaikkuna1:
    def __init__(self):
        self.__paaikkuna = Tk()
        self.__paaikkuna.title("Kuvaesimerkki")
        self.__paaikkuna.geometry("700x500")
        self.__teksti = Label(self.__paaikkuna, \
                               text = " Mukavaa syksyn jatkoa!")
        self.__teksti.pack()
        self.__alusta = Canvas(self.__paaikkuna, \
                                width = 680, height = 450)
```


Kuvaesimerkki 1, jatkoa

```
self.__kuva = PhotoImage(file = "syksy.png")
self.__alusta.create_image(340, 225,\
                            image = self.__kuva)

self.__alusta.pack()
mainloop()
```

```
oma_ikkuna = Kuvaikkuna1()
```

Kuvaesimerkki 2

```
from tkinter import *
import tkinter.messagebox

class Kuvaikkuna2:
    def __init__(self):
        self.__paaikkuna = Tk()
        self.__paaikkuna.title("Kuvaesimerkki2")
        self.__paaikkuna.geometry("400x400")
        self.__kuva = PhotoImage(file = "suorakaide.gif")
        self.__nappi = Button(self.__paaikkuna, \
                               image = self.__kuva, \
                               command = self.anna_ilmoitus)

        self.__nappi.pack()
        mainloop()
```

Kuvaesimerkki 2 jatkuu

```
def anna_ilmoitus(self):  
    tkinter.messagebox.showinfo("Vastaus", \  
                                "Olitko utelias?")
```

```
oma_ikkuna = Kuvaikkuna2()
```