

VEŽBA 11

Pregled

InkScape – program otvorenog izvornog koda namenjen vektorskom načinu crtanja koristi SVG format kao primarni način skladištenja crteža. SVG format zasnovan je na XML jeziku za opis podataka u strukturi stabla. U prethodnoj vežbi, implementirane su (softverske) ekstenzije za InkScape primenom neposredne interpretacije XML dokumenta, izmene sadržaja i ponovnog izdavanja izmenjenih podataka u vidu XML (odnosno SVG) dokumenta primenom Python modula `ElementTree`.

Autori InkScape-a predvideli su i nešto sistematičniji pristup pisanju ekstenzija u čijem centru se nalazi modul `inkex`. U ovoj vežbi biće proučen ovaj pristup. Ovaj tip ekstenzije pretrpeo je značajne izmene u odnosu na ranije verzije InkScape-a, pre pojave verzije 1.0.1.

Klase od posebnog značaja

Klasa `inkex.EffectExtension`

Ova klasa namenjena je kreiranju ekstenzija koje modifikuju trenutno aktívni crtež (obično selektovane elemente) unutar InkScape-a. Svaki ekstenzija namenjena ovoj vrsti posla treba da nasledi i proširi ovu klasu. U centru pažnje je metoda `effect`.

Klasa `inkex.GenerateExtension`

Ova klasa, za razliku od prethodne, nije namenjena izvršenju izmena u crtežu, nego dodavanju novih elemenata na njega. U centru pažnje je metoda `generate`.

Metode koje zahtevaju implementaciju

Svrha nasleđivanja prethodno pomenutih klasa je kompletno nova implementacija određenih metoda. Ukoliko se odgovarajuće metode ne implementiraju, to će rezultovati porukom o grešci. Navešćemo te metode redom, uz neophodne napomene.

Metoda `add_arguments`

Ekstenzije primaju argumente preko odgovarajućeg dijaloga preko kog je moguće zadati čitav niz parametara. Ova metoda treba da sadrži pozive metode `add_argument` parser objekta kojom se jedan po jedan u bazu analizatora (parser-a) argumenta dodaju relevantni parametri. Forma je sledeća:

```
def add_arguments(self, pars):
    pars.add_argument('--naziv_par', type=int, default=10,
                      help='Number of lines')
    pars.add_argument('--str_par', type=str, default='ABC',
                      help='What to print')
```

U primeru su data dva tipa argumenta, `int` i `str` (ceo broj i string) šta je neophodno zadati preko imenovanog argumenta pod nazivom `type`. Osim toga, zadaju se i podrazumevane vrednosti (za

slučaj odsustva validno zadate vrednosti) i string za pomoć. Veoma je važan prvi (pozicioni) argument koji određuje ime argumenta iz dijaloga za podešavanje (zadatog u *.inx* fajlu). Argument treba da bude identičan imenu argumenta uz dodatak dve crte ispred.

Na ovaj način podaci iz dijaloga će automatski biti preuzeti (analizom komandne linije kojom InkScape poziva ekstenziju) i dodeljeni promenljivama članovima objekta ekstenzije pod imenom *self.options.NAZIV*, pri čemu *NAZIV* odgovara prvom argumentu bez vodećih crta. Nedostajući ili neispravno uneti argumenti imajuće vrednosti navedene kao default.

Metoda *generate*

Ova metoda je neizostavna za *GenerateExtension* tip ekstenzije. Ovakva metoda treba (pomoću *return*) da vrati grafički element. Ukoliko vraća više grafičkih elemenata, elemente treba hijerarhijski grupisati – najbolje unutar elementa grupa. Za grafičke elemente i grupe videti odgovarajući odeljak niže.

```
def generate(self):
    grp = inkex.Group.new('NewGroup')
    # generise se novi element
    new_elem = generisi_element_X()
    # dodaje se novi element u grupu
    grp.append(new_elem)
    return grp
```

Alternativno, moguće je vraćati pojedinačne grafičke elemente pomoću *yield*, čime se ova metoda efektivno pretvara u generator.

Važno: ova metoda vraća grafički element koji će biti dodat na crtež. *return* naredba je obavezna na kraju metode (ili *yield* na odgovarajućem mestu), inače se prijavljuje greška koja upozorava da ništa neće biti dodato na crtež.

Metoda *effect*

Ova metoda obavezna je za ekstenzije tipa *EffectExtension*. Iako metoda ima pristup celokupnom svg dokumentu, uobičajeno je dejstvo efekta usmeriti na elemente koji su selektovani pre poziva ekstenzije, tj. efekta.

Pristup selektovanim elementima se može obaviti pristupom podatku *self.svg.selected* koji je po svom tipu rečnik.

Pozivom metode *self.svg.get_current_layer()* dobija se trenutno aktuelni sloj (layer) za crtanje aktuelnog dokumenta. Na njega se mogu direktno nadovezivati (metodom *append*) novi grafički elementi.

Širina i visina dokumenta na raspolaganju su preko elemenata članova *self.svg.width* i *self.svg.height*.

Važno: ova metoda ne vraća ništa (*return* naredba nije obavezna na kraju). Metoda treba da izvrši potrebne izmene na strukturi dokumenta kroz pristup elementima člana *self.svg*.

Obrada stilova novih i već postojećih elemenata

Kompletan stil (boja, linija, unutrašnjost, itd.) svake grafičke komponente je na raspolaganju preko

člana *style* koji je po tipu rečnik. Ključ svakog elementa stila odgovara nazivu u originalnoj svg specifikaciji, npr. *stroke*, *stroke-width*, itd. Podešavanje boje linije objekta (u primeru je se na objekat poziva promenljiva *gr_elem*) se može uraditi na sledeći način:

```
gr_elem.style['stroke'] = 'blue'
```

Postupak je isti, bilo da je element već postojeći ili novo kreiran.

Grafički elementi crteža

Linija

Početak i kraj linije se zadaju kao par (tuple) od dva elementa kao u sledećem primeru:

```
new_line = inkex.Line.new((x1, y1), (x2, y2))
```

Pravougaonik

Pravougaonik se zadaje preko koordinate gornjeg levog ugla, širine i visine u vidu 4 uzastopna parametra:

```
new_rect = inkex.Rectangle.new(x, y, width, height)
```

Elipsa

Koordinate centra kao i *x* i *y* poluprečnik, zadaju se kao parovi (tuple) od dva elementa kao u sledećem primeru:

```
new_ellipse = inkex.Ellipse.new((xc, yc), (xr, yr))
```

Grupa

Ovo je kontejner element. Sam element nema nikakav prikaz, ali spaja elemente koji su mu podređeni.

```
new_grp = inkex.Group.new('naziv_nove_grupe')
```

Provera tipa elementa

Postojeći elementi crteža se pojavljuju kao objekti određenih klasa u odgovarajućim rečnicima. Tip elementa može se proveriti na osnovu pripadnosti klasi. Npr. provera da li je element pravougaonik može se uraditi kao

```
if isinstance(element, inkex.Rectangle):
    # akcija ako je element tipa pravougaonik
```

Parametri elementa

U zavisnosti od tipa elementa, na raspolaganju su različiti atributi koji sadrže parametre geometrije (koordinate, dimenzije, itd.). U primeru su dati atributi sa podacima o pravougaoniku.

```
# el_rct - objekat koji sadrzi pravougaonik
el_rct.left      # x-koordinata leve ivice
el_rct.top       # y-koordinata gornje ivice
el_rct.width     # sirina pravogaonika
el_rct.height    # visina pravougaonika
```

Drugi elementi crteža imaće drugačije atribute. Pre pristupa treba proveriti tip elementa.

Poruke o grešci za krajnjeg korisnika

U posebnom dijalogu koji se prikazuje korisniku može se pojaviti proizvoljna poruka o grešci. Za prikazivanje ove poruke treba pozvati funkciju:

```
inkex.errormsg('string sa porukom o gresci')
```

String koji je argument funkcije će bez izmene biti prikazan korisniku.

Struktura ekstenzije

Ekstenzija sadrži definiciju nove klase koja nasleđuje odgovarajuću klasu definisanu u modulu `inkex`. Na samom kraju se instancira nova klasa i poziva se metoda `run`. Time se pokreće procesiranje i efektivno obrada postojećih i generisanje novih elemenata `svg` crteža.

```
import inkex
class NewEx(inkex.EffectExtension):
    def add_arguments(self, pars):
        # dodavanje opcija

    def Effect(self):
        # obrada elemenata

nc = NewEx() # instanciranje objekta
nc.run() # ovim se efekat pokrece
```

Zadaci

Za rešavanje svih zadataka koristiti modul `inkex`. U skladu sa zadatkom izabrati tip ekstenzije – `GenerateExtension` ili `EffectExtension`.

1. Napisati ekstenziju koja crta pravougaonik oko celog crteža (koristiti `width` i `height` attribute `svg` dokumenta). Boja treba da bude crvena, a debljina linije (`stroke-width`) 2 mm¹. Dijalog za zadavanje parametara nije potreban.
2. Prethodni zadatak uraditi tako što će se pre poziva ekstenzije unutar dijaloga zadavati string parametar koji će se konvertovati u `float` vrednost koja će potom biti korišćena za određivanje debljine linija. Koristiti `try/except` konstrukciju za detekciju `ValueError` izuzetka koji se dobija ako string nije moguće konvertovati u `float`. Debljina mora biti veća od 0. U slučaju bilo kakve greške debljina linije treba da bude 1.0
3. Napisati ekstenziju koja će promeniti boju linije (`stroke`) svim izabranim elementima. Za izbor boje, dijalog preurediti tako da daje izbor crne, crvene, zelene i plave boje pomoću radio dugmića.
4. Napisati ekstenziju za `InkScape` koja iscrtava elipsu crvenom bojom debljine linije 1.0 mm unutar izabranog pravougaonika. Obavestiti korisnika porukom o grešci ako nema izabranih elemenata ili je selektovan element koji nije pravougaonik.

¹ element sitla `fill` postaviti na `none`. U suprotnom će se `fill` smatrati nedefinisanim i biće nasleđen od nekog roditelja i element će najverovatnije izgledati popunjen.

Dodaci

Razni tipovi podataka u dijalogu za podešavanje pre poziva ekstenzije

Izgled dijaloga za zadavanje parametara zadaje se u *inx* fajlu ekstenzije. Tipovi relevantni za ovu vežbu navedeni su ispod. Imena atributa bi trebalo da govore sami za sebe. U slučaju *string* i *int* parametara tekst sadržaj elementa predstavlja početnu vrednost u trenutku pojave dijaloga, dok kod radio dugmića tekst deo predstavlja natpis pored odgovarajućeg dugmića. Takođe, kod radio dugmića atribut *value* određuje vrednosti od kojih će jedna biti vraćena kao vrednost parametra u zavisnosti od izabranog dugmića.

String

```
<param name="width" type="string" gui-text="Sirina linije">2.0</param>
```

Integer

```
<param name="segs" type="int" min="10" max="60" gui-text="Br. segmenata">20</param>
```

Višestruki, međusobno isključiv izbor – radio dugmići

```
<param name="color" type="optiongroup" gui-text="Izaberite boju" appearance="radio">
  <option value="black">black</option>
  <option value="red">red</option>
  <option value="green">green</option>
  <option value="blue">blue</option>
</param>
```

Ako se umesto *radio* za atribut *appearance* stavi *combo*, izbor će se obavljati preko *combo-box*-a umest dugmića.