

13 DOGODKI GRADNIKOV IN ODZIVI NANJE

Gradniki grafičnega vmesnika dobijo svojo polno funkcionalnost šele takrat, ko se začnejo odzivati na uporabnikove akcije. Zaradi akcij uporabnika in drugih virov se v programskem okolju zgodijo različni "dogodki": uporabnik klikne na gumb, povleče z miško, izbere nekaj iz seznama in podobno. Želimo, da se program na dogodke ustrezno odziva. Odziv na posamezen dogodek opišemo s programsko kodo – z odzivno *metodo*. Kako je to organizirano v programskem jeziku Java?

V Javi je marsikaj že pripravljeno:

- **razredi za dogodke:** *ActionEvent*, *MouseEvent* ...
Da, tudi dogodki so v Javi obravnavani kot objekti – s svojimi spremenljivkami in metodami. Ko se zgodi določen dogodek, se ustvari objekt takšnega razreda, v programu pa lahko uporabimo njegove spremenljivke in metode. Dogodki miške (objekti razreda *MouseEvent*) imajo na primer metodo *getX()*, ki vrne vodoravni položaj miškinega kazalčka na zaslonu;
- **vmesniki za dogodke:** *ActionListener*, *MouseListener*, *WindowListener*
Vmesnik za dogodke predpisuje metode, ki naj jih ima razred, da bo lahko obravnaval posamezen dogodek. Tako na primer vmesnik (*interface*) *MouseListener* določa pet metod: *mouseClicked()*, *mouseEntered()* ...
- **model obravnave dogodkov.**
Ta model je precej kompleksen, njegove osnove bomo opisali v naslednjem razdelku.

13.1 Obravnava dogodkov v Javi

Model obravnave dogodkov v Javi bomo opisali v nekaj točkah.

1. Gradniku grafičnega programa se lahko 'zgodijo' določeni dogodki.

Dogodki so na primer:

- klik na gumb,
- vstop (angl. *enter*) z miškinim kazalcem nad gradnik,
- klik na izbiro iz seznama ali iz menija,
- pritiski na tipke pri vnosu besedila v vnosno polje,
- in drugi.

2. Gradnik oz. komponenta, na kateri se zgodi dogodek, je izvor (vir) dogodka ali dogodkov.

Izvori dogodkov so lahko gumb, napis, seznam, meni, vnosno polje in drugi gradniki, pa tudi zunanje naprave, časovniki in drugi.

3. Za spremljanje dogodkov moramo vključiti spremljanje oz. 'poslušanje' dogodkov.

Na dogodke, ki nas zanimajo, moramo prežati in – ko se zgodijo – ustrezno reagirati. Dogodki, ki jih ne prestrežemo, odidejo brez učinka na program.

4. Poslušalec je objekt, ki spremlja določene dogodke in ima programsko kodo za odziv nanje.

Tako razlikujemo npr. poslušalce za klik na gumb, poslušalce za dogodke okna (zapiranje, pomanjšanje ...), poslušalce za dogodke miške, poslušalce za dogodke tipkovnice in druge. Poslušalec bo izpolnil predpisane zahteve, če bo izdelan iz ustreznega razreda.