

# GUI - događaji (Events) i izuzeci

Bojan Tomić

# Događaji

---

- GUI reaguje na događaje (events)
- Događaj je neka akcija koju korisnik programa ili neko drugi izvrši korišćenjem perifernih uređaja (uglavnom miša i tastature)
- Svaka GUI komponenta reaguje na određene događaje
- Postoje dve vrste događaja:
  - Primitivni događaji (low-level events)
  - Semantički događaji (semantic events)

# Događaji

---

- Primitivni događaji su direktno vezani za neku komunikaciju preko perifernog uređaja ili jednostavnu promenu nad komponentom, npr.:
  - `mouseClicked` - kada korisnik klikne mišem na komponentu
  - `mousePressed` - kada korisnik klikne mišem ali ne pusti dugme
  - `mouseReleased` - kad korisnik pusti dugme posle klika
  - `keyPressed` - kada korisnik pritisne neki taster na tastaturi a komponenta je u fokusu
  - `key Released` - kada korisnik pusti dugme

# Događaji

---

- Semantički događaji predstavljaju neke uopštene događaje bez obzira na to da li su nastali kao posledica aktivacije nekim perifernim uređajem ili na drugi način
- U principu, često se koristi samo jedan semantički događaj - « actionPerformed »
- On predstavlja akciju neke komponente bez obzira na to da li je izazvana mišem, tastaturom...
- Korišćenjem semantičkih događaja kod je robustniji

# Događaji

---

- Svaka komponenta može da reaguje na događaje
- Da bi to mogla, komponenti se dodaju tzv. oslušivači (« action listener ») - objekti odgovarajuće klase
- U oslušivač se dodaju metode koje se aktiviraju kada se neki događaj desi
- Ove metode se nazivaju « event handlers »

# Događaji

---

- Svaka komponenta može da ima nula, jednu ili više oslušivača i metoda koje reaguju na događaje
- Kada se desi događaj, izvršiće se kod u okviru tela odgovarajuće « event handler » metode

# Događaji

- Kod jednog « event handler » a za dugme koji reaguje na « actionPerformed » događaj izgleda ovako:

```
jButton.addActionListener(new
    java.awt.event.ActionListener() {
public void
    actionPerformed(java.awt.event.ActionEvent e) {
//Neki kod koji se izvrsava svaki put kada se
//klikne na ovo dugme
}
});
```

# Unos podataka preko GUI-ja

---

- Svi podaci koji se unesu su u tekstualnoj formi
- JTextField
  - getText metoda
- JTextArea
  - getText metoda
- Potrebno je pretvoriti podatke u odgovarajući tip
  - Double.parseDouble(String s)
  - Integer.parseInt(String s)



# Izuzeci i GUI

---

- Kada se baci izuzetak u Java programu koji nema GUI
  - Program se prekida
  - Poruka izuzetka se prikazuje na standardnom izlazu
- To **NIJE** slučaj kod GUI aplikacija
  - Program se ne prekida - samo se prekine izvršavanje metode koja je bacila izuzetak i onih metoda koje su je pozvale, ali program može da nastavi da radi
  - Ne prikazuje se poruka izuzetka jer se standardni izlaz ne vidi preko GUI forme

# Izuzeci i GUI

---

- Niti – procesi koji konkurentno rade na jednom procesoru
- Izuzetak prekida jednu nit
- U GUI programu rade makar dve, tri niti
  - Nit koja predstavlja main metodu (Initial thread)
  - EventDispatch nit koja omogućava reagovanje GUI-ja na događaje
  - Radne niti (Worker thread)
- Kad se desi izuzetak, prekida se samo EventDispatch nit, ali se odmah napravi nova i program nastavlja da radi

# Izuzeci i GUI

---

- Initial thread
  - Pokretanje programa iz main metode i to je to (EventQueue)
- EventDispatch
  - Registruje sve događaje i stavlja ih na EventQueue
  - Izvršava sve event handler metode (ako se izvršavaju brzo)
- Worker thread
  - Programer ih definiše i aktiviraju se da urade deo posla kad je taj posao spor ili drugo traje

# Izuzeci i GUI

---

- Ako se izuzetak ne obradi, neće se ni primetiti da se išta desilo
- Rešenje
  - Uхватiti svaki izuzetak (try-catch blok)
  - Prikazati poruku izuzetka u okviru neke komponente GUI forme
- Jedna varijanta je da se napravi posebna forma koja će da služi samo za to
- Druga varijanta je da se iskoristi JOptionPane klasa

# Izuzeci i GUI

- Klasa “javax.swing.JOptionPane”
  - Polu-gotova forma koja služi za brzi prikaz poruka, opomena isl.
  - statička metoda “showMessageDialog()”

```
jButton.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent e) {  
  
        try{  
            double dinari = Double.parseDouble(jTextField.getText());  
            double evri = dinari / 85;  
            jTextField1.setText(evri+" evra");  
        } catch (Exception exc){  
            JOptionPane.showMessageDialog(jButton,  
                "Morate uneti broj", "Greska",  
                JOptionPane.ERROR_MESSAGE);  
        }  
    }  
});
```