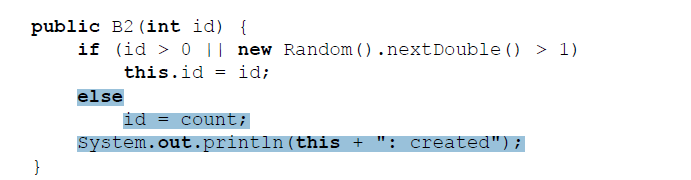
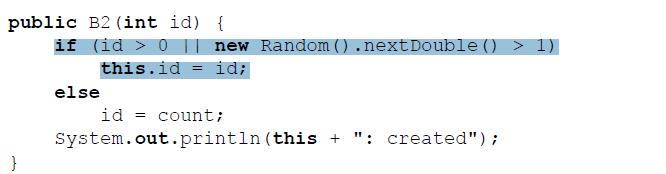
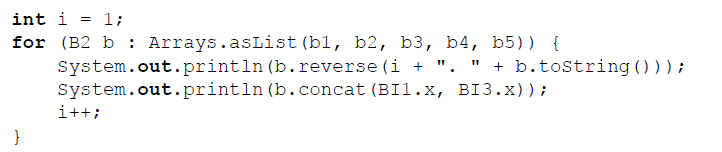
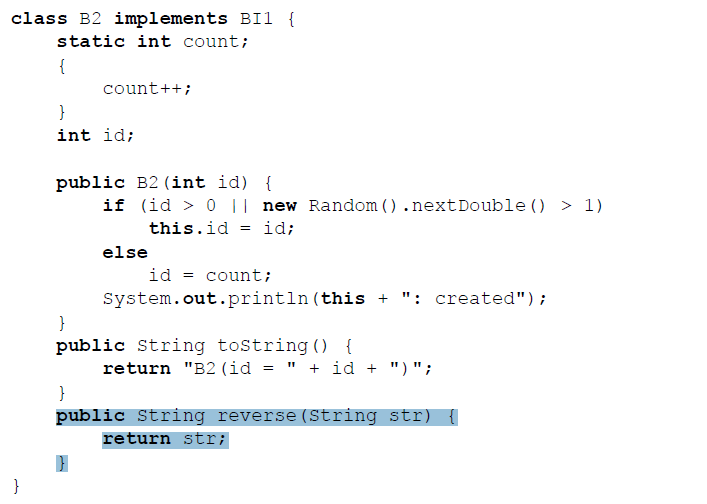
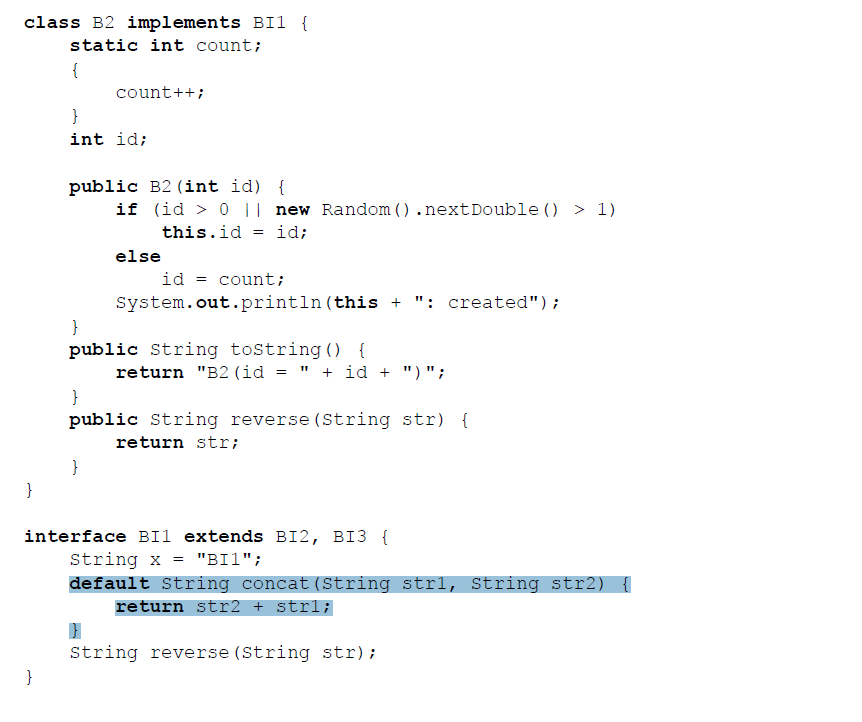
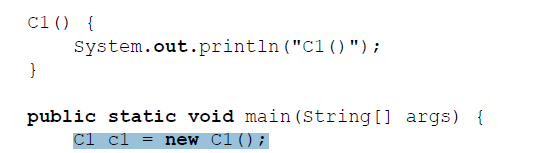


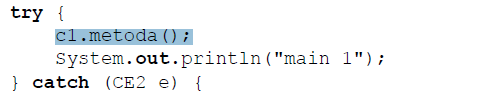
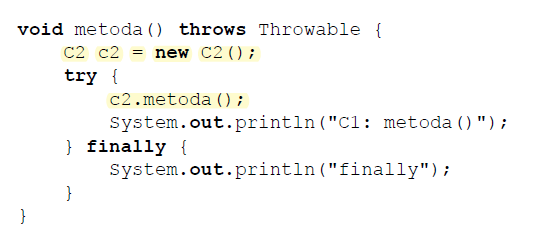
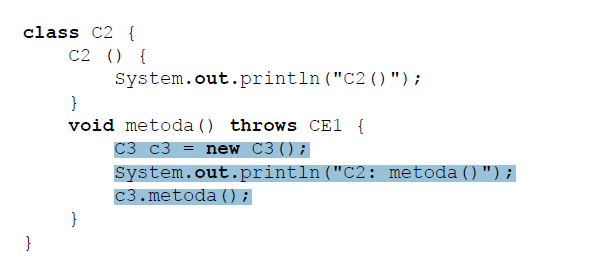
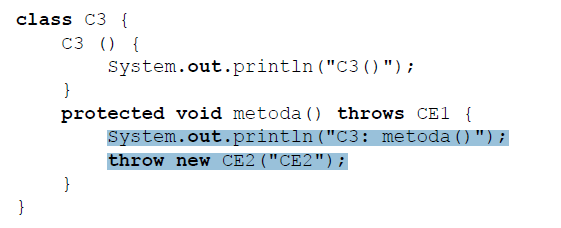
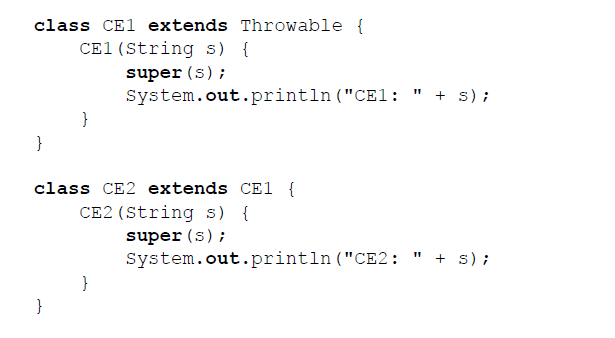
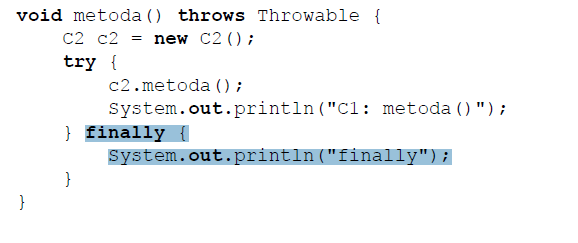
Polazimo od glavne funkcije:  
B2 b1 = new B2(-2);   
B2 b2 = new B2(-1);   
B2 b3 = **new** B2(0);  
  
  
id = count; se odnosi na **id – argument Constructor-a**, promjenjiva koja je argument funkcije znači, dok se **this.id** ne mijenja a kako se isti nigdje nije eksplicitno inicijalizovao, Java kompajler podrazumijeva da je njegova vrijednost = 0!!

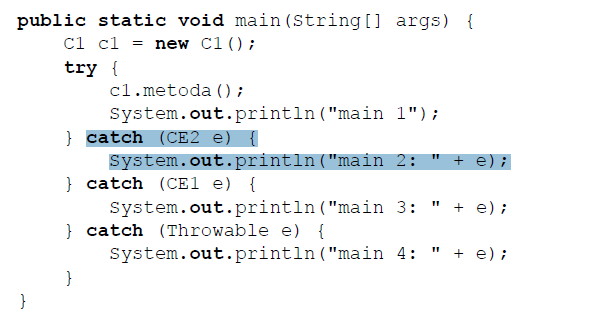
B2 b4 = **new** B2(1);   
B2 b5 = **new** B2(2);  


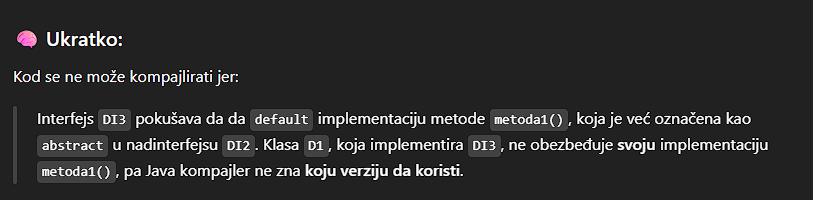
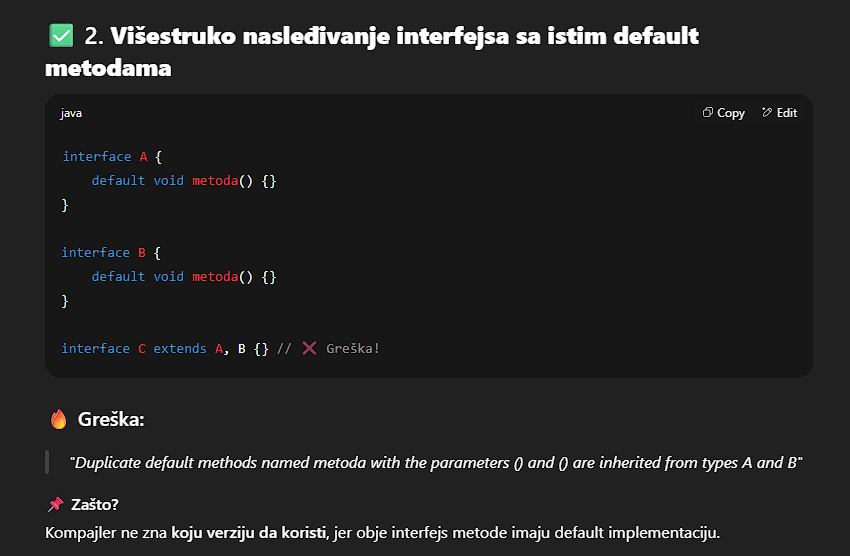
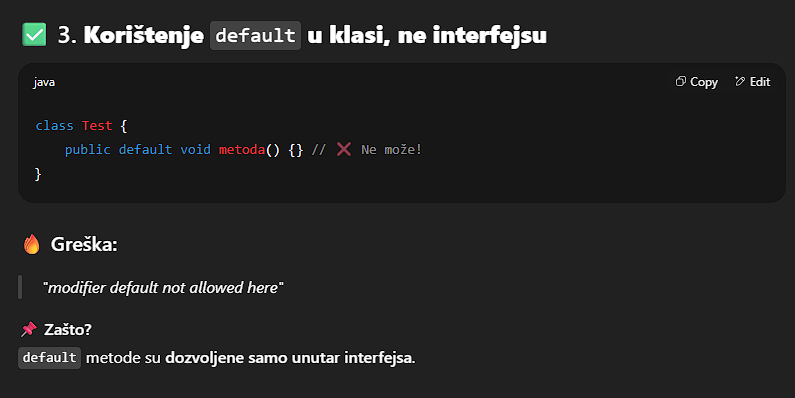
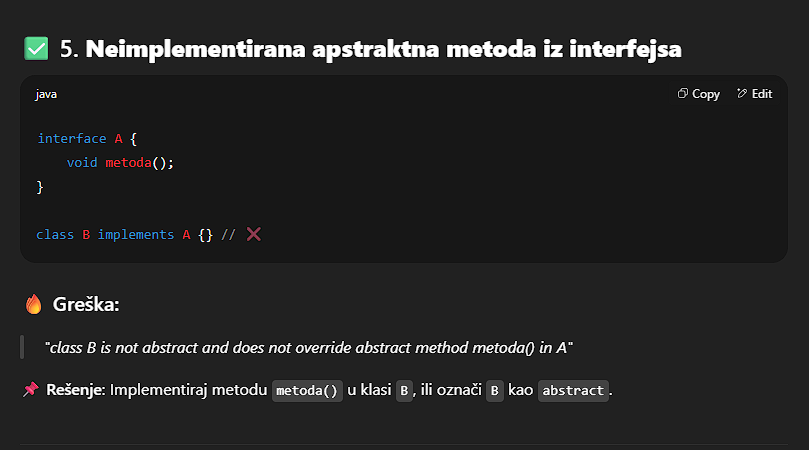
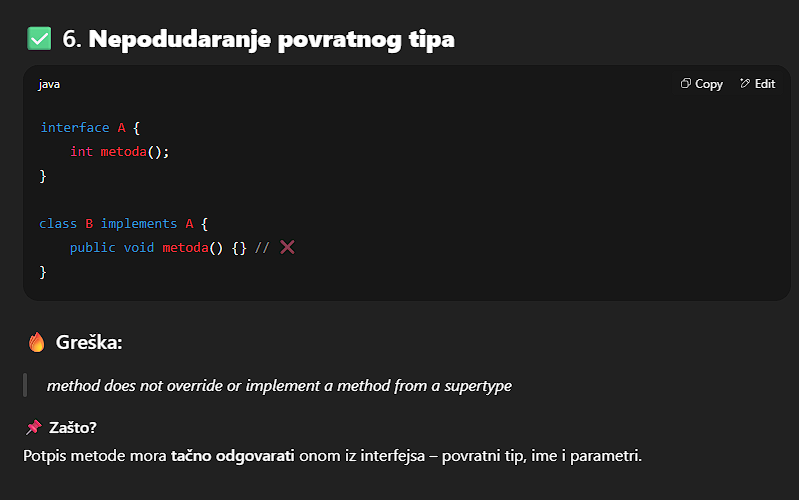
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
b.reverse(..  
  
jer b - direktno referencira objekat klase B2, u toj klasi imamo jasno redefinisanu metodu reverse()!!  
  
*i + ". " + b.toString())*što ispiše **i.B2(id=X**), znači vrijednost od i. pa poziv metode **toString()** B2 klase!!  
  
Klasa B2, implementira interfejs BI1 u kojem imamo jasno redefinisanu metodu concat()  


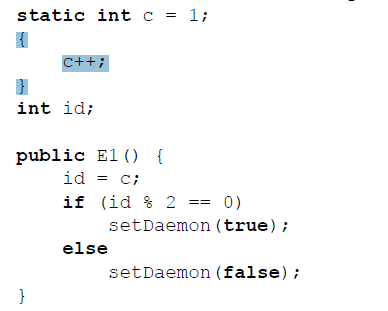
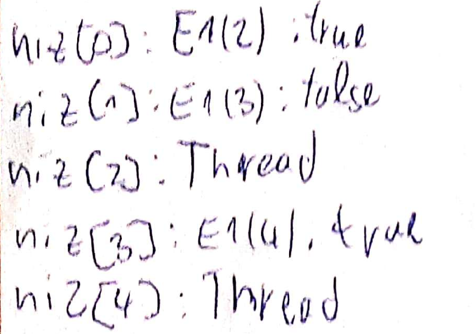
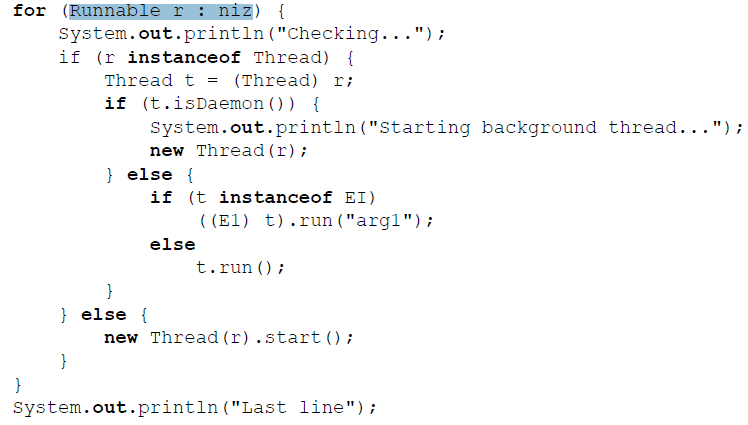
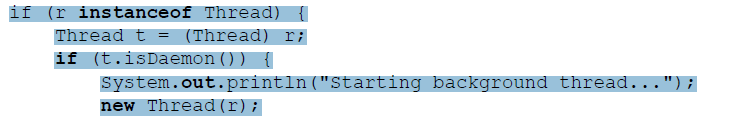
c)

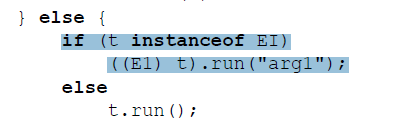
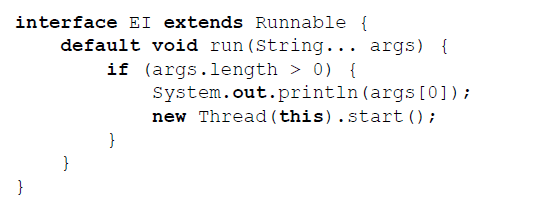
  
Pozvaće Constructor i ispisaće C1() ..

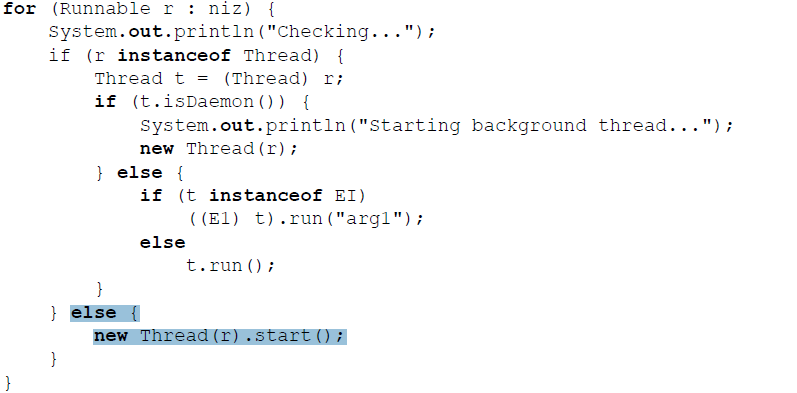
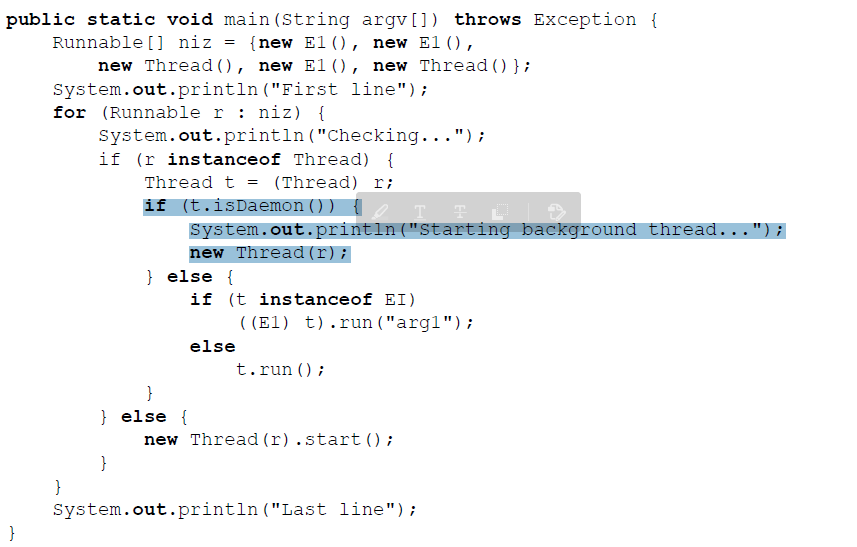
  
vodi me u  
  
Prethodno što će ispisati C2() zbog ConsturctorCall, a dalje me vodi u c2.metoda():  
*(Elem, obrati pažnju na throws Throwable –što znači da je metoda spremna na bacanje bilo kakve vrste izuzetaka!!)*  
  
I tako se te tri linije code-a izvršavaju redom..došli smo do  
  
Dalje idemo u (Jer je bačen CE2!!):  
  
To nam ispisuje:  
CE1: CE2  
CE2: CE2  
  
E SAD, BAČEN JE CE2 KOJI SE PROPAGIRA!!  
Izuzetak propada, sa Stack-a se skidaju prethodno pozvane metode i dolazimo do  
  
pri čemu se izvršava finally ali CE2 nastavlja da pada!!

  
UHVAĆEN JE!!  
main 2: CE2 je naš poslednji ispis!!!

d)  
  
  
  
  
  
  
  


e)  
  
To nas vodi u:  
  
Svaki put kada uđeš u klasu, IZVRŠIĆE SE TAJ NON-STATIC-BLOCK!!  
Ovakav imaš rezultat:  
  
Dalje:  
  
  
niz(0):  


Idemo onda na niz(1):  
  
što nas dalje vodi u:  


To se redom izvršava pa idemo u niz(2):  
  
  
  
Nakon izvršavanja, idemo u niz(3):  


Nakon toga za niz(4) kao i za niz(2) da bi na kraju imali još:  
