1) Za 1. zadatak, koristiti debugger a ne AI!!

2) U napisanim rješenjima, postoji mogućnost da se ipak negdje nalazi neka pogrešna misao, rješenje i tome slično ..

3)

1.Zadatak,

a) mogući scenario je sljedeći:  
-poziv privatnog Constructor-a, klasična kompajlerska greška..  
-poziv nepostojećeg Constructor-a, poziv Constructor-a sa pogrešnim brojem/redoslijedom argumenata, ...opet klasična kompajlerska greška ..  
-tokom izvršavanja programa može se desiti StackOverflow Exception ..

U slučaju kompajlerskih grešaka, program se uopšte neće ni kompajlirati.  
Potrebno je naći grešku, podvući liniju code-a u kojoj nastaje ista i time je zadatak 100% kompletiran!

U slučaju StackOverflow Exception-a, potrebno je pisati output code-a sve do momenta dok ne dođe do Exception-a!!

b)

-ako klasa koja nasleđuje interfejs, ne implementira( ne redefiniše!!) neku od metoda tog interfejsa --PROGRAM SE UOPŠTE NEĆE KOMPAJLIRATI!!

Može se desiti da u interfejsu recimo imamo metode imena "clone/finalize" kao što u Object klasi imamo iste okarakterisane specifikatorom pristupa "protected"!

Sve metode interfejsa, po default-u su "public abstract" što znači da potpisi metoda iz "clone/finalize" iz Object klase i interfejsa, nisu identični!!

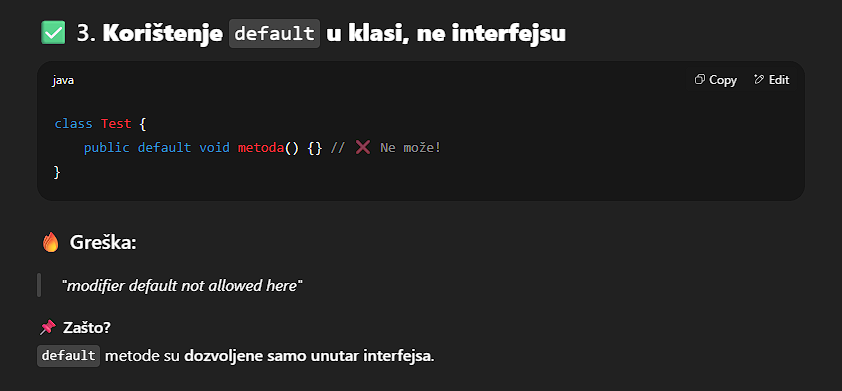
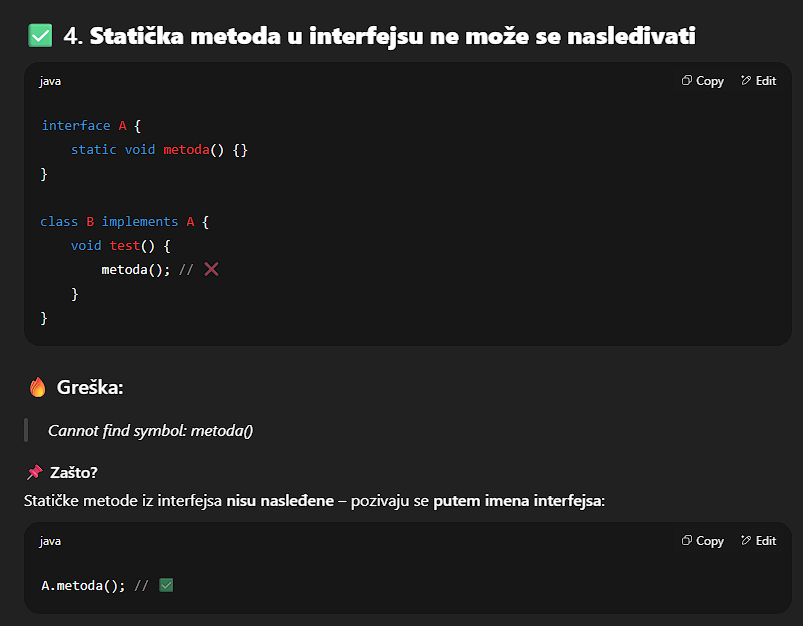
Dakle: \*Obratiti pažnju na potpis metode!!

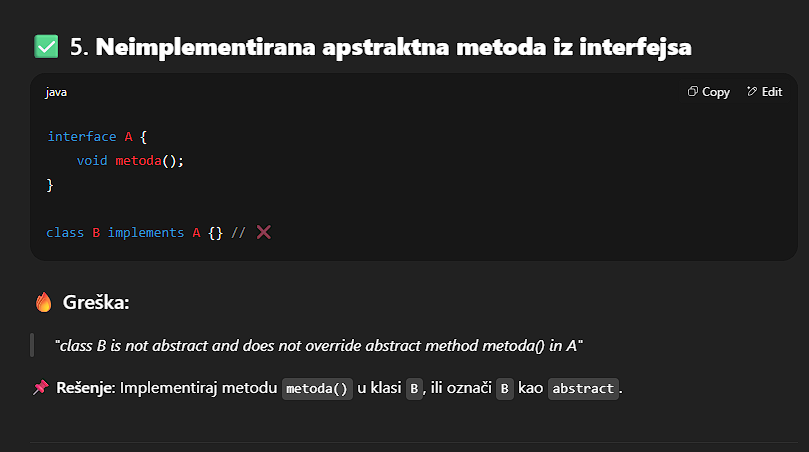
\*Obratiti pažnju na listu argumenata, na specifikator pristupa!!

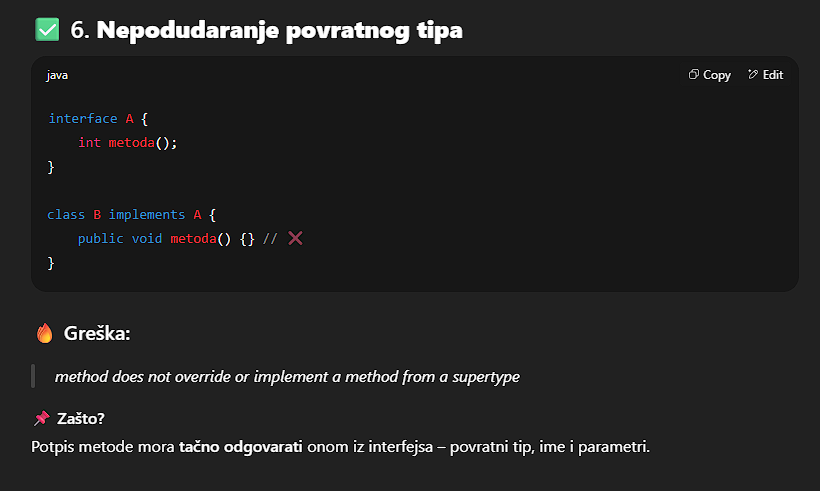
\*Obratiti pažnju na povratni tip metode!!

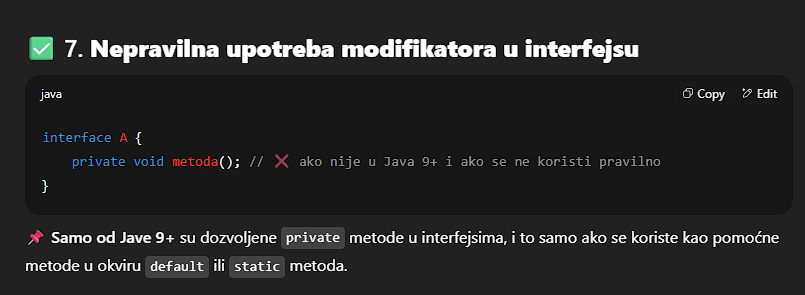
\*Obratiti pažnju na metode okarakterisane specifikatorom pristupa "static"!!

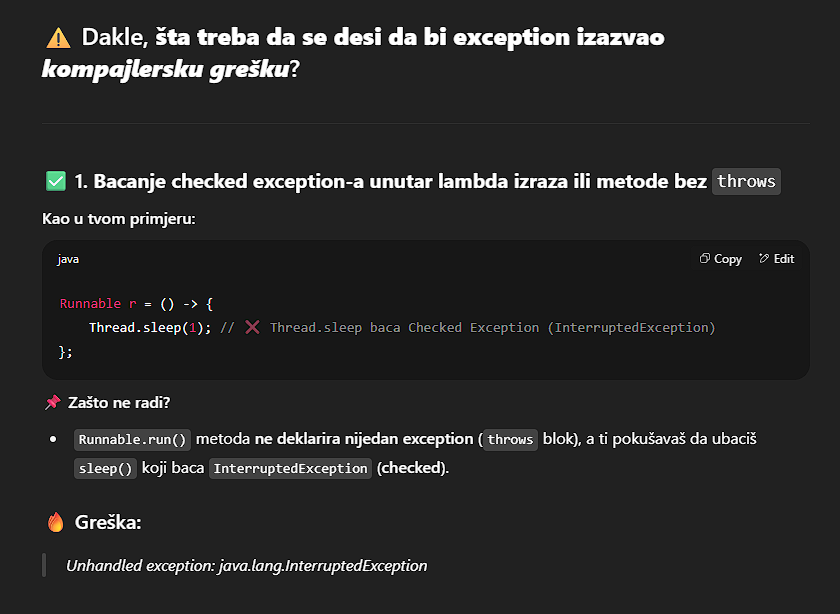
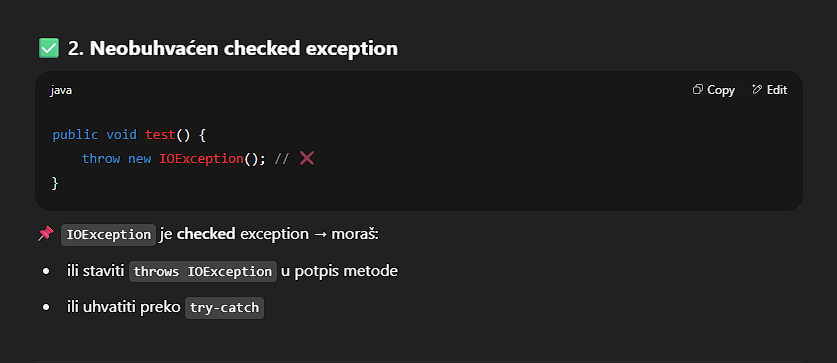
ZNAČI, hint je da se prvo pogleda koja to klasa implementira koji interfejs, pa ako ta klasa ne redefiniše neku od njegovih metoda, to je kompajlerska greška, program pada!

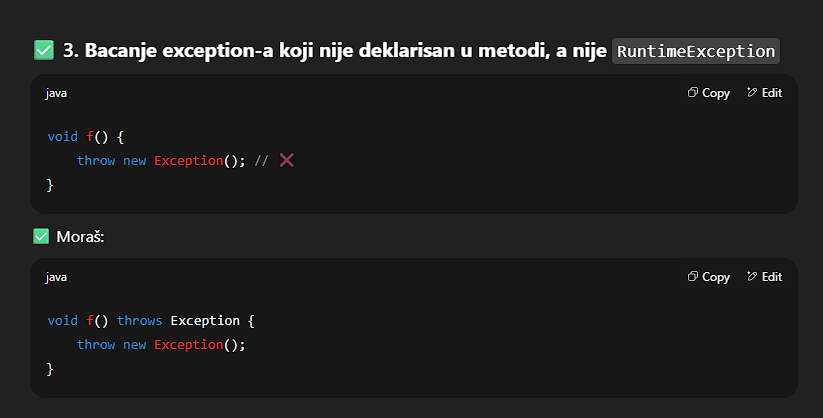
Najskuplji potez je praćenje izvršavanja rada programa line-by-line, ispisivanje output-a i tome slično!  
  
  
  
  


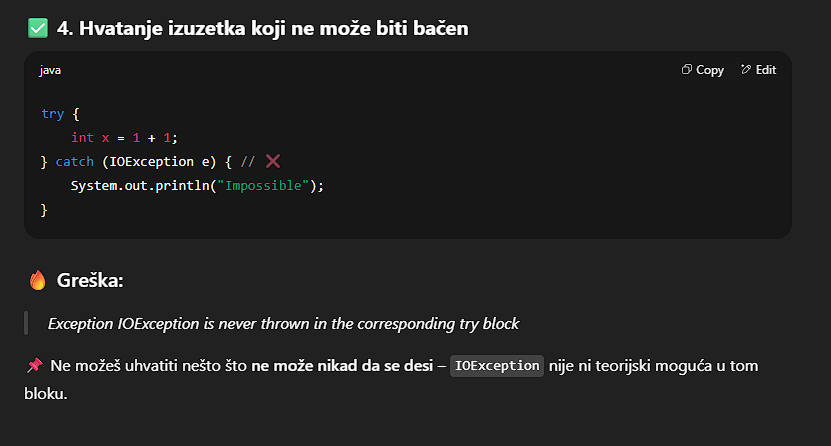


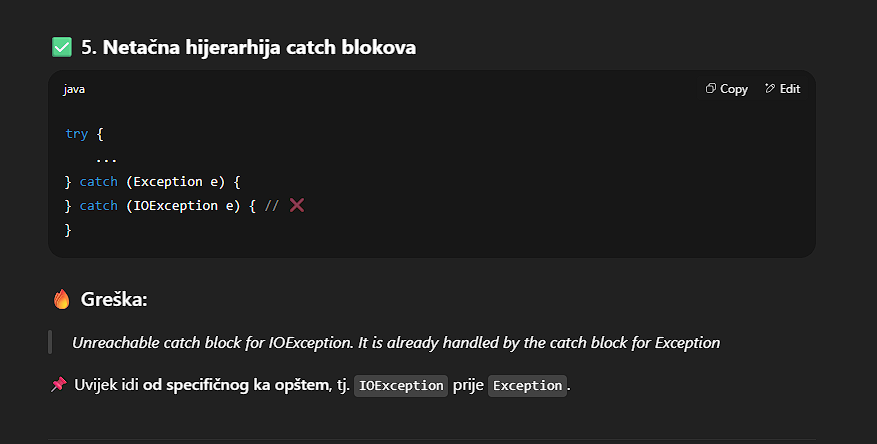


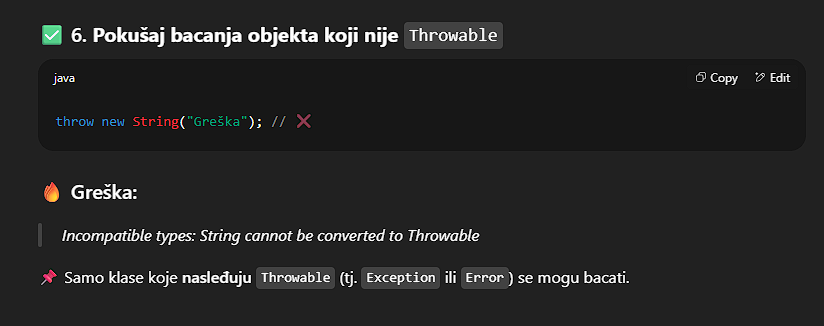


c)  
  














d) Često, najčešće bude nekakav kratak i konkretan output ..  
e) Threads..  
  
f)  
