

ND7

Uzd 2 (vidinės klasės)

ND

- Sukurit dvi klasės:
 - IsorineKlase2 - public klasę
 - Turi kintamuosius, pvz.:
 - private String pavadinimas;
 - Protected String nr;
 - Int sk;
 - VidineKlase2 - IsorineKlase2 klasės bloke sukurkite dar vieną public klasę
 - Turi metoda spausdinti()

Main metode pabandyti sukurti

abejų klasių egzempliorių

IsorineKlase2 klasės kintamiesiems priskirti reikšmes

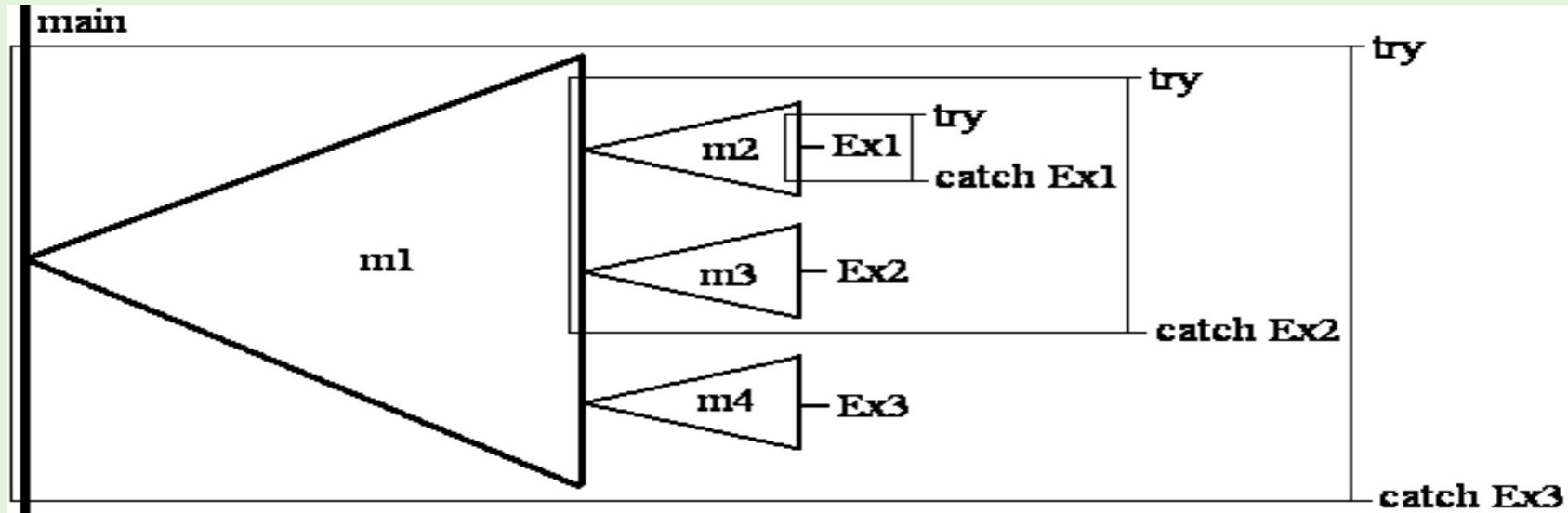
ir jas atspausdinti VidineKlase2 klasės *spausdinti()* metodo pagalba

Išimčių apdorojimas ND

MIF

-
1. Susikurkite tris išimtis E1, E2 ir E3, paveldėtas iš Exception. Susikurkite klasę A, turinčią metodus m1, m2, m3, m4 ir main. Iš metodo main kvieskite metodą m1, o iš metodo m1 – metodus m2, m3 ir m4. Metode m2 generuokite išimtį E1, metode m3 – E2, metode m4 – E3. Metode main įdėkite išimčių E1, E2 ir E3 gaudymą.
 2. Pakeiskite metodą main taip, kad būtų gaudomos bazinės klasės išimtys.
 3. Pakeiskite metodą main taip, kad išimtys būtų ne gaudomos, o išvedamos į konsolę.
 4. Klasėje A išimčių gaudytojus išdėstykite taip, kaip pavaizduota kitoje skaidrėje. Patikrinkite, ar ir dabar reikalingi visi sakiniai throws.
 5. Sukurkite klasę B, turinčią tuos pačius metodus, kaip ir klasė A. Išimčių generavimą pakeiskite realiais išimtis generuojančiais įvykiais: E1 keisti dalyba iš nulio, E2 – kreipimusi į masyvo elementą už masyvo ribų, E3 – kreipimusi į neegzistuojantį objektą (pvz., String tipo kintamajam priskirti null ir kviesti kokį nors String metodą). Atitinkamai modifikuokite išimčių gaudytojus. Patikrinkite, ar reikalingi sakiniai throws. Kodėl?

Išimčių apdorojimas (ND pav.)



Self learning

- Finansai: pagalvoti apie klaidų šeimą, kur kitos klaidos galėtų būti metamos;
- **@Annotation** - kas tai, kam tai skirta?
- Java 8 Stream API - kam skirta, kaip naudoti. [1link](#), [2link](#), [3link](#), [4link](#), [5link](#);
- Java 8 Lambda (“lėmdos”) Expressions: [1](#), [2](#), [3](#), [4](#), [5](#).