



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

IDK2015 Objektorienteeritud programmeerimine Javas

Martin Rebane (martin.rebane@ttu.ee)

Järgmisel nädalal

- Loengu lõpus test
- Kordamisküsimused Moodles



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Testis

- Põhimõisted – valikvastustega (klass, objekt, väli ehk muutuja, meetod, käitumine, olek, nähtavus, pakett, laiendamine jmt)
- Süntaks – vali korrektne süntaks või paranda viga
- Tsüklite tööpõhimõtted – for, foreach, while



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Loengumaterjalid

- Moodle – laadige alla uusim versioon



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Eelmisel nädalal

- Unboxing ja autoboxing – konverteerimine primitiivse tüübi ja sellele vastava wrapper klassi vahel
 - `Integer i = 5; // autoboxing, ilma new()-ta`
 - `int b = i % 2; // unboxing – Integer-tüüpi objektist tehakse primitiivset tüüpi muutuja`



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Eelmisel nädalal

- `static` keyword – klassi omadus, mitte objekti!
- `final` keyword – sellise välja väärtust ei saa muuta
- `Collections()` - arenenum vorm massiividest
 - Nendes saab hoida ainult objekte (mitte primitiivset tüüpi muutujaid)
 - Näide: `ArrayList<Type>`



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Eelmisel nädalal

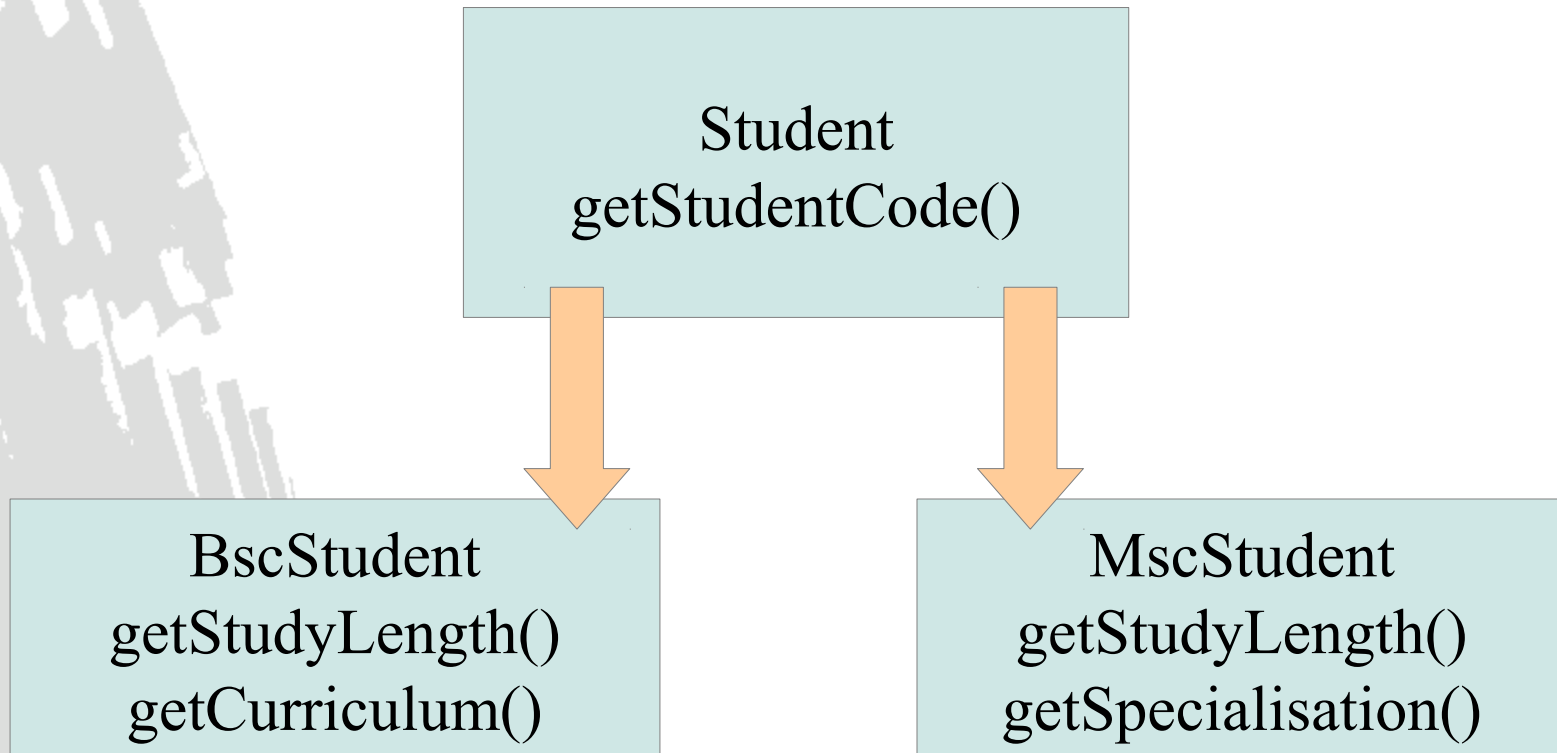
- Iga klass laiendab vaikumisi Object klassi
- Vaatasime pärimist, kus üldine objektitüüp määrab ühise käitumise ning kitsam objektitüüp sellele kitsale osale omase käitumise. Näide:
 - Üldine tüüp Student
 - Kitsam tüüp BscStudent, MscStudent



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Eelmisel nädalal



- Alamtüübis erinevalt käituvad või ainult sellele tüübile omased meetodid

API – application programming interface

- Tarkvarakomponent, mis pakub teenust
- Arendaja annab kasutajale lubaduse API võimaluste kohta, sh nii meetodid kui ka **erindid**
- Kasutaja ei pea teadma, kuidas API realiseeritud on



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Veatöötlus, erindid

- Oskame ette näha olukordi, kus programm ei pruugi käituda soovitud
- Võib-olla suudame programmi töö taastada
- Informeerime kasutajat veast viisakal moel



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

try – catch – finally

```
try {  
    new Item(8);  
} catch (Exception e) {  
    System.out.print(e.getMessage());  
}
```

- Lisaks võimalik *finally* osa – täidetakse alati kui programm ellu jääb



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erind on object

- `new Exception(„Viga”);`
 - tavaline uue objekti loomine
 - konstruktorile anname String-tüüpi argumendi „Viga”
- `catch (SomeException e)`
 - püütakse kinni SomeException-tüüpi objekt, mida saab kasutaja nime „e” abil



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erind on objekt

- `Student s = new Student(„John”);`
- `Exception e = new Exception(„Veateade”);`
- Ärge unustage, et erind on objekt!



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erindite tüübid

- Checked exceptions – kontrollitud erindid
 - Informeerivad kasutajat olukordadest, kus normaalse töö taastamine on võimalik, näiteks: meetodile on antud vale argument
 - Kasutaja peab nendega tegelema!
- Unchecked exceptions – kontrollimata
 - Taastumine pole tavaliselt võimalik
- Errors - vead
 - Programmikoodist sõltumatud vead



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erindite tüübid

- Reeglina on arendaja jaoks olulised *checked exceptions* e kontrollitud erindid
- *Unchecked exception* viitab mingile (loogika)veale programmis
- *Error* on programmiväline viga – seda me ise kunagi ei loo



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Kontrollimata erind

- Kood kompileerub, aga kuskil on (loogika)viga
- Näiteks meetod ootab objekti, aga saab hoopis *null*; või jagamine nulliga
- Võib jätta töötlemata, eriti kui tulevad kellegi teise loodud API kaudu



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Error ehk viga

- Programmiväline viga, nt faili lugemine katkeb kõvaketta vea tõttu
- Kasutajale kuvatakse töötlemata veateade



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Kontrollitud erind (checked ex.)

- Kontrollime alati kas erind tekitati
- Kui jah, siis töötleme seda
- Võimalusel taastame programmi normaalse töö ...
- ... ja/või vihjame kasutajale, kuidas seda teha



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Näide

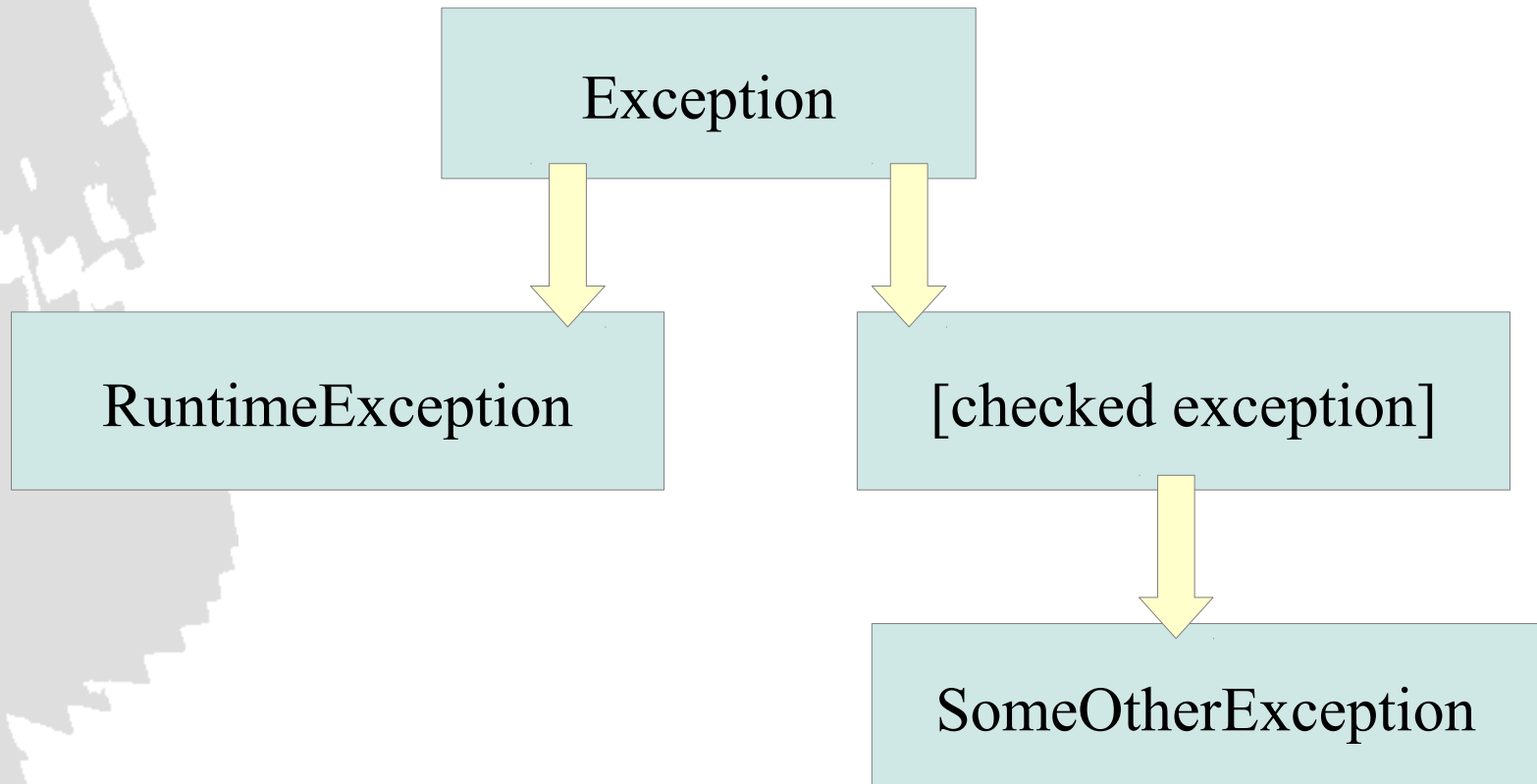
- Student – age limit



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erindid on laiendatavad



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erindi loomine

- Esmalt mõtle, kas erindit on vaja!
- Võib-olla saab asendada if-else blokiga
- `obj.doSomeStuff()` ;

versus

```
if (obj.someStuffPossible()) {  
    obj.doSomeStuff();  
}
```



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erindi loomine

- **throw** new StudentException();
- meetod peab informeerima kontrollitud erindi võimalikkusest:
throws StudentException
- Kui veaolukord tekib – loo kõige sobivamat tüüpi erind
- Üldist tüüpi Exception objekti ei tohiks kunagi ise luua



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Olulised meetodid

- `printStackTrace()` ;
 - kuvab informatsiooni vea esinemise kohta – nn veapinu
- `getMessage()` ;
 - kuvab veateate



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Enda erindite loomine

- Esmalt püüa kasutada Java erindeid
- Loomise eesmärgiks võib olla pakkuda spetsiifilisemaid meetodeid vea põhjuste kohta



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Erindite aheldamine e chaining

- Liiga üldine viga täpsemaks
- Liiga täpne viga üldisemaks
- Mitme vea liitmine



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Kus püüda erindit?

- Kui püüda probleemi tekkekohale lähemal, siis on rohkem võimalusi probleemi lahendamiseks
- Kui püüda võimalikult kõrgel tasemel, siis ei pea allpool liiga palju koodi erinditele kulutama (tsentraliseeritud veahaldus)



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Säilita veaeelne olukord (nn failure atomicity)

- Kui erind tekib, püüa see luua nii, et objekti olek ei saaks muudetud
 - nii on võimalik objekti peale vea põhjuse kõrvaldamist edasi kasutada



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Try with resources

- Kui soovite avada ressursi, siis ei pea muretsema selle saatuse pärast
- ```
try (FileInputStream i = new
FileInputStream("file.inf")) {

}
```



1918

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

# Finally..

- Finally blokk võimaldab lõputegevusi, mis tehakse igal juhul, sõltumata sellest, kas kood läbis try või kukkus catchi
- Üldiselt ei kasutata eriti



1918

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

# Erindid - kokkuvõte

- Kaalu hoolikalt, kas erindit on vaja
- Kui erind luuakse, siis **ära ignoreeri!**
- Kontrollitud erind püüa alati kinni!



1918

**TALLINNA TEHNIKAÜLIKOOL**  
TALLINN UNIVERSITY OF TECHNOLOGY