



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

IDK2015 Objektorienteeritud programmeerimine Javas

Martin Rebane (martin.rebane@ttu.ee)

Eelmisel nädalal

- Testi tulemused:
 - Keskmine skoor 8.67 / 10.00



Skoop!

Access Levels

Modifier	Class	Package	Subclass	World
<code>public</code>	Y	Y	Y	Y
<code>protected</code>	Y	Y	Y	N
<i>no modifier</i>	Y	Y	N	N
<code>private</code>	Y	N	N	N

- <http://docs.oracle.com/javase/tutorial/java/javaOO/accesscontrol.html>



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Eelmisel nädalal

- Objektitüüpidele eraldatakse mälu objekti loomisel (new võtmesõnaga)
- Igal väli (muutuja) viitab konkreetsele objektile mälus
- Igal objektil on oma aadress (ID)
- Seda ID'd saab kasutada debugimiseks



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Eelmisel nädalal

- Java dokumentatsioonist leiate infot kõigi Java meetodite kohta, sh erinevad konstruktorid
- Eclipse pakub teile suure osa Javadocist kohe IDE sees
- Uurisime seda `Java.util.String` klassi näitel



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Teemad täna

- equals()
- Veel kollektsioonidest



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

equals() vs ==

- equals() on meetod, mis kontrollib kahe objekti sisulist võrdsust
- Objektide korral kontrollib == ainult seda, kas kaks viita viitavad samale objektile
- Objektide korral kasutage alati equals()



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

equals()

- Iga objektitüübi jaoks tuleb equals() üle kirjutada, vaikeimplementatsioon Object klassis kontrollib, kas tegu on sama objektiga
- Stringi (jt Java tüüpide) jaoks on Java arendajad selle töö teinud. Enda tüüpide jaoks peame ise equals()-i üle kirjutama



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

equals() omadused

- **Sümmeetria:** $a.equals(b)$ saab olla tõene ainult siis kui ka $b.equals(a)$
- **Refleksiivsus:** $a.equals(a)$
- **Transitiivsus:** kui $a.equals(b)$ ja $b.equals(c)$, siis $a.equals(c)$



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

hashCode

- equals() ja hashCode(): kui kirjutate üle ühe, tuleb kirjutada ka teine
- hashCode() - tagastab objektile unikaalse räsiväärtuse (hashi), mis ei tohi muutuda kui objekti olek ei muutu



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

`equals()` ja `hashCode` ühilduvus

- Kui:
 - a.`hashCode()` tagastatav räsi on identne b.`hashCode()` tagastavaga
- ..siis:
 - a.`equals(b)` peab olema tõene
- Kui hash on sama, siis on objektid identsed

Andmekogumid

- Massiivid (arrays)
 - String[]
 - int[]
 - Student[]
- Kolleksioonid
 - Kolleksioonid (Collections)
 - List, Set, Queue
 - Sõnastikud (Maps)
 - Map



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Andmekogumid

- Interface – nt List, Map. Otse koodis ei saa luua `new List<>` või `new Map<>`
- Üldine tüüp, alamtüüp
 - Kasulik kui on mitut erinevat tüüpi liste
 - Erinevatel tüüpidel on erinevad omadused, nt kiiruse omadused



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

HashMap<keyType, valueType>

- *key* – *value* paari hoidmine
- Kiire otsing *key* järgi
- *key* on unikaalne – kahte sama võtmega kirjet hoida ei saa
- ei taga lisamise järjekorra säilimist



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

HashMap

- Fail-fast – struktuuri ei tohi itereerimisel muuta
- ConcurrentModificationException
 - v.a remove() meetod
- Katsuge *key*'d mitte muuta!



1918

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY