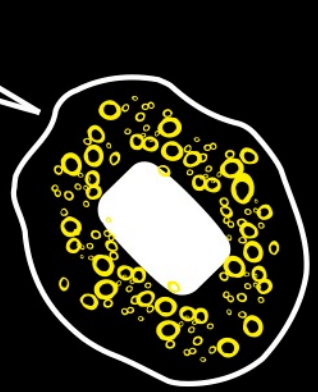
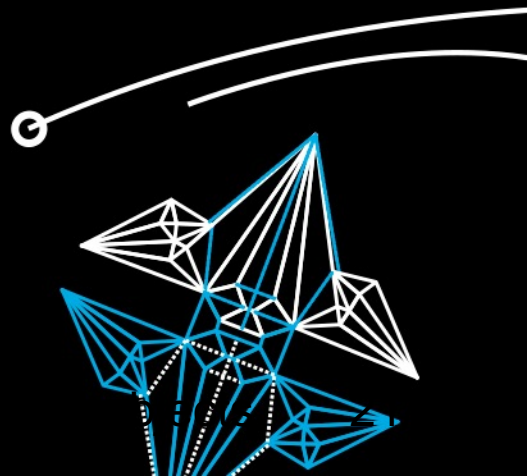
UNIVERSITY OF TWENTE.



Module 2 Programming Q&A 1

Lecturer: Tom van Dijk



YOUR QUESTIONS

Q1: Can you give a real-life analogy of the `static` keyword?

Something that is *shared* by **all** objects of the class, or that is *identical* for **all** objects of the class. Examples:

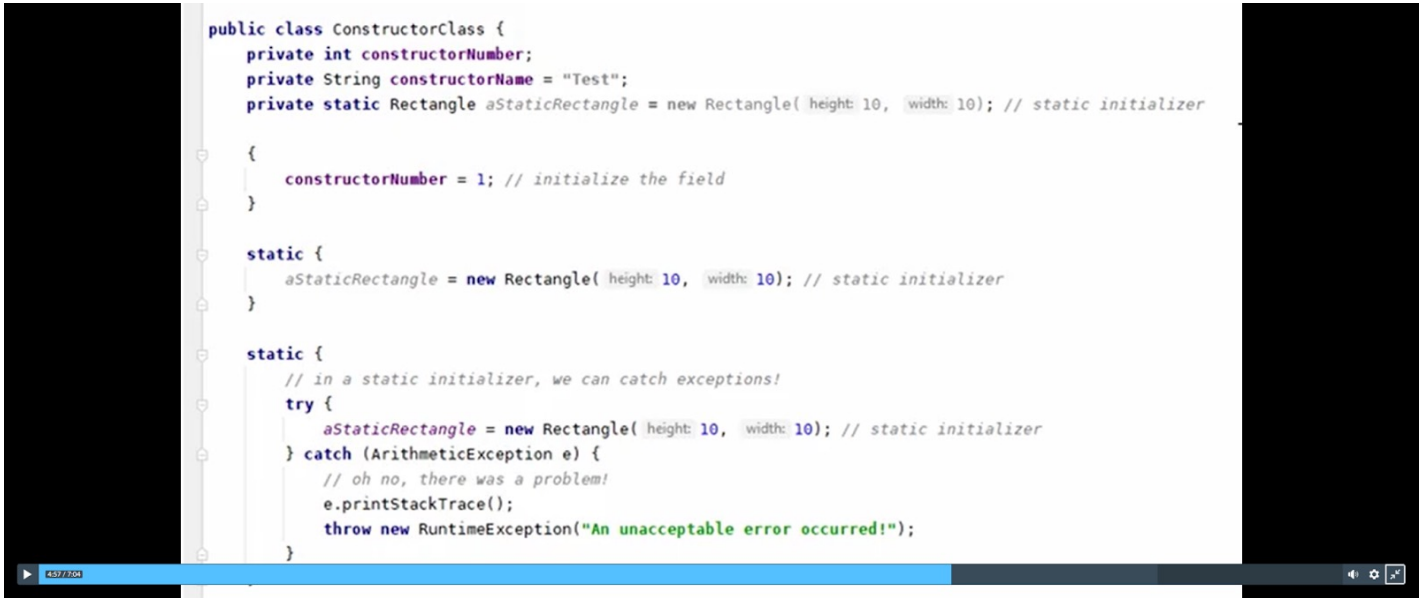
- Constants, such as: PI, or the size of the game board, etc.
- Mathematical functions, that don't require state.
- Precomputed lookup tables or caches

YOUR QUESTIONS

Q2: Why can't you access `this` from a `static` method?

Because static methods by definition are independent of specific objects. So they can only access other static methods and static fields.

YOUR QUESTIONS



In this example, if we had a longer program than this one given, if we tried at some point to change the value of aStaticRectangle, would we receive an error?

YOUR QUESTIONS

Q3: Can we change the value of a `static` field?

Yes, why not?

YOUR QUESTIONS

Q4: What is the difference between *constructors* and *initialization blocks*?

- Initialization blocks are executed before the constructor
- There can be multiple independent initialization blocks
- Constructors are invoked, initialization blocks are not
- Constructors have parameters, initialization blocks have no parameters

YOUR QUESTIONS

Q5: How do initialization blocks work?

- They’re code blocks inside the class
- They’re not used very often
- They’re executed just before the constructor

(Try it out with initialization blocks containing a println)

YOUR QUESTIONS

Q6: Why put every class in a separate java file?

- Easier to find
- Public classes **must** be in the file of the same name

YOUR QUESTIONS

Q7: do reference types always point to objects with primitive-type variables?

Reference types point to `null` or to an object. The object can have any kind of fields, including reference types, including references to the same class

YOUR QUESTIONS

Q8: Java automatically destructs objects. Should I set variables to `null`?

- Java “occasionally” runs *garbage collection*.
- Explicitly setting to `null`: only in specific circumstances.

QUIZ QUESTIONS

A class has a final field `private final int[] x = new int[4];`

- Is it allowed to change the contents like: `x[0] += 5;`

A class has a final field `private final List x = new ArrayList();`

- Is it allowed to change `x` to a different list?
- Is it allowed to change the list by adding or removing items?

QUIZ QUESTIONS

When does Java make a *default constructor* (no parameters, empty body)

A. Always
B. Only if there is no constructor without parameters yet
C. Only if there is no constructor yet
D. Only if all defined constructors are private

QUIZ QUESTIONS

What is the default implementation of `equals` (comparing two objects)?

A. Throw a `NotImplementedError` runtime exception
B. Compare the contents of all fields using `==`
C. Compare the contents of all fields using `equals`
D. Compare the two objects if they are the same reference
E. Always returns `false`