

# SMA: Software Modeling and Analysis

*Practical Session*  
***Week 05***

# Assignment 05

## ***Discussion***

**You have to attend the lecture to reveal such slides.\***

---



*\*Disclaimer:*

*The content that has been shown on this slide is irrelevant for the exam.*

# A05 - Exercise 01 | General questions

- Is code reading a problem? Argue the answer.  
*Yes it is! Developers spend usually more than 50% of their time on code reading.*
- Give an example where a custom tool improved the productivity in addressing a problem or issue.
  - status indicators* → *awareness of project state*
  - customized IDE plug-ins* → *report a variety of issues*
  - DIY tools* → *help in fixing things...*

# A05 - Exercise 02 | Smalltalk coding

- ***Write an inspector extension*** that prints the instance of a `DateAndTime` object in the following format:  
YYYY-MM-DD HH:MM

```
DateAndTime>>gtInspectorHumanReadableIn:  
  composite <gtInspectorPresentationOrder: 10>  
  composite text  
  title: 'Human readable';  
  display: [ self asStringYMDHM ]
```

# A05 - Exercise 02 | Smalltalk coding

- ***How many inspector extensions exist*** that use a table presentation? Implement code that counts those extensions.

*27. The required code:*

```
(#gtInspectorPresentationOrder:  
  pragmas intersection: #table senders) size
```

# A05 - Exercise 03 | BONUS questions

- **Enumerate at least two different embedded artifacts** that are essential for a live document.

*CompiledMethod.extension* → enables inline code in inspector  
*Color.extension* → shows color panels in inspector

- What is the **main difference between extensions** of the old and the new inspector?

*The old extensions make use of the <gtView> pragma instead of the <gtInspectorPresentationOrder> pragma.*

# Assignment 06

***Preview***



# A06 - Exercise 01 | Metrics

## General *knowledge*.

- What is cyclomatic complexity? Explain!
- Which other metrics do you know?
- Do metrics always express problems?
- How and when are nowadays metric checks in place?

# A06 - Exercise 02 | Evaluation of metrics

## Pharo *coding*.

### First, write a query to...

- retrieve all classes with more than 42 methods.
- find all methods with a CYCLO larger than 84.

### Then respond...

- What methods suffer from such a complexity?
- Is 84 a large CYCLO value?

# A06 - Exercise 03 | Evaluation of metrics BONUS

**More Pharo *coding*.**

**Write a query to...**

- obtain classes that call deprecated methods.
- obtain all attributes that show certain properties, but are not declared as final.
- obtain a list of methods that make more than one call to deprecated classes.