Labb 1: Vad, hur, och varför?

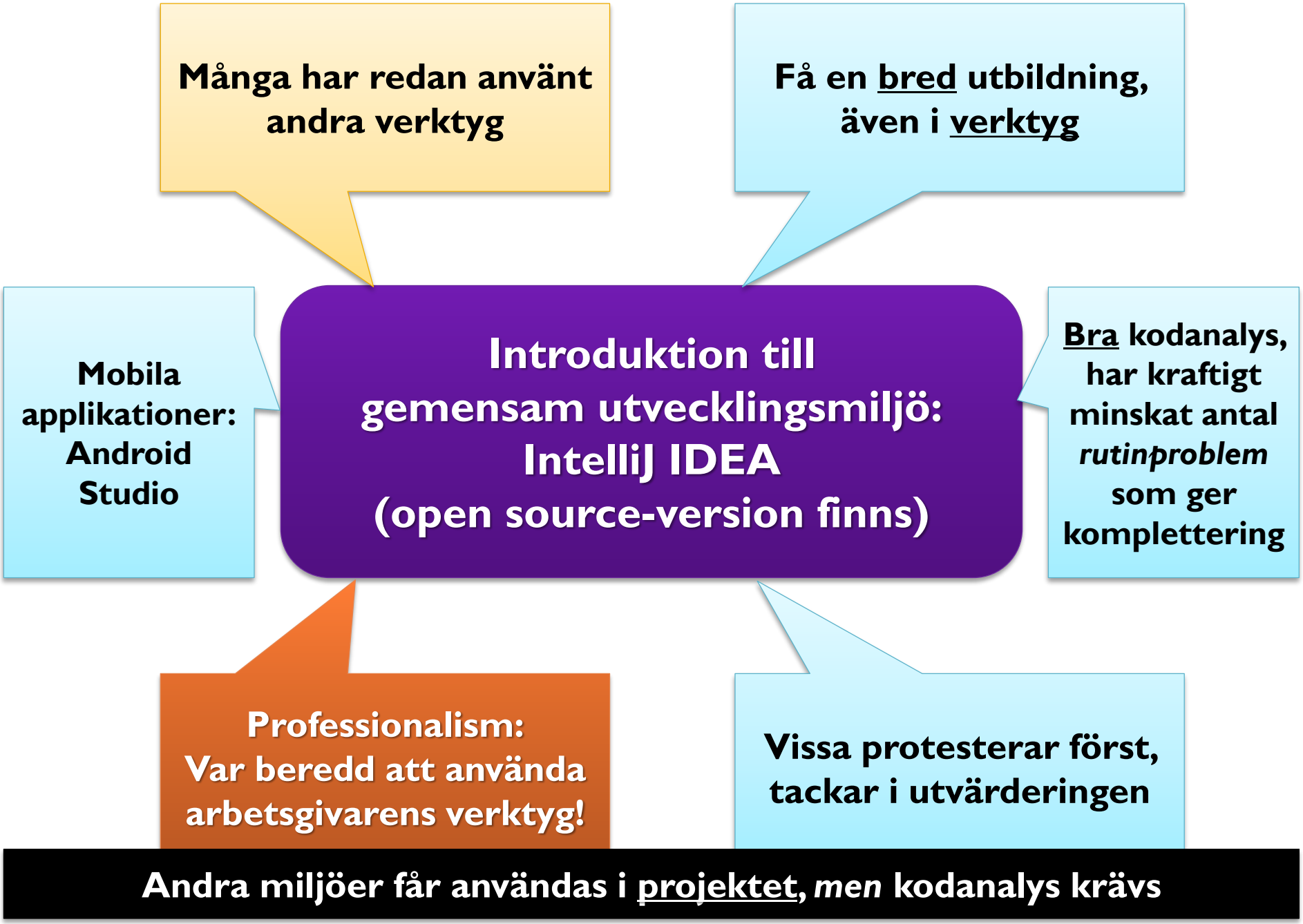
jonas.kvarnstrom@liu.se – 2017

"En sak i taget":
Öva grunder innan det blir mer komplicerat

Starkt önskemål från studenter:
Prova på kontrollstrukturer

Labb 1:

Intro till grunder i Java helt utan objektorientering


Många har redan använt andra verktyg
Få en bred utbildning, även i verktyg
Mobila applikationer: Android Studio
Introduktion till gemensam utvecklingsmiljö: IntelliJ IDEA (open source-version finns)
Bra kodanalys, har kraftigt minskat antal *rutinproblem* som ger komplettering
Professionalism: Var beredd att använda arbetsgivarens verktyg!
Vissa protesterar först, tackar i utvärderingen
Andra miljöer får användas i projektet, *men* kodanalys krävs

Varningar för möjliga problem (600 ”inspektioner”)

```
try {
    new FileInputStream(f).read(data, 0, len);
} cat
    e.printStackTrace();
}
```

Result of 'FileInputStream.read()' is ignored more... (Ctrl+F1)

```
    IncMor incMor = new IncMor(expectedSize);
    Map<Node, SNode> iNodeToSNode = new HashMap<>(expectedSize);
```

Contents of collection 'iNodeToSNode' are updated, but never queried more... (Ctrl+F1)

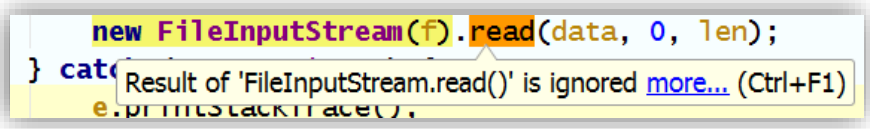
```
// This means the we first see if the wait should be converted by the unco
final int x = -conditioning.outNegCon.value;     // this is the x in Morri
```

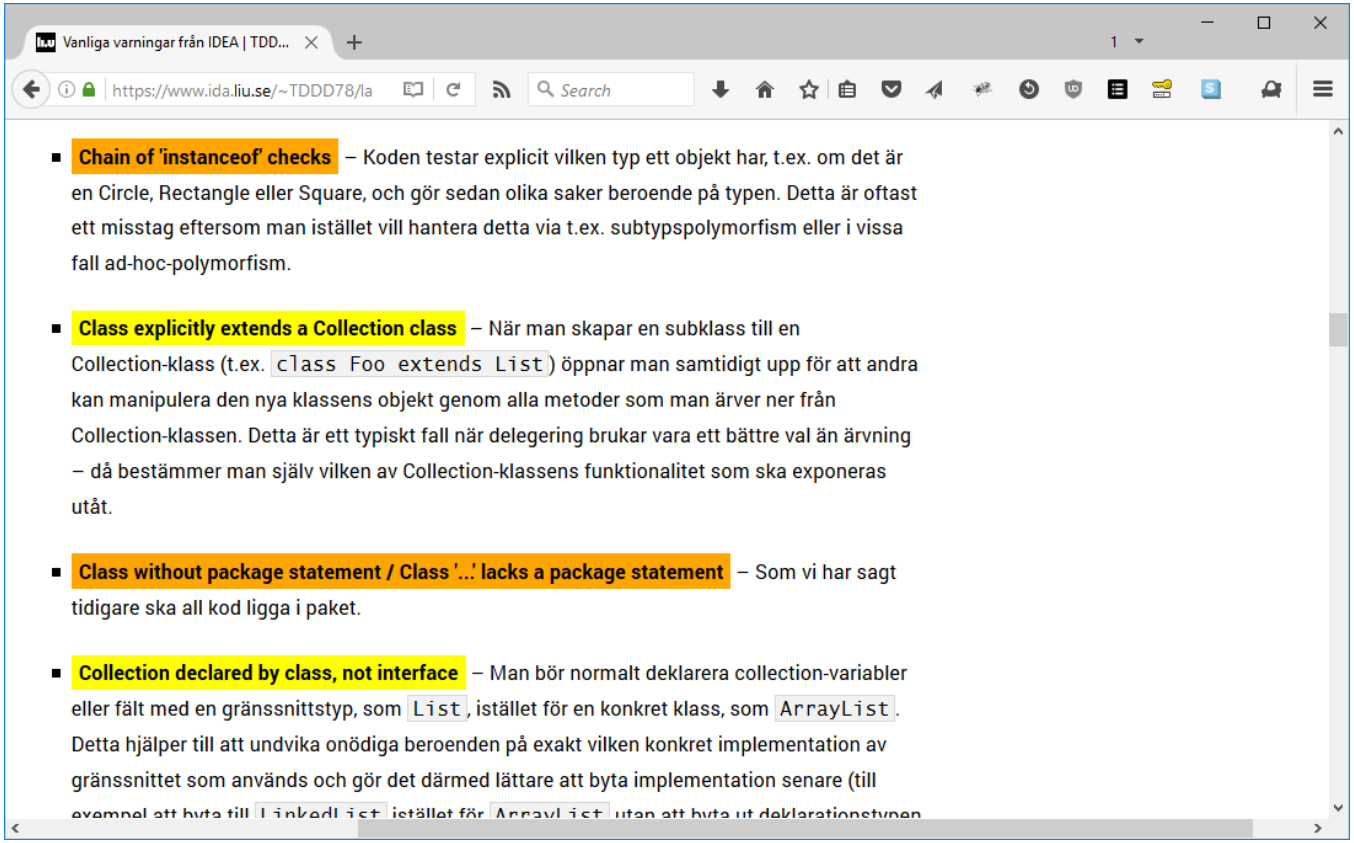
Dereference of 'conditioning.outNegCon' may produce 'java.lang.NullPointerException' more... (Ctrl+F1)

Fixa buggar, undvik kompletteringar, **lär er** direkt från varningarna!

Om du inte (är säker på att du) förstår varningen

1. Förklaringar från IDEA
2. Förklaringar på kurswebsidorna





3. Fråga handledaren
4. Fråga examinatorn

Smart kodkomplettering – Ctrl+Shift+SPACE

```
import java.util.Random;

public final class RandomInteger {

    public static void main(String[] args){
        log("Generating 10 random integers in range 0..99.");

        Random randomGenerator = new R
        for (int idx = 1; idx <=
            int randomInt = rand
            log("Generated : " + randomInt);
        }

        log("Done.");
    }

    private static void log(String aMessage){
        System.out.println(aMessage);
    }
}
```

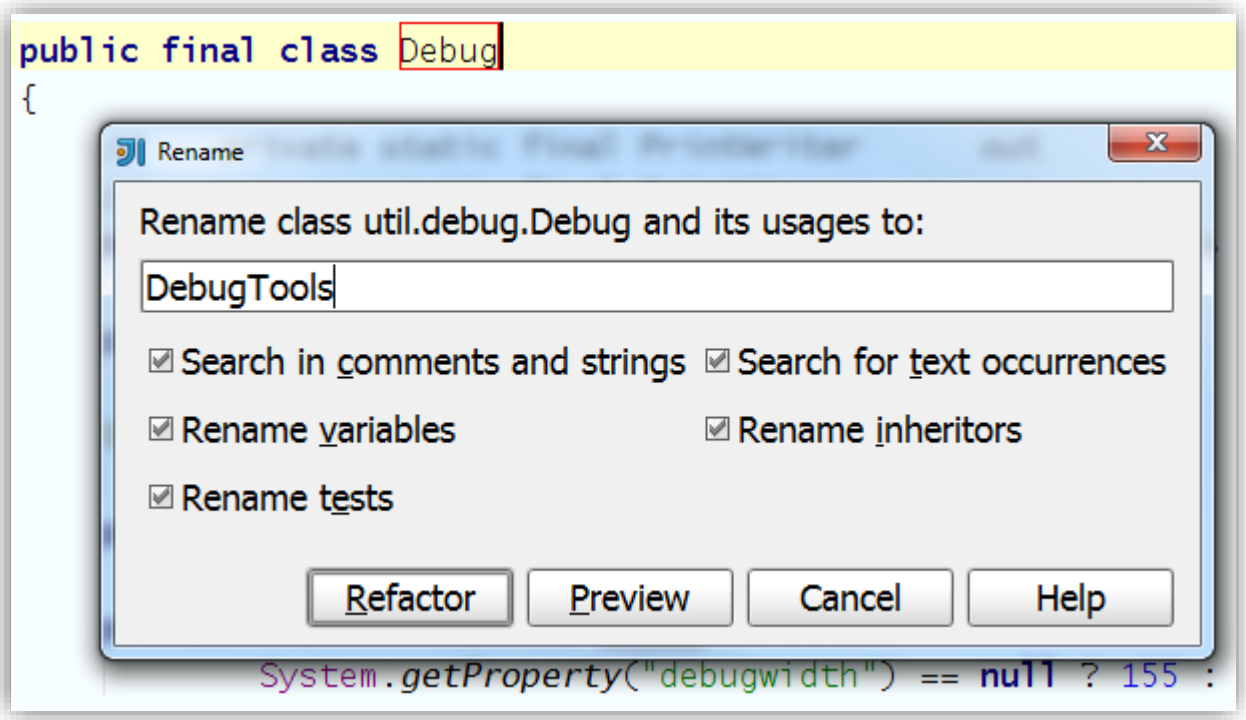
```
Random (java.util)
SecureRandom (java.security)
```

LÄS: *http://jetbrains.dzone.com/articles/top-20-code-completions-in-intellij-idea*

Stöd för refactoring

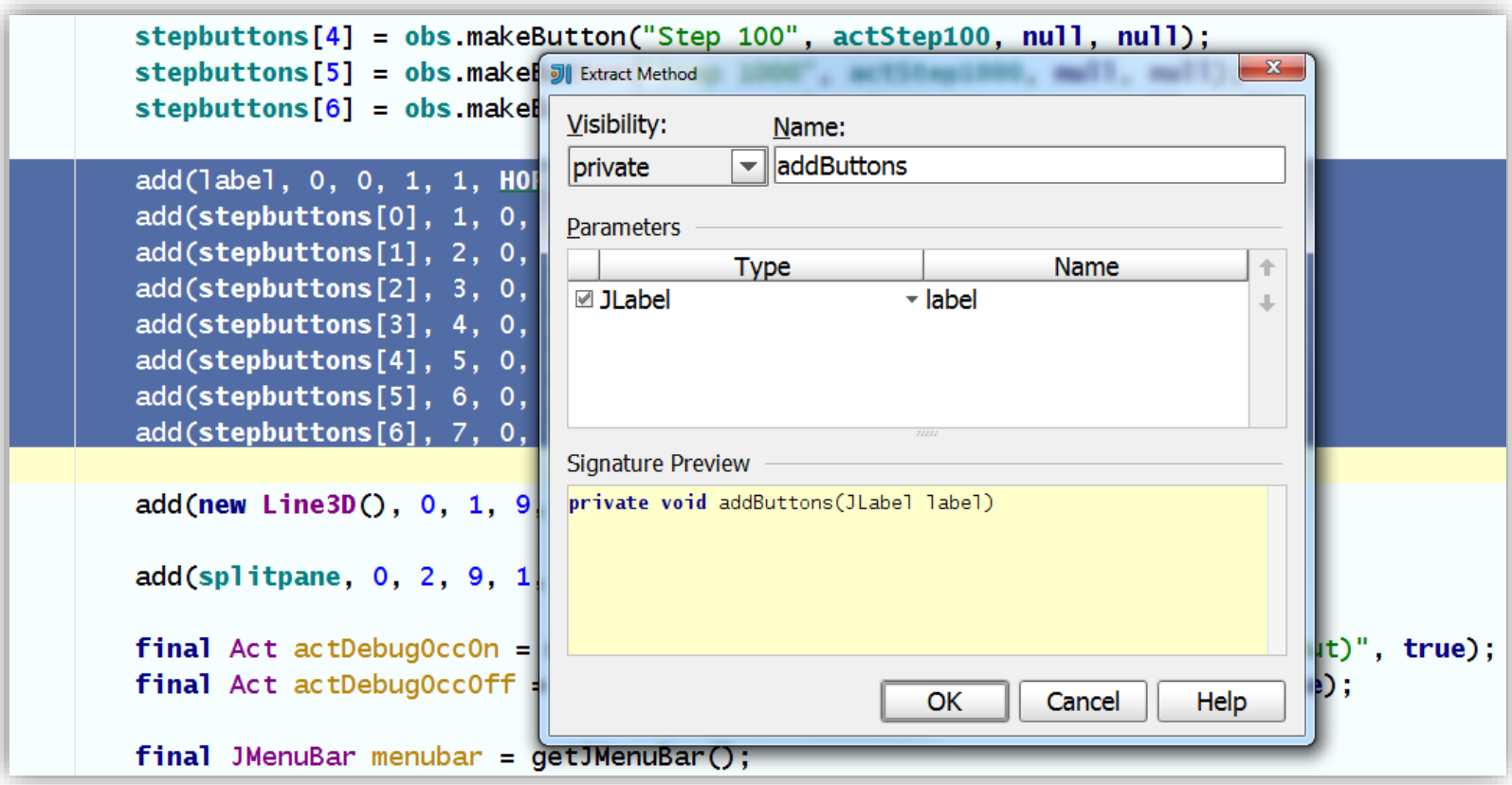
- Omstrukturering av koden
  - Ofta för att förbättra design, läsbarhet, …
  - Viktigt i större projekt

- Exempel: Döp om
  - "Sök och ersätt"? Kan ersätta något orelaterat också!



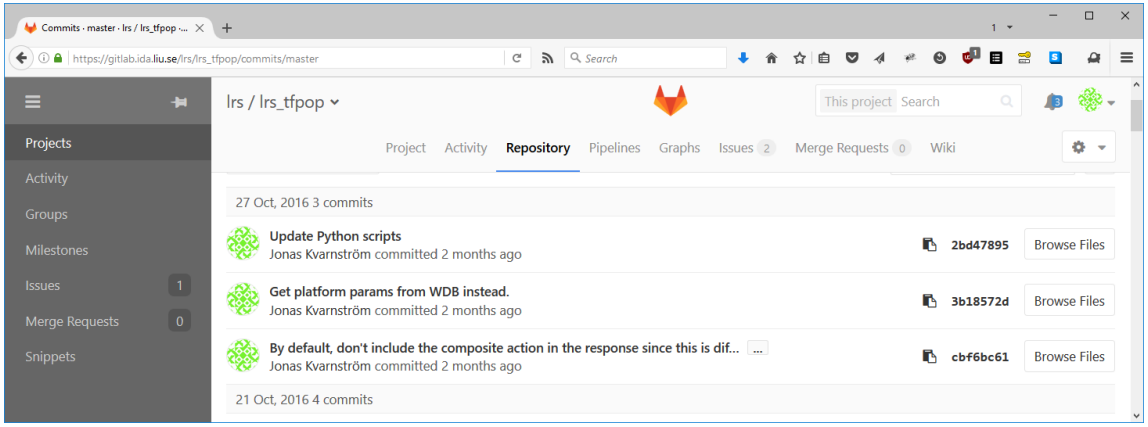
Stöd för refactoring

- Gör en metod (funktion) för många olika saker?
  - Hitta en meningsfull del, extrahera en underfunktion!
  - IDEA tar hand om parametrar, …

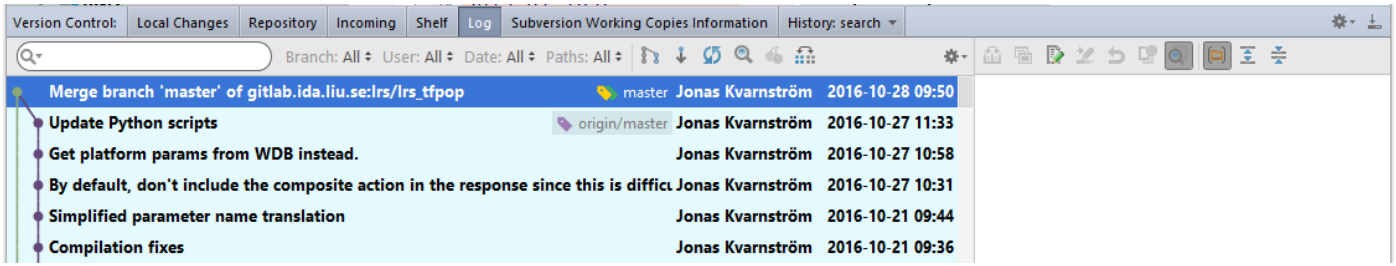


Börja (eller fortsätt) använda versionshantering: Git

- Skapa ett **filarkiv** där varje ny version av koden sparas
  - **Checka in** efter att en övning avklarats, en feature implementerats, …
  - Se **historiken** på gitlab.ida.liu.se…



  - …eller i IDEA



  - Slipp **manuell** hantering: "project", "project-old", "project-newer", …

Börja (eller fortsätt) använda versionshantering: Git

- Se vad som ändrats i en fil, när, av vem…

```
5  2013-01-18 Persson
6  2014-06-13 Persson      #include "lrs_srvs_mp/MPPlanM.h"
7  2014-06-13 Persson      #include "lrs_srvs_mp/PFMotionPlanInfo.h"
8  2016-10-27 Kvarnström   #include "lrs_srvs_wdb/WDBGetPlatformParams.h"
9  2013-01-18 Persson
10 2013-01-18 Persson      #include <iostream>
11 2013-01-18 Persson      #include <sstream>
12 2013-01-18 Persson      #include <fstream>
13 2013-01-18 Persson
14 2016-10-21 Kvarnström   #pragma clang diagnostic push
15 2016-10-21 Kvarnström   #pragma ide diagnostic ignored "OCUnusedGlobal
```

- …och återställ ändringar som inte checkats in.

```
26
27      Rollback (Ctrl+Alt+Z)  int i = 0; i < 200; i++) addToQueue(i, circ, plain);
28                             i = 0; i < 100; i++) Assert.assertEquals(circ.po
29          for (int i = 0; i < 400; i++) addToQueue(i, circ, plain);
30          for (int i = 0; i < 200; i++) Assert.assertEquals(circ.po
31          for (int i = 0; i < 200; i++) Assert.assertEquals(circ.po
32      }
```

Börja (eller fortsätt) använda versionshantering: Git

- Infoga ändringar från annan dator, annan projektmedlem
  - VCS ➔ Update Project, inte skicka zip-filer
  - Skriv inte över varandras ändringar

- Lämna in kod via Git (Tetris/projekt)
  - Högerklicka rätt version, sätt en etikett (tag), skicka *etiketten* via epost – inte en zip-fil

Nytt för i år!
Informera oss direkt vid problem / oklarheter

Annars förskjuts hela kursen – och andra kurser!

Ni får bättre hjälp om handledarna kan fokusera på en labb i taget

Håll takten!

Labb 1 bör vara klar måndag/tisdag
Deadlines finns på nätet

Professionalism!

Eget arbete utanför schema:
4h/vecka för labbar = 16h
8h/vecka för Tetris = 24h

Labb 2:

Intro till objekt och klasser i Java
(efter föreläsning 3)

Labb 3:

Hierarkier av objektklasser
(en *hund* är en sorts *djur*)

Labb 4:

Hur man blir av med vissa problem

Labbar:
Vad är fokus?

När är ett program (en inlämning) bra?

När körningen "ger rätt resultat"?

Nej!

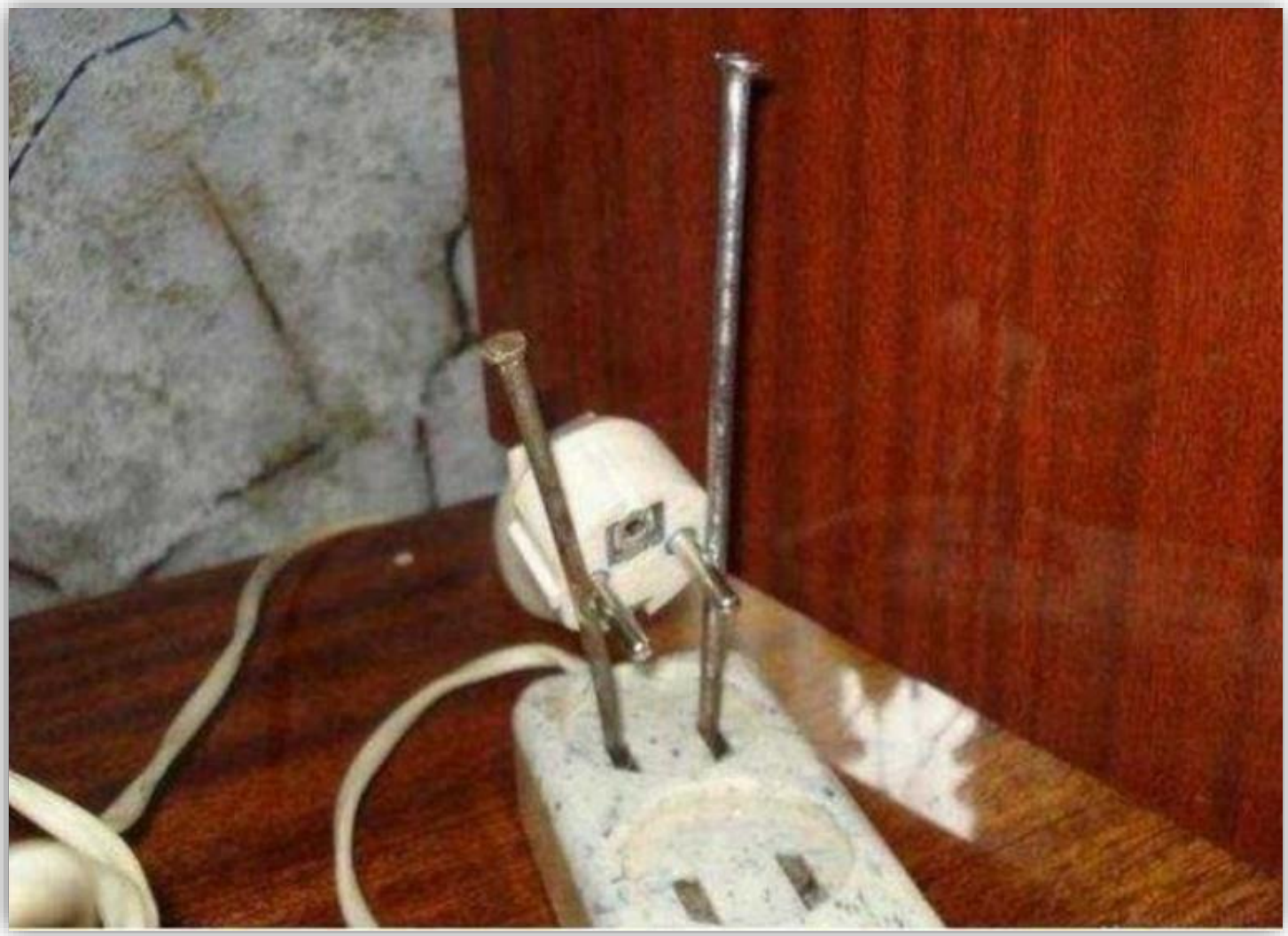
När programkoden är tydlig, välstrukturerad och visar att ni vet vad ni gör!

Kvalitet (1)

- Ser det ut så här,
  blir vi misstänksamma…

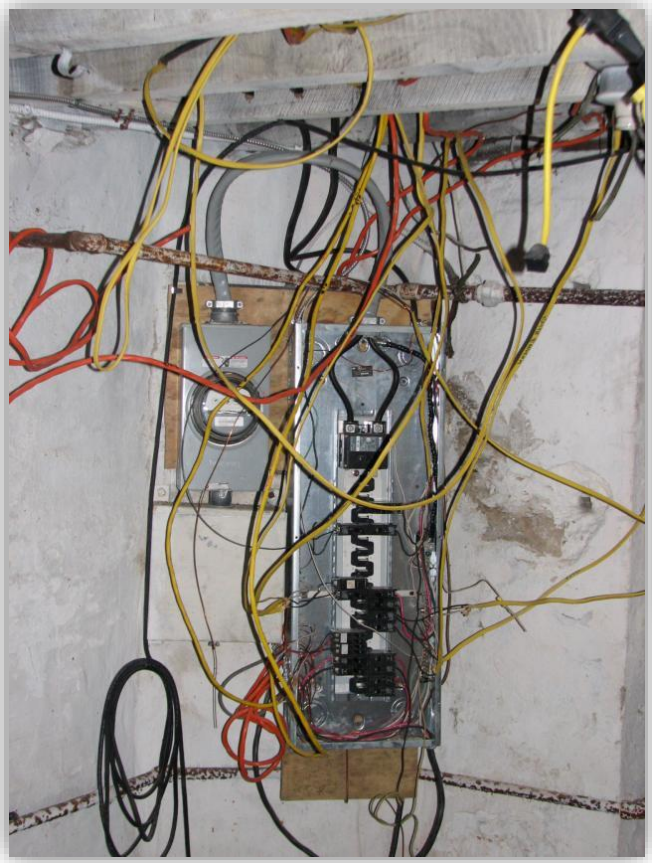
Kvalitet (2)

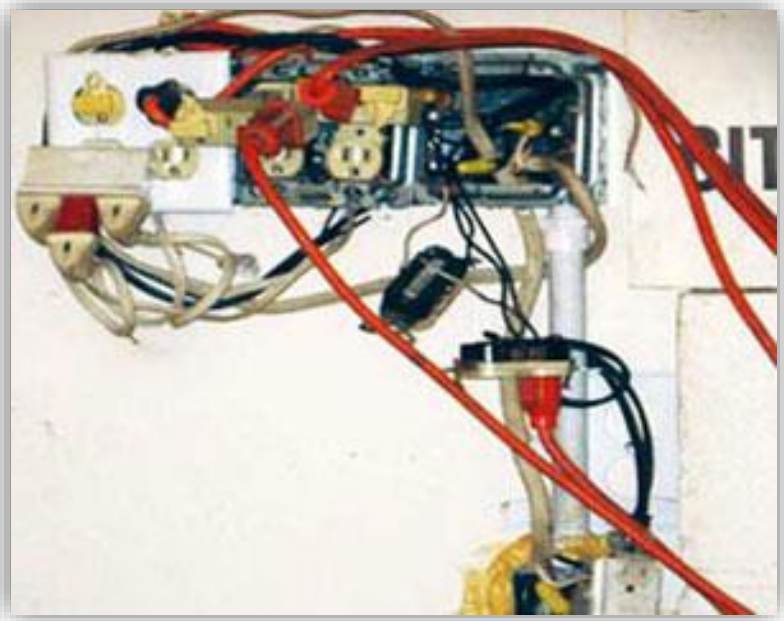
- Har ni kopplat så här, är det farligt…



Kvalitet (3)

- Ser det ut så här, undrar vi vad ni egentligen har gjort
  - **Även** om ni faktiskt fick lampan att lysa





Exempel

jonkv@ida

- Ett extremt exempel:
  - ```
    class Div{static $1 $_;class $1{void _$(String $_){System.//
    out.print($_);}$1(){_();}void _(){int _,$,_$,$$,__,ä=(1<<5),
    å=100,ö=12,åö=ä*ö;å-=1<<4;while(åö>0){for($$=_$=_=__=$=(int)
    å;_$>($$-(1<<2));_$(""+(char)(_$)),_$-=1<<1)for(_=$=__-9;_>(
    $-6);_-=1<<1,_$(""+(char)(_$!=å?_:ä)));char S$=(char)(å+ö+1)
    ;_$("te"+S$+(char)(ö+(int)S$));åö--;}}}Div(){$_=new $1();}
    public static void main(String []$){Div å=new Div();}}
    ```

Skriver ut "TIGERteam" 384 gånger
    ```

Exempel

jonkv@ida

- Ett alternativ :

```
// https://codegolf.stackexchange.com/questions/22533/weirdest-obfuscated-hello-world
import java.io.ByteArrayOutputStream;

public class MysteryCode {
    public static void main(String[] unused) throws Exception {
        ByteArrayOutputStream stoned = new ByteArrayOutputStream(20480);
        int[] magic = {104, 116, 116, 112, 58, 47, 47, 98, 105, 116, 46, 108, 121,
                       47, 49, 98, 87, 119, 51, 75, 111};
        for (int weird : magic) stoned.write(weird);
        int crazy, unknown = 0;
        java.io.InputStream wtf = new java.net.URL(stoned.toString()).openStream();
        while((crazy = wtf.read()) != -1) stoned.write(crazy);
        for (int strange : stoned.toByteArray()) {
            if (unknown == 2) {
                if (strange == 38) break;
                System.out.print((char) strange);
            } else if (17 + (unknown + 1) * 21 == strange) {
                unknown++;
            }
        }
    }
}
```

Skriver ut ”Hello, World”

Orsak och verkan

jonkv@ida

~~Programmet fungerar (ger rätt utmatning, …)~~ → ~~Jag kan programmera!~~

Jag kan programmera! → Programmet fungerar (ger rätt utmatning, …)

Jag kan programmera! → Programmet är välskrivet, lätt att förstå, välstrukturerat, …

Jag kan programmera! → Jag förstår varför jag gör på ett visst sätt

Kvalitet: Fokus

- Skriv **bra** kod!
  - Lättläst
  - Bra namngivning
  - Strukturerad, modulär
  - Väldokumenterad och kommenterad – där det behövs mest!
- Extremt viktigt för:
  - Hur programmet kan utökas i framtiden
  - Hur andra kan förstå programmet
  - Hur robust det är när fel uppstår
  - …

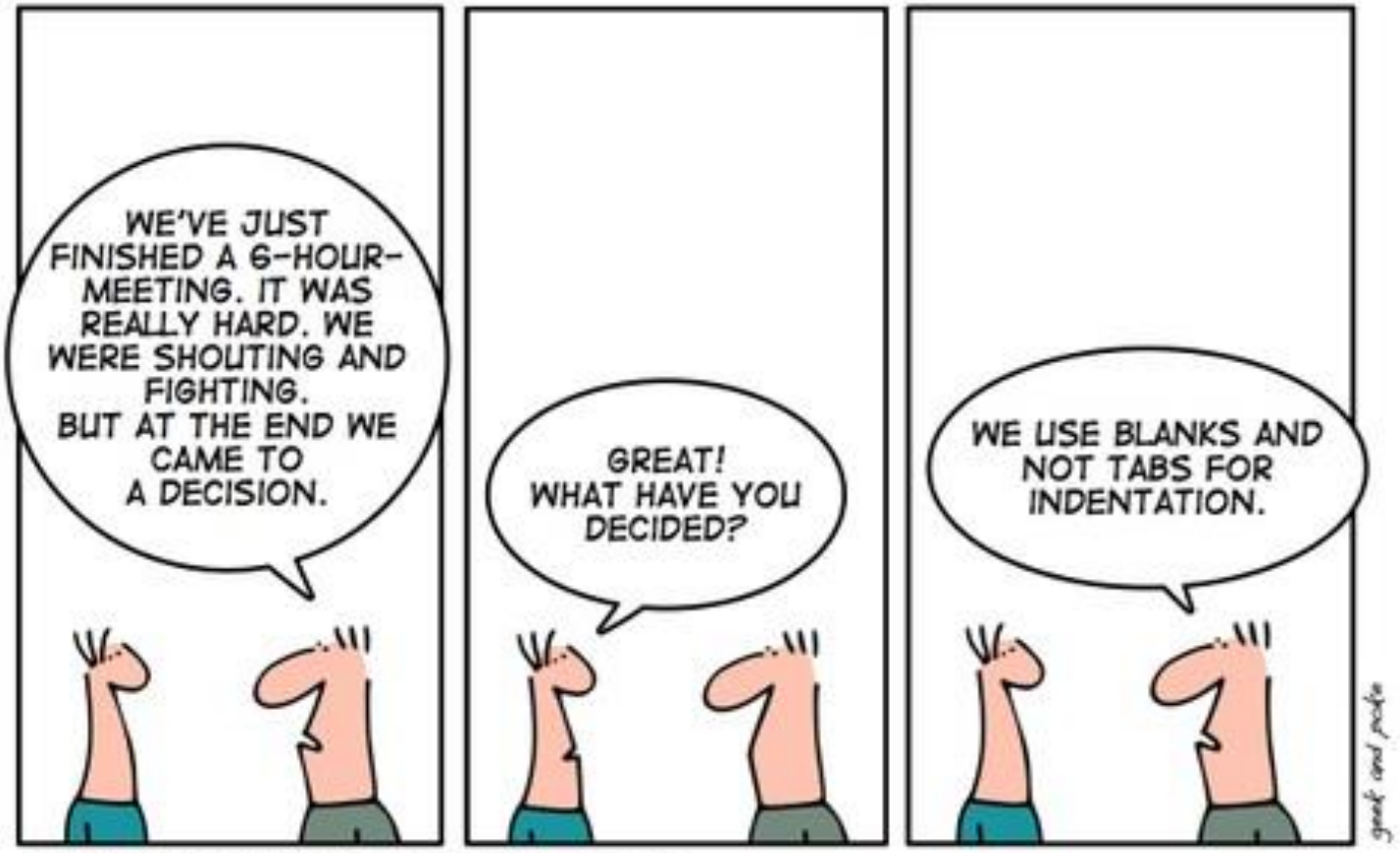
Utveckling är ofta 20% av arbetet…

Underhåll är 80%!

Kvalitet: Icke-fokus

jonkv@ida

- Inte fokus på petitesser
  - Indenteringsstil, …



Hur ska man skriva kod?

- **Tips:**
  - **Always code as if the person who ends up maintaining your code is a violent psychopath who knows where you live.**

  - *"I usually maintain my own code, so the as-if is true."*
    – http://c2.com/cgi/wiki?CodeForTheMaintainer