Technische Hochschule Rosenheim

Modul - Fortgeschrittene Programmierkonzepte

Bachelor Informatik

01 - Einführung

Prof. Dr. Marcel Tilly

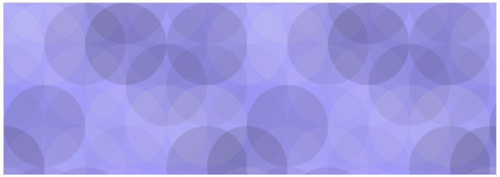
Fakultät für Informatik, Cloud Computing

Vers. 7.10.2021

Before we start..

Tiny logo competition!

Let's try to find a new logo and replace the boring one...



Price

Milka
MILK CHOCOLATE CONFECTION WITH WHOLE HAZELNUTS
NET WT 3.52 OZ (100g)
100% ALPENMILCH SCHOKOLADE
Ganze Haselnüsse
Whole Hazelnut

- Send your proposals to me until Wednesday!
- Voting starts until Friday eol (=end of lecture)!

Organisatorisches

- Material:
  - GitLab: https://inf-git.fh-rosenheim.de/inf-fpk
  - Learning Campus: https://learning-campus.th-rosenheim.de/course/view.php?id=1482
  - **WICHITG: Slides, Skript, Übungen …auf Englisch!** ☺
- Einschreiben unter **Fortgeschrittene Programmierkonzepte (INF-B3), WiSe21/22** (Selbsteinschreibung ohne Schlüssel!)

Vorlesung

- Vorlesungstermin: Freitags, 09:45 - 11:15 in R0.02
- Übungen
  - Tutor: Simon Grad
  - Donnerstags, 2./3./4. Stunde
  - Übung 1: ONLINE
  - Übung 2: Präsenz
  - Übung 3: Präsenz
  - Gruppenwahl über Learning Campus
- Kommunikation über Discord (**#fpk INF**)

Leistungsnachweis

Klausur!

- schriftliche Prüfung (SP, 90 Minuten)
- am Ende des Semesters
- erlaubt ist ein Buch mit ISBN Nummer
- Anmeldung über OSC
- Was kommt dran?
  - Alles was in der Vorlesung dran war!

Projects

I would like to work with you on small coding projects!

- Form a team (2-4 members)
- Create **your own idea** and drive your **own** project
- Improve your coding skills
- Collaborate with **Simon** and **me**
- Use GitLab or Github
- Show at the end your result (if you want!)

Ideas: Footballmanager, lightweight Editor, Notes, TicTacToe, Trainingsplaner ...

Lernziele

Aus dem Modulhandbuch

Die Studierenden …

- … **vertiefen** ihre Kenntnisse in der objektorientierten Programmierung am Beispiel einer geeigneten Programmiersprache (hier: Java!)
- … können die Möglichkeiten und Gefahren der objektorientierten Programmierung **beurteilen**.
- … **sind befähigt**, alle wichtigen Programmierkonzepte für das Programmieren im Großen im Sinne der Komponentenorientierung anzuwenden.
- … **erarbeiten sich die Grundlagen** der funktionalen Programmierung und deren Anwendungsgebiete.

Review

Programmieren 1

- Imperative programming in C
- Constants, Variables, Expressions, Functions, I/O
- Data structures (fields, arrays, lists)
- Pointer ☺

Review

Programmieren 1

- Imperative programming in C
- Constants, Variables, Expressions, Functions, I/O
- Data structures (fields, arrays, lists)
- Pointer ☺

Programmieren 2 (OOP)

- Objekt-oriented Programming (OOP) in Java
- Classes and Objects
- Interfaces and Inheritance
- Exceptionhandling

Agenda of FPK

see http://matworx.org/index.html?inf-fpk/README

or in Learning Campus

Agenda für heute

1. *Inform:*
   Your trusted advisors: Google -- SO -- Java Docs -- Google Translate
2. *Memorize:*
   The git version control system (https://git-scm.com/)
3. *Automate:*
   The Gradle build tool (https://gradle.org/)
4. *Organize:*
   The IntelliJ IDEA (https://www.jetbrains.com/idea/)
5. *(Optional) Collaborate*
   Practice cross-repository pull requests and learn about *continuous integration*

SO = stackoverflow

Exercise 1

Which version control systems do you know?

Version Control

Git

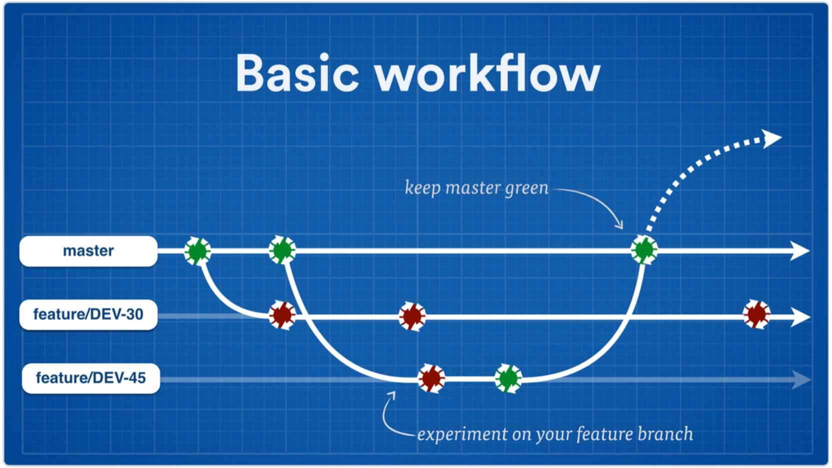
- Git is a distributed version-control system for tracking changes in source code during software development.
- It is designed for coordinating work among programmers, but it can be used to track changes in any set of files
- Git is the *de-facto* state of the art version control system.
- Some of you might remember CVS (concurrent versions system) or subversion.
- Generally speaking, you should always use a version control system (VCS) when working on code, so you can keep track of changes.
- Print and laminate: https://services.github.com/on-demand/downloads/github-git-cheat-sheet.pdf
- For the more visual: http://ndpsoftware.com/git-cheatsheet.html
- If you run into a mess (and you will): http://justinhileman.info/article/git-pretty/git-pretty.png

Exercise 2

Why is versioning so important?

Git and feature branches

https://www.atlassian.com/continuous-delivery/continuous-delivery-workflows-with-feature-branching-and-gitflow



- Git Guide: https://rogerdudler.github.io/git-guide/
- Git != GitHub

Automate

Technische
Hochschule
Rosenheim

The Gradle Build Tool (GBT)

Gradle is an open-source build-automation system that builds upon the concepts of Apache Ant and Apache Maven and introduces a Groovy-based domain-specific language instead of the XML form used by Apache Maven for declaring the project configuration.

Gradle Build Tool

https://gradle.org

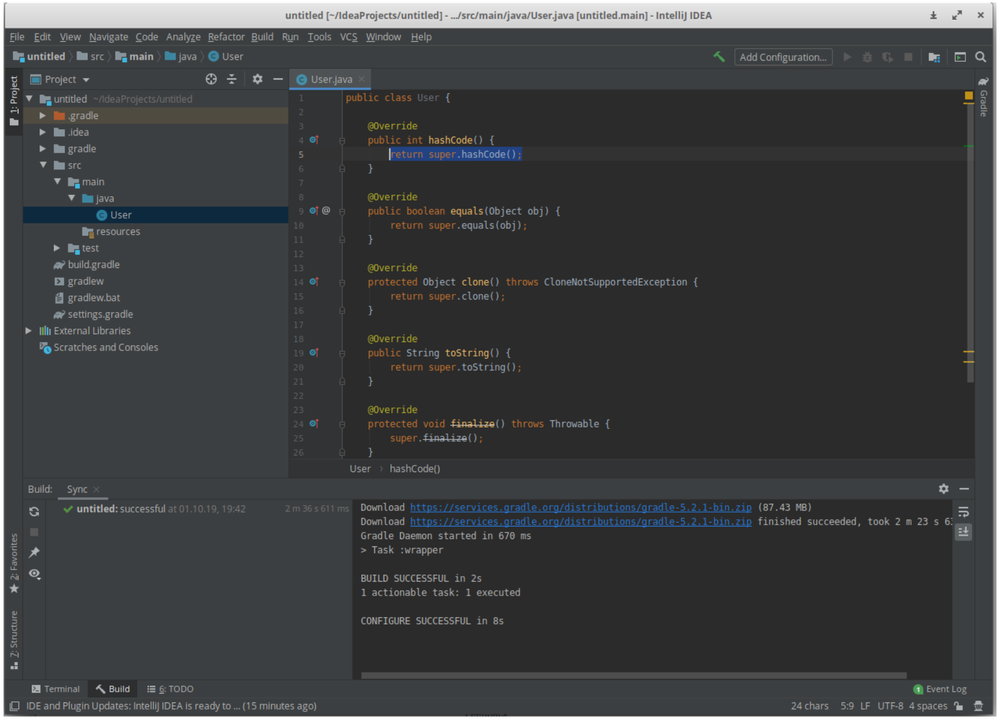
- `gradle init --type java-application` to bootstrap a project
- `./gradlew build` to use the [Gradle wrapper]() to be independent of locally installed Gradle
- `apply plugin: 'eclipse'` and `./gradlew eclipse` to generate Eclipse project files
- `apply plugin: 'idea'` and `./gradlew idea` to generate IntelliJ files (note: these are file-based project descriptions, not the new directory based `.idea/*`)

Exercise 3

Why is automation so important?

IntelliJ

https://www.jetbrains.com/idea/



```java
public class User {

    @Override
    public int hashCode() {
        return super.hashCode();
    }

    @Override
    public boolean equals(Object obj) {
        return super.equals(obj);
    }

    @Override
    protected Object clone() throws CloneNotSupportedException {
        return super.clone();
    }

    @Override
    public String toString() {
        return super.toString();
    }

    @Override
    protected void ~~finalize~~() throws Throwable {
        super.~~finalize~~();
    }
```

✓ untitled: successful at 01.10.19, 19:42 2 m 36 s 611 ms

```
Download https://services.gradle.org/distributions/gradle-5.2.1-bin.zip (87.43 MB)
Download https://services.gradle.org/distributions/gradle-5.2.1-bin.zip finished succeeded, took 2 m 23 s 6
Gradle Daemon started in 670 ms
> Task :wrapper

BUILD SUCCESSFUL in 2s
1 actionable task: 1 executed

CONFIGURE SUCCESSFUL in 8s
```

IDE and Plugin Updates: IntelliJ IDEA is ready to ... (15 minutes ago) 24 chars 5:9 LF UTF-8 4 spaces

Collaborate

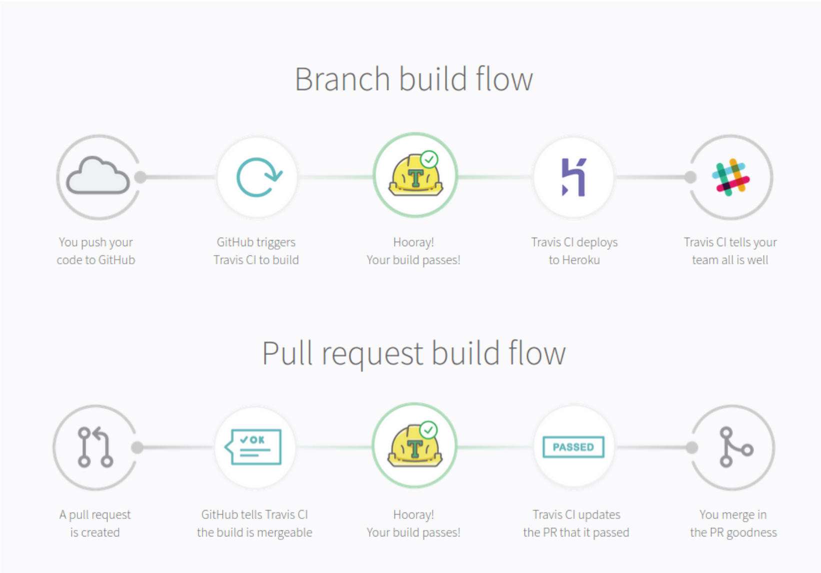
- Collaboration means splitting the work
- Teamwork means working together
- Use feature branches and automated tests (JUnit)
- Use automated build and test runner

=> **CI/CD**

- CI and CD are two acronyms that are often mentioned when people talk about modern development practices.
- CI is straightforward and stands for **continuous integration**: A practice that focuses on making preparing a release easier.
- CD can either mean **continuous delivery** or **continuous deployment**.

Travis CI

Travis CI is a hosted continuous integration service used to build and test software projects hosted at GitHub. Travis CI provides various paid plan for private projects, and a free plan for open source. (https://travis-ci.org/)


Branch build flow
You push your code to GitHub
GitHub triggers Travis CI to build
Hooray! Your build passes!
Travis CI deploys to Heroku
Travis CI tells your team all is well
Pull request build flow
A pull request is created
✓OK
GitHub tells Travis CI the build is mergeable
Hooray! Your build passes!
PASSED
Travis CI updates the PR that it passed
You merge in the PR goodness

Jenkins

- An open source automation server
- Jenkins provides hundreds of plugins to support building, deploying and automating any project.
- https://www.jenkins.io/

GitLab

- GitLab is a web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking and continuous integration and deployment pipeline features, using an open-source license, developed by GitLab Inc.
- https://about.gitlab.com/

.gitlab-ci.yml:

```
stages:
  - deploy

deploy:
  stage: deploy
  image: mcr.microsoft.com/azure-cli:2.9.1
  script:
    - apk add rsync
    - mkdir build
    - rsync -av --progress --exclude="build" --exclude=".*" --exclude="*.pptx"
    - az storage blob delete-batch -s "\$web" --pattern "inf-fpk/*"
    - az storage blob upload-batch -d "\$web\inf-fpk" -s ./build
  only:
    - master
```

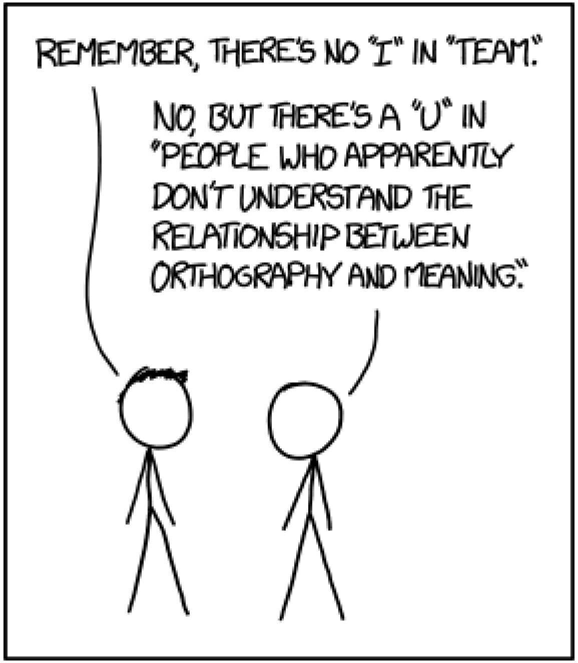
Comparison

- https://stackshare.io/stackups/gitlab-ci-vs-jenkins-vs-travis-ci#pros
- We are going to use **GitLab**!
- We (TH Rosenheim) are hosting our own server: https://inf-git.fh-rosenheim.de
- Ensure that you have access!

Summary

- We will look into advanced programming concepts in Java (starting next week!)
- We will use professional software engineering tools
    - Git
    - IntelliJ Idea
    - Gradle
    - GitLab
- Let's try to have fun!

Final thought!


REMEMBER, THERE'S NO "I" IN "TEAM."
NO, BUT THERE'S A "U" IN "PEOPLE WHO APPARENTLY DON'T UNDERSTAND THE RELATIONSHIP BETWEEN ORTHOGRAPHY AND MEANING."