

Anleitungen und Infos

Programmieren Tutorium Nr.12

Aleksandr Zakharov | 11. November 2025

Zusätzlicher Foliensatz

- Im Laufe des Studiums begegnet ihr viele Schwierigkeiten, wie bspw. "Warum wurde dies in der Vorlesung nicht erklärt?"
- Heute geht es darum, wie man mit den Übungsblättern im Modul Programmieren umgeht

Artemis

Einführung
○○

VPN
○○○

Wie verwenden wir Artemis
○○○○○○○

Git
○○○○○○○○

Warum Artemis?

- **Artemis** ist die Plattform, die wir für den Übungsbetrieb und auch für die Abschlusssaufgaben verwenden
- In diesem Kurs findet ihr eine Woche nach Herausgabe des Übungsblatts jeweils Aufgaben passend zu denen auf dem Übungsblatt
- Ihr könnt dann dort die Aufgaben lösen oder eure bereits erstellten Lösungen einreichen
- In Artemis bekommt ihr zu euren eingereichten Lösungen automatisiert erstes Feedback

Wie kann man auf Artemis zugreifen?

- Zugriff auf Artemis ist nur innerhalb des KIT-Netzes möglich
 - An der Uni
 - Auch in vielen Wohnheimen
- Sonst: Fernzugriff mittels VPN

Einrichtung des VPNs

Schritt 1 - VPN für Fernzugriff installieren

- Zugriff auf **Artemis** ist nur innerhalb des KIT-Netzes möglich, wie auch für weitere Dienste des SCC → Einrichtung des VPNs ist empfehlenswert

Dafür ist diese Kurzanleitung:

- Die Software OpenVPN wie unter **Anleitung** beschrieben herunterladen und installieren
- Eine Konfigurationsdatei herunterladen und in OpenVPN importieren
- Zum Verbinden mit u-Kürzel und Shibboleth-Passwort authentifizieren

Achtung!

- Die Installationsanleitung ist abhängig vom Betriebssystem - lest vorsichtig!

Einrichtung des VPNs

Schritt 2 - Passende VPN-Konfiguration auswählen

Die Konfigurationsdateien findet ihr genauso wie die Anleitung fürs OpenVPN [hier](#)
Das Standard kit-VPN:

- schleust jegliche Internetzugriffe durchs KIT-Netz

Das Split kit-VPN (die Konfigurationsdateien verstecken sich [hier](#))

- Schickt alle Anfragen, die nicht auf KIT-Domänen zugreifen, über normale Internetverbindung
- Empfehlenswert, da das Uni-Netz weniger belastet wird und Zugriffe auf Nicht-Uni-Dienste schnell bleiben

Achtung

- Für Split VPN empfiehlt es sich, das Split IPv4-Connect zu wählen, da das Split IPv6-Connect u.U. nicht fehlerfrei funktioniert!

Einrichtung des VPNs

Schritt 3 - Passende VPN-Konfiguration einrichten

- Lade die gewählte Konfigurationsdatei herunter und speichere sie.
Importiere die Datei in OpenVPN
- Nun sollte die Verbindung in OpenVPN unter "Verbinden " auswählbar sein
- Unabhängig vo der gewählten Konfigurationsdatei müsst ihr euch mit eurem u-Kürzel und dem Shibboleth-Passwort authentifizieren

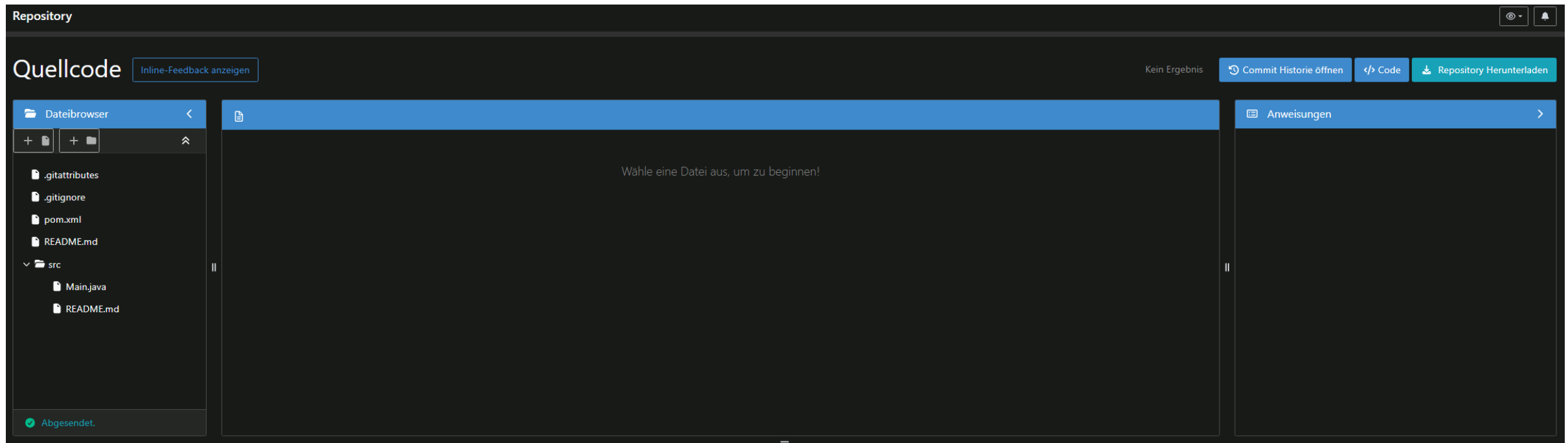
Wie verwenden wir Artemis?

Schritt 1 - Anmeldung in Artemis

- Erstmalige Registrierung:
 - Login ist das u-Kürzel, Passwort ist das Shibboleth-Passwort
- In dem Bereich "Kurseinschreibung" den Kurs **Programmieren (WS25/26)** auswählen
- Man kann sich auch weiterhin mit dem Shibboleth-Passwort anmelden
 - Dazu: KIT Login auswählen
 - Auch möglich eigenes Passwort zu setzen via Einstellungen, damit ihr Zugriff auf den zu Artemis gehörenden GitLab-Account zugreifen könnt

Schritt 2 Aufgaben ansehen und bearbeiten

- In Artemis seht ihr eine Woche nach Herausgabe des Übungsblatts die entsprechenden Aufgaben
- Zu jeder Aufgabe ist ein Verzeichnis (Repository) angelegt:



Einführung



VPN



Wie verwenden wir Artemis



Git



Schritt 2 Aufgaben ansehen und bearbeiten

Eine gewöhnliche Programmier-Aufgabe auf Artemis sieht so aus:

Blatt 1 Aufgabe C

Punkte
0 / 8

Abgabe bis:
20. Nov. 2025 06:00

Startdatum:
in einem Tag

Status
Nicht angefangen

Teilaufgaben:

Push your code to receive automatic feedback for your submission. Click on the results to see more information about test failures. Read those information carefully to fix your code.

1. **Mandatory Precondition** Keine Ergebnisse
This stage ensures that your code complies to certain criterias enlisted at the beginning of the exercise sheet.

2. **Mandatory Minimum Requirement** Keine Ergebnisse
This stage checks whether your code passes the required Checkstyle, provided in Ilias, and other requirements given in the exercise sheet, e. g. an example interaction.

3. **Mandatory Tests** Keine Ergebnisse
This is a container stage, you can ignore it.

4. **Functional** Keine Ergebnisse
This stage contains test cases for the described functionality of this exercise.

5. **Optional - Not Graded** Keine Ergebnisse
This stage heuristically checks for style aspects. Passing doesn't guarantee no deductions, nor does failing ensure deductions - it's simply a hint. They get assessed manually after submission due.

All mandatory test cases must pass to be eligible to gain points. Use our [Wiki](#) for more information about failure messages and reference to avoid mistakes. Also note the [inspections](#) article there.

After due date your **last** submission (no exception!) will get assessed. Artemis may be showing an **incorrect** result (e. g. 0% or "No Submission"/"Nicht abgegeben") **while the assessment is taking place**. This is a confirmed visual bug and does **not** represent your actual score or a missing submission. Check out the assessment due date below to verify whether the result shown is reliable or not.

If you've received an unjustified deduction in the manual assessment, a complaint can be filed for one week after the assessments have been released. You're obligated to have read the blue info feedback and related Wiki articles beforehand and base your complaint clearly on those to state why the deduction is unjustified.

Aufgabendetails

Veröffentlichungsdatum	5. Nov. 2025 12:00
Startdatum:	12. Nov. 2025 12:00
Abgabe bis:	20. Nov. 2025 06:00
Veröffentlichungsdatum der Beispiellösung	20. Nov. 2025 16:00
Bewertung bis	27. Nov. 2025 18:00
Beschwerde möglich	Ja

Aufgabe starten

Kommunikation

Suche nach Nachrichten

Ungeklärt Eigene Reagiert Datum: ↓

Keine Nachrichten gefunden

+ Neue Nachricht

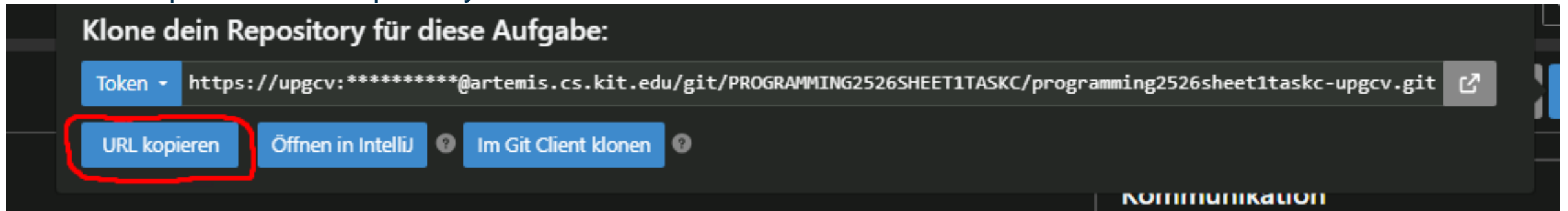
Schritt 2 Aufgaben ansehen und bearbeiten

- Klickt auf **Aufgabe starten**
- Ihr habt nun 3 Möglichkeiten, die Aufgabe zu lösen:
 1. in eurer IDE
 2. mit Git lokal
 3. mit GitLab online (theoretisch)

In eurer IDE

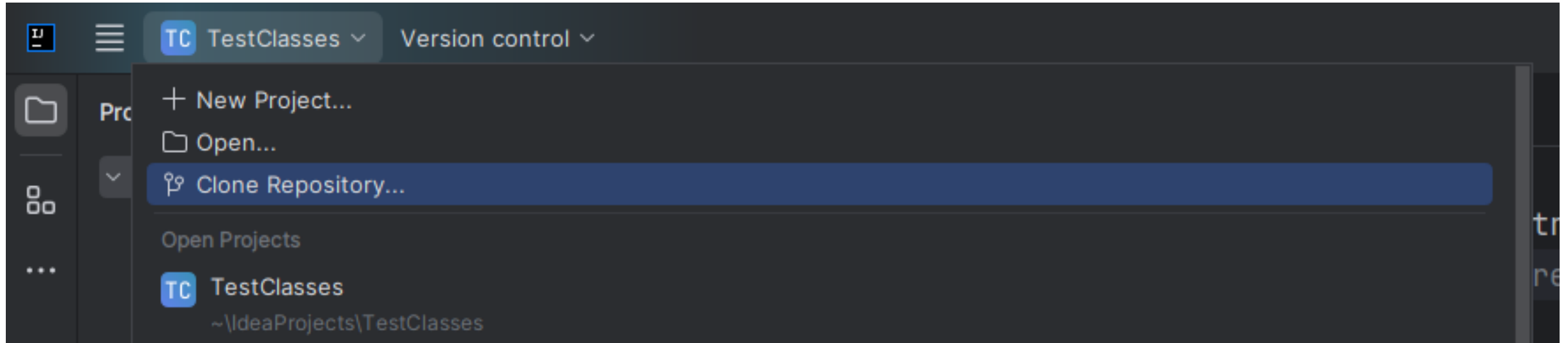
Am Beispiel von IntelliJ IDEA:

Allererste kopiert ihr den Repository-Link von Artemis



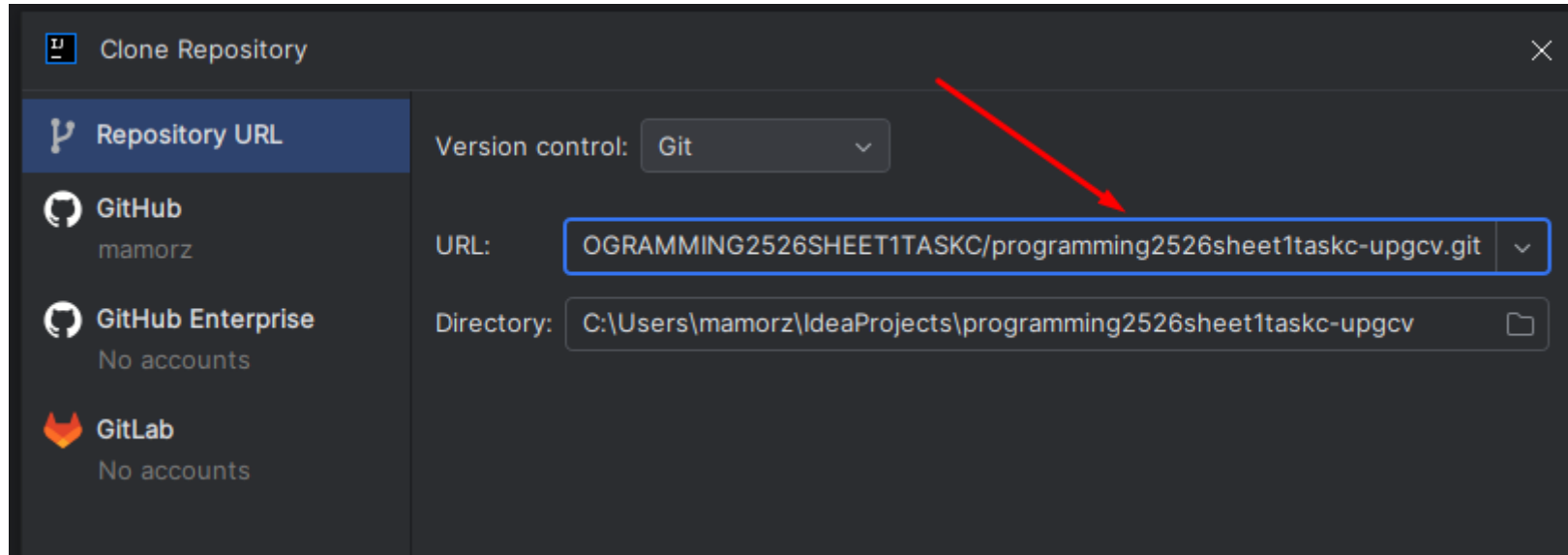
In eurer IDE

In IntelliJ links oben findet ihr im ausklappenden Fenster die **Clone Repository**-Taste



In eurer IDE

Dann könnt ihr den Repo in diesem Fenster clonen:



Git

Einführung
○○

VPN
○○○

Wie verwenden wir Artemis
○○○○○○○

Git
●○○○○○○○

Versionskontrollsystem - Motivation

- Warum kompiliert/baut/whatever das Projekt schon wieder nicht???
 - Wo liegt das Problem?
 - Was wurde denn seit dem letzten Mal geändert?
- Der Texteditor hat (wenn überhaupt) irgendeinen Zustand in der Vergangenheit gesichert
 - Immerhin besser als die sonst vorhandene Leere
 - Explizite Versionen bei Git

Git Crashkurs

Installation und optionale GUI

Git für die Konsole

- Linux: Über den jeweiligen Paket Manager Git installieren
- Anleitungen für verschiedenste Linux-Distributionen finden sich [hier](#)
- Windows: Beispielsweise [hier](#) herunterladen
- MacOS: Verschiedene Möglichkeiten, um Git auf MacOS zu installieren, sind [hier](#) zu finden

Wer lieber mit GUIs als mit der Kommandozeile arbeitet, findet [hier](#) eine große Auswahl für verschiedene Plattformen (Ich würde in IntelliJ empfehlen zu arbeiten)

Git Crashkurs

Git lokal konfigurieren

- Zunächst muss die eigene Identität (Name und Mail-Adresse) in Git über die Kommandozeile hinterlegt werden:
 - `git config --global user.name „Dein Name“`
 - `git config --global user.email deine.mail@irgendwas.com`
- Die Email sollte die KIT Email sein
- Ausführlich erklärt wird es [hier](#)

Git typischer Ablauf

```
$ cd "Pfad"
$ git clone username@host:/path/to/repository
$ git pull
... Aufgabe bearbeiten ...
$ git add . oder $ git add -all oder $ git add -A
$ git commit -m "Commit message "
... Die Nachricht ist notwendig und darf sinnvoll sein ...
$ git push
```

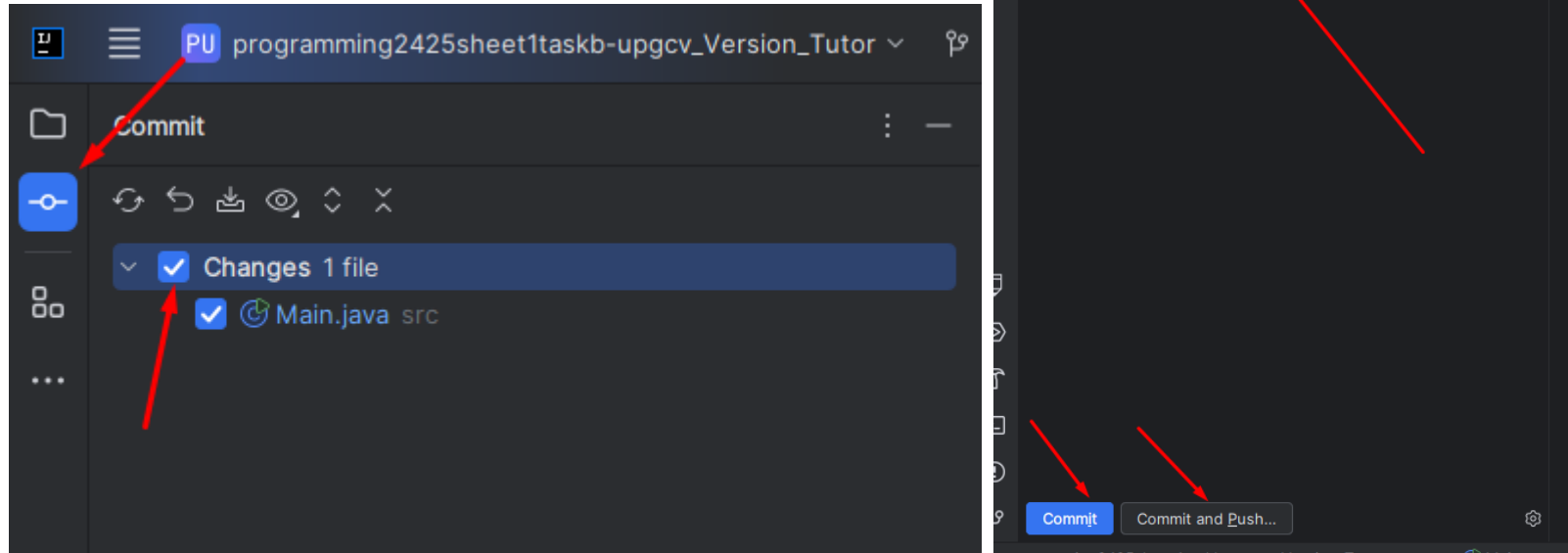
Git Crashkurs

Typischer Ablauf - Cheat Sheet

- Navigiere in der Konsole in das Verzeichnis, in dem du arbeiten möchtest, zB mit `$ cd "Pfad`
- Oder im Explorer im entsprechenden Verzeichnis Rechtsklick:
„Git Bash Here“, Terminal hier öffnen“
- Lade das Verzeichnis herunter mit `$ git clone <link>`
- Jede Aufgabe hat ein eigenes Verzeichnis, den Link findet man unter *Repository klonen* in Artemis
 - Option 1: mit **HTTPS** und Authentifizierung mit Artemis-Benutzername und Artemis-Passwort
 - Option 2: mit **SSH** (dazu nach Wunsch nächstes Mal)
- Bearbeite die Aufgabe im heruntergeladenen Verzeichnis
- Füge neue Dateien oder Dateiänderungen zu Git hinzu (spätestens vor dem Hochladen):
 - `$ git add <datei>` für einzelne Dateien
 - `$ git add -all` oder `$ git add .` für das gesamte Verzeichnis
- Speichere die Änderungen durch Einbuchten in Git: `$ git commit -m "Nachricht"`
- Änderungen dem Online-Verzeichnis hinzufügen mit `$ git push`

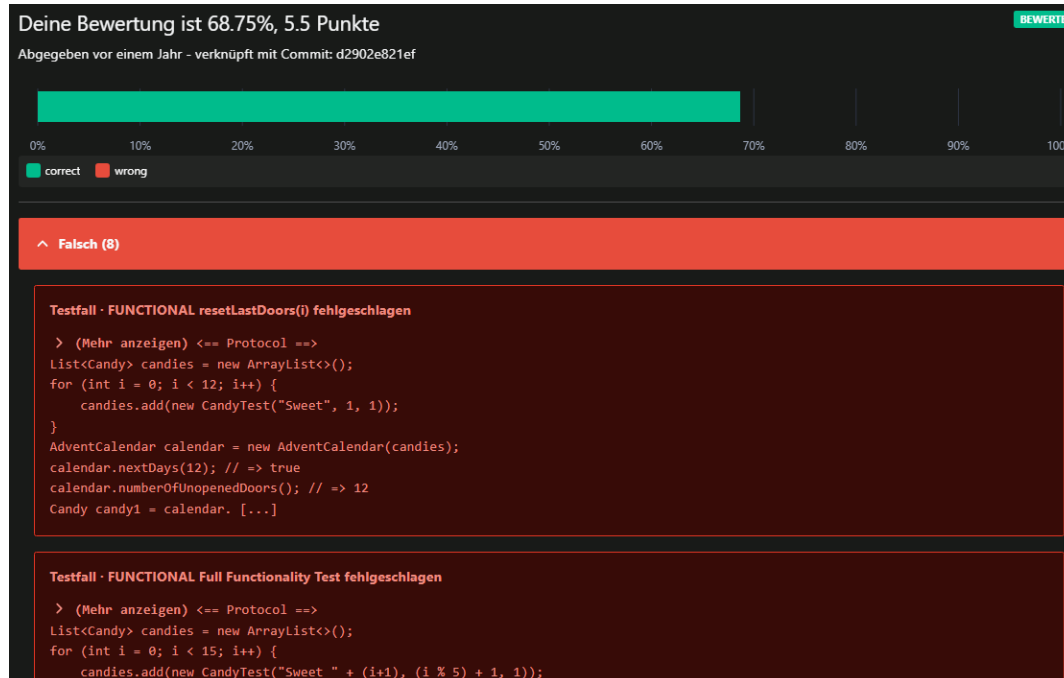
Git in IntelliJ

Nachdem ihr in IntelliJ die Aufgabe gelöst habt, könnt ihr die jetzt über die IDE commiten und pushen



Als letztes

- Warten und hoffen, dass die Abgabe alle Tests erfolgreich besteht
- Wenn nicht, wieder die Aufgabe bearbeiten mit Rücksicht auf die Fehlermeldungen



Einführung
○○

VPN
○○○

Wie verwenden wir Artemis
○○○○○○○

Git
○○○○○○○●