

Organisation der Repositories

Durch den vermehrten Einsatz von GitHub und Classrooms müssen wir unsere Repositories organisieren.

Organisation	Einsatzzweck
<i>Für Lehrpersonen</i>	
bzz-fgict	Aktuell für alles, etwas chaotisch
teacher-python	Aufgaben, Leistungsbeurteilungen, Lösungen für M319, M320, ???
teacher-java	Aufgaben, Leistungsbeurteilungen, Lösungen für M403, M404, M120, M320, M322, ???
Modulbereich „Programmierung - Webapplikationen“	Aufgaben, Leistungsbeurteilungen, Lösungen für M133/M151
BZZ-Commons	Skripts, Workflows, ... die wir weltweit freigeben
<i>Für Vorlagen (Templates)</i>	
templates-python	Vorlagen für Assignments im Classroom
templates-java	
<i>Für Lernende</i>	
BZZ-2022	Jahrgang 2022 (IA22a, IA22b, IM22a, IM22b, ...)
BZZ-2023	Jahrgang 2023 (IA23a, IA23b, IM23a, IM23b, ...)

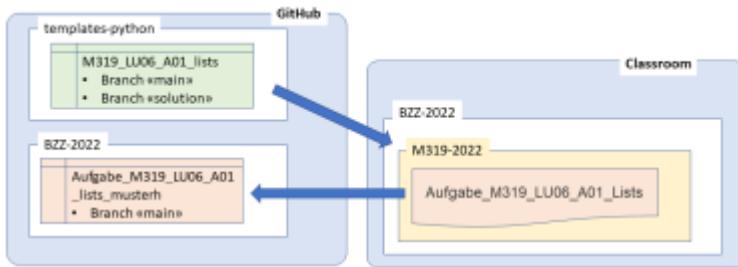


Ich beschreibe den Ablauf jeweils am Beispiel einer neuen Programmieraufgabe im Modul 319. Der Titel der Aufgabe lautet: „LU06.A01: Listen“. Ich möchte diese Aufgabe bei den Klassen IM22a und IM22b einsetzen.

Ablauf ohne geheime Lösung



Bei dieser Variante wird nur ein Repository mit Branches für die Vorlage (main) und den Lösungsvorschlag (solution) genutzt.



1. Neues Repository erstellen

Ich erstelle ein neues Repository für diese Aufgabe (siehe [Vorlage für Pycharm mit GitHub Classroom](#)). Dieses Repository ist **public** definiert. Die Lernenden müssen Zugriff auf dieses Repository haben.

- Organisation: templates-python
- Repository: M319_LU06_A01_Lists
- Branch: main

Ich kennzeichne dieses Repository als „Template“.

2. Vorlage erstellen

Die Vorlage enthält alle Dateien und Funktionen, welche die Lernenden als Startpunkt erhalten:

- Branch: **main**

lists.py	Dieses Skript sollen die Lernenden erstellen/ergänzen
lists_test.py	Unit-Tests für die Lernenden
.gitignore	
requirements.txt	Libraries die importiert werden müssen
.github	Ordner für die Workflows
classroom/autograding.json	Testfälle mit Bewertung
workflows/classroom.yml	Workflow zum Durchführen der Testfälle

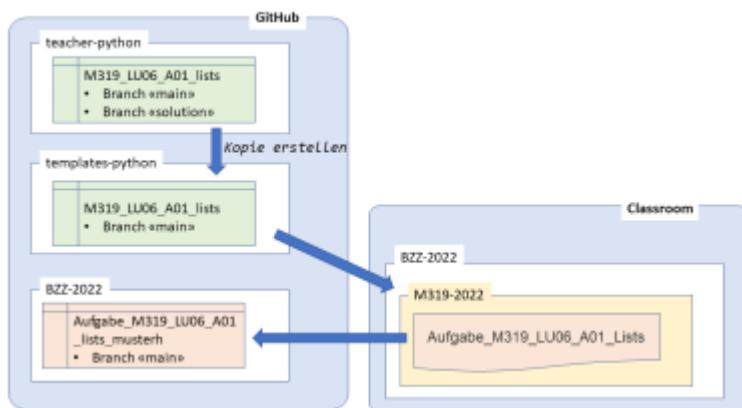
3. Musterlösung erstellen

Die Musterlösung kann direkt im gleichen Repository als eigener Branch z.B. **solution** erstellt werden.



Mein Repository ist nun vollständig.

Ablauf mit geheimer Lösung



Bei dieser Variante haben die Lernenden keinen Zugriff auf den Branch „solution“. Dieses Vorgehen ist vor allem bei Leistungsbeurteilungen sinnvoll.

1. Neues Repository erstellen

Ich erstelle ein neues Repository für diese Aufgabe. Dieses Repository ist **internal** (oder private) definiert. Die Lernenden sollen keinen Zugriff auf dieses Repository erhalten.

- Organisation: teacher-python
- Repository: M319_LU06_A01_Lists
- Branch: main

2. Vorlage erstellen

Die Vorlage enthält alle Dateien und Funktionen, welche die Lernenden als Startpunkt erhalten:

- Branch: **main**

lists.py	Dieses Skript sollen die Lernenden erstellen/ergänzen
lists_test.py	Unit-Tests für die Lernenden
.gitignore	
requirements.txt	Libraries die importiert werden müssen
.github	Ordner für die Workflows
classroom/autograding.json	Testfälle mit Bewertung
workflows/classroom.yml	Workflow zum Durchführen der Testfälle

3. Musterlösung erstellen

Die Musterlösung kann direkt im gleichen Repository als eigener Branch z.B. **solution** erstellt werden.



Mein Repository ist nun vollständig.

4. Kopie erstellen

Ich erstelle eine Kopie meines Repositories, wobei ich nur den **main**-Branch hochlade. Damit wird meine Lösung nicht übernommen wird.

- Name: „templates-python/M319_LU06_A01_Lists“
- Visibility: „public“

Dieses Repository kennzeichne ich als „Template“.

Repository im Classroom einsetzen



Um ein Repository in einem Classroom zu verwenden, muss das Repository **public** sein oder in der gleichen Organisation wie der Classroom liegen.

- Das Template für die Assignments wird in der Organisation „templates-python“ angelegt. |
- Für die Classrooms der Klassen „IM22a“ und „IM22b“ verwende ich die Organisation „BZZ-2022“.

1. Classroom erstellen

Falls noch kein Classroom für meine Klassen „IM22a“, „IM22b“ und das Modul „319“ existiert, erstelle ich einen neuen Classroom in der Organisation „BZZ-2022“.

- Classroom-Name: „M319-2022“

2. Assignment erstellen

Ich erstelle ein Assignment im Classroom „M319-2022“ für die Aufgabe.

- Assignment: „Aufgabe_M319_LU06_A01_Lists“
- Template: „templates-python/M319_LU06_A01_Lists“

Den Link zu diesem Assignment stelle ich den Lernenden zur Verfügung.

Anpassungen / Fehlerkorrekturen in der Vorlage

Anpassungen am Template werden in künftigen Repositories übernommen. Es gibt keinen Automatismus um bereits erstellte Repositories der Lernenden zu aktualisieren.

Housekeeping



Aufräumen von Organisationen / Repos / Benutzerkonten

Abschlussklassen

Am Ende der Lehre

- Lernende auffordern Ihre Repos zu sichern
 - Repository in anderes Konto transferieren
 - UserId, Emailadresse ändern und Organisation verlassen

Nach Ablauf der Rekursfristen

- Organisation löschen
- Classroom löschen

From:

<https://wiki.bzz.ch/> - BZZ - Modulwiki



Permanent link:

<https://wiki.bzz.ch/howto/git/organisation>

Last update: **2024/03/28 14:07**