

LU05.A12: Split in pieces



Erstelle ein Programm Schritt für Schritt mit Hilfe von GitHub Issues.

Auftrag

Diese Aufgabe besteht aus einem grösseren Programm, dessen Funktionalität in kleinen Schritten implementiert wird. Dieses Vorgehen wird beim Programmieren immer dringend empfohlen. Die einzelnen Ausbauschritte sind als Issues im GitHub-Repository eingetragen.

Issues

Bei dieser Aufgabe sind die einzelnen Arbeitschritte als Issues in deinem GitHub Repository festgehalten. Mit Issues in unserem GitHub Repository können wir offene Punkte und Fehler dort verwalten, wo auch der Sourcecode ist.

The screenshot shows a GitHub repository page for 'm319-ix24 / m319-lu05-a12-pieces-graphics80'. The navigation bar at the top has tabs for Code, Issues (which is highlighted with a red box), Pull requests, Actions, Projects, Wiki, and Security. Below the navigation bar, the repository name is displayed along with a 'Private' badge. It says 'forked from m319-ix24/m319-ix24-m319-lu05-a12-pieces-m319-lu05-a12-pieces'. There are buttons for 'main' (with a dropdown arrow), '1 Branch', '0 Tags', and a search icon. A message indicates the branch is '4 commits ahead of, 1 commit behind m319-ix24/m319-ix24'. Below this are buttons for 'Contribute' and 'Sync fork'. At the bottom, there is a list of recent commits, with one entry by 'graphics80' that says 'Update main.py' and has a green checkmark. Navigation links for 'github' and 'Find file' are also visible.

Vorbereitung

1. Akzeptiere das Assignment wie gewohnt
2. Um die Issues zu erhalten, trage deinen Namen (*es ist egal ob du deinen Vorname, Nachname, GitHub-Username oder ein Fantasynamen einträgst*) ins main.py ein und commit und push

```
1  def main():
2      # Write your program here and remove the line "pass"
3      # TODO: To receive the Issues write your Name here {} and Commit and Push
4      pass|
5
6
7  if __name__ == '__main__':
8      main()
```

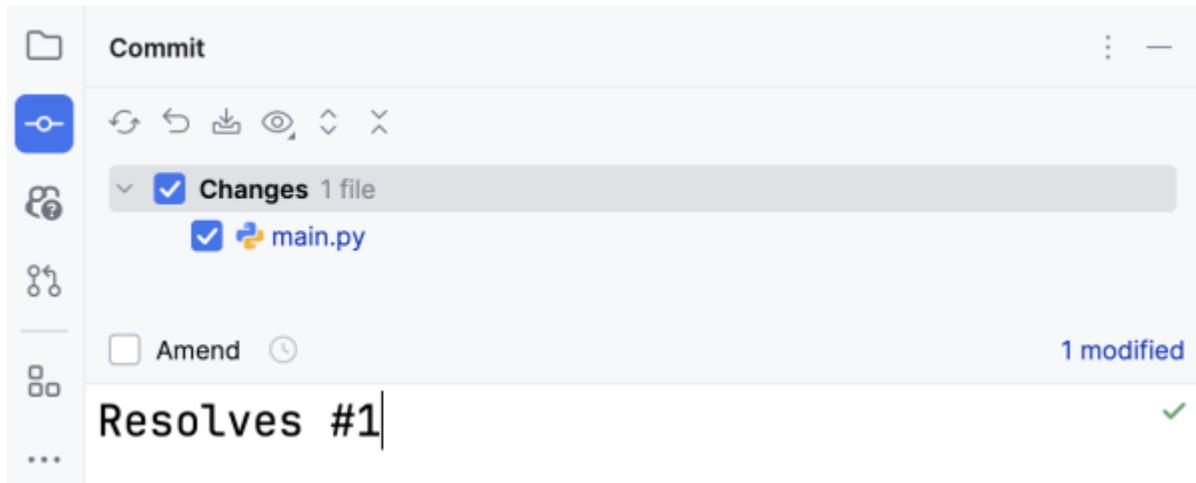
3. Überprüfe dein Repo auf GitHub ob die Issues vorhanden sind (dauert etwa 2 Minuten, Seite aktualisieren nicht vergessen)

Issues (Aufträge) bearbeiten

Nun kannst du die einzelnen Issues abarbeiten.

1. Wähle einen Teil der Aufgabe aus, den du eigenständig umsetzen kannst.
2. Codiere den Sourcecode für diese Teilaufgabe.
3. Führe die relevanten Testfälle durch.
4. Sind alle Tests erfolgreich:
 - Führe einen Commit und einen Push durch.
5. Sonst
 - Identifiziere und korrigiere die Fehler.
 - Zurück zu Schritt 3.
6. Zurück zu Schritt 1

Du kannst erledigte Issues automatisch schliessen, wenn du in der Commit Message `resolves #n` angibst (wobei n die Nummer des Issues auf GitHub ist).



Dadurch wird der Commit mit den relevanten Issues verknüpft und geschlossen, was die Übersicht noch weiter erhöht.

The screenshot shows a GitHub repository page for 'm319-ix24 / m319-lu05-a12-pieces-graphics80'. The 'Issues' tab is selected, showing 4 open issues and 1 closed issue. The closed issue, '#1', is titled 'Iteration für Eingabe' and was closed by a GitHub Actions bot 23 minutes ago.

Natürlich werden die Issues erst aktualisiert, wenn du einen Push durchführst.

Abgabe

Die Abgabe erfolgt durch den Push ins GitHub Repository. In Moodle ist keine Abgabe vorgesehen oder möglich.

Anmerkung: Kümmere dich im Moment nicht zu sehr um `if __name__ == '__main__':`. Dieser Programmablock legt fest, welche Funktion beim Ausführen gestartet wird. Wir werden dies im Verlauf des Moduls noch genauer betrachten.

M319-LU05



Kevin Maurizi, Marcel Suter

Diese Aufgabe ist eine übersetzte und angepasste Aufgabe von [Scott Morgan](#), verwendet unter CC BY NC SA.

From:

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

Permanent link:

<https://wiki.bzz.ch/de/modul/m319/learningunits/lu05/aufgaben/splitinpieces?rev=1758614056>

Last update: 2025/09/23 09:54

