

# LU15.A01 - Lotterie



Erstelle Testfälle mit Pytest um die Lotterie-Applikation zu testen. Prüfe die Codeabdeckung (Coverage) um die Applikation möglichst vollständig zu testen.

## Auftrag

Für die Applikation „Lottery“ müssen die Unittests erstellt werden. Das Ziel ist eine möglichst hohe Codeabdeckung. Dazu müssen sowohl korrekte Werte als auch Fehlersituationen getestet werden.

## Coverage

Geben Sie im Terminal von Pycharm diese Befehle aus:

<code>coverage run</code>	Führt alle PyTest-Funktionen aus.
<code>coverage html</code>	Erstellt einen Bericht im Ordner „htmlcov“

Im Ordner „htmlcov“ finden Sie eine Datei „index.html“. Diese können Sie in einem Webbrowser öffnen.

## Test-Dateien

- Erstellen Sie sinnvolle Dateien für Ihre Tests. Nicht zu jedem Modul muss eine eigene Testdatei erstellt werden.
- In `test_main.py` sind schon drei Unit-Tests vorgegeben, welche die Funktion `main()` abdecken.
  - Diese Tests nutzen simulierte Funktionen (Mocks). Für die Integrationstests brauchen Sie passende Testfälle **ohne** Mocks.

## Automatisierte Tests

Um die Tests bei jedem Push auszuführen, passen Sie die Einstellungen für den automatisierten Workflow an. Die beiden Dateien finden Sie im Ordner `.github/autograding`.

## lint.json

Diese Datei enthält eine Aufzählung aller Dateien, die mittels `pylint` geprüft werden. Ergänzen Sie die Liste mit Ihren neuen Test-Dateien.

## unittests.json

Diese Datei enthält eine Aufzählung aller Tests, die auszuführen sind. Ergänzen Sie die Aufzählung um Ihre neuen Testfälle.

---

### M450-LU15



Marcel Suter

From:

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

Permanent link:

<https://wiki.bzz.ch/modul/m450/learningunits/lu15/aufgaben/lottery?rev=1732527146>

Last update: **2024/11/25 10:32**

