Module 1 (Python Crash Course) Final Project

Welcome and congratulations on reaching the final project of Module 1. This project is graded, and with your newly acquired skills and our guidance, we are confident you'll excel. Best of luck!

If you have any questions, please post a message on Teams. You will have four individual consultations, two with each lecturer.

Project Requirements

General Requirements

- Adhere to the PEP 8 style guide. Use an auto-formatter for consistency.
- Aim for ~80% code coverage with well-designed test cases.
- The project must be pip-installable and available on PyPI.
- The project must be publicly available on GitHub with a clean Git history.
- Project structure should include: module directory, tests directory, README.md (with user guide or documentation), .gitignore, requirements.txt/pyproject.toml (specifying dependencies and Python version compatibility - ideally 3.10+).
- Organize the module directory efficiently. Avoid large files or unusual naming conventions.
- Maintain clean code. Exposed functions must have docstrings, clear naming, and type annotations.

Specific Requirements

- Implement web crawling functionality returning well-structured data of multiple types (e.g., text, numeric, categorical, images, audio), with at least two data types.
- Access all functionality through a single function in the main module.
- Function capabilities:
  - Time limit for execution (e.g., 60 seconds).
  - Option to select the source for crawling (e.g., lrytas.lt, eurovaistine.lt).
  - Different data presentation formats: CSV string and a dictionary of records. For non-CSV data (images, audio), use dict records.
  - Include sample crawled data.

Example Code Snippets

1. **Calling main function**:

```
crawl(time_limit=60, source='lrytas', return_format='csv')
```

2. **Example of checking code coverage with pytest**

```
pytest --cov=my_module tests/
```