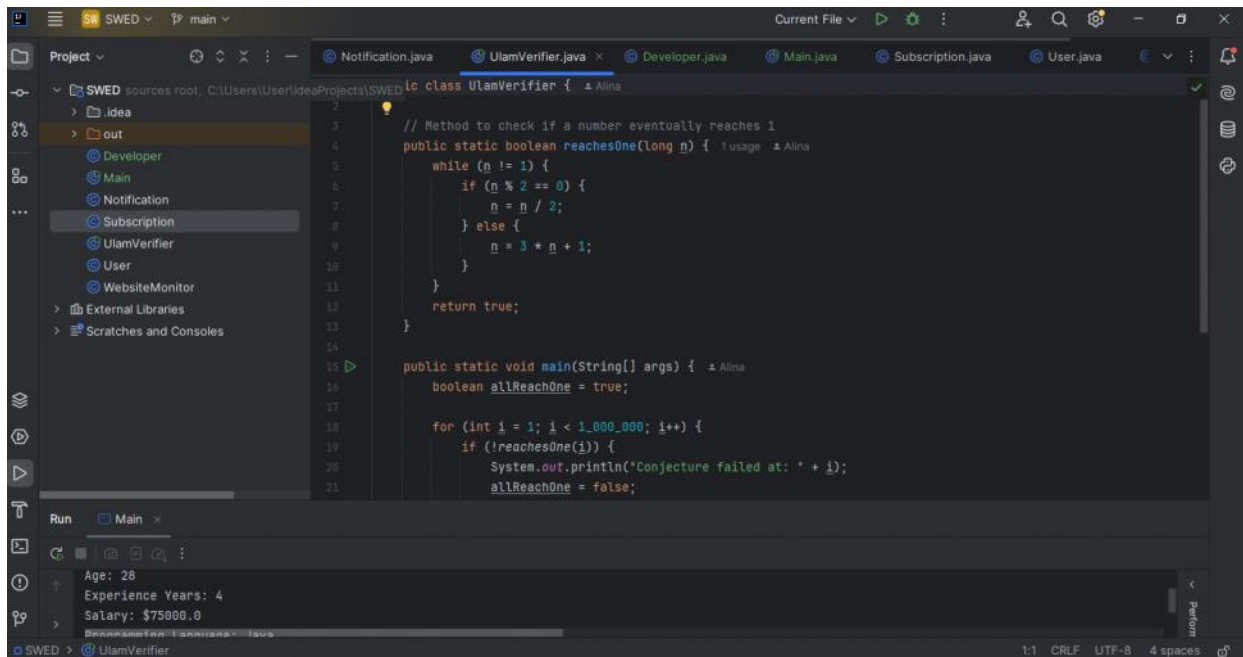# Exercise 1

Montag, 28. April 2025   08:29

1)

1) Automatic code completion - time saving, reduces typing mistakes
2) Syntax highlighting - code is easier to read and understand
3) Debugging tools - you can set breakpoints and step through code
4) Error detection - errors and warnings are often underlined immediatley
5) Integrated terminal/console - run, test and debug code in the DIE
6) Refactoring tools - automatically rename variables, extract methods, etc.
7) Version control integration (Git,...) - commit, push and pull without leaving the DIE
8) Project and file management - clear handling of project structure
9) Plugin system - extensible with many plugins
10) Code navigation - easy access to definitions, references and implementations

2)
- You cannot prove the Ulam function terminates for all positive integers mathematically (unsolved)
  - But you can verify it by computer for all numbers under 1 million (Java Code implemented)

3)
- Created GitHub repository

4)

MagicDraw installation

5)
- UML diagrams for 'Developer' and Java implementation

6)

<mark>Software design:</mark>
1) The process of defining the architecture, components, interfaces and other characteristics of a system or component.
2) The result of the process in 1)

- Focus: how the system will be built
- Goal: blueprint for developers to follow when coding
→ How exactly will we build the software?

<mark>Software analysis:</mark>
- Process of understanding and documenting what the software needs to do
- Focus: what problems need to be solved, what functionalities are needed
- Goal: clearly define problem space before any coding/design begins
→ What should the software do?

7)
- Provides a clear plan for building a system
→ Without it: projects can become disorganized, harder to maintain, more expensive to fix later
- Small, personal projects might skip this step

8)
- The design activities are interdependent
→ Changes in one area can affect the others

Example: database design changes (new field is added to save user prefernece)
- UI design must be adapted to allow users to enter their preference
- Component design must handle saving and retrieving this data
- Architectural design might need to adjust how different parts of the system communicate
→ All parts of the design must work together to build a system that functions properly