Software Engineering – Design Exercise 1

1. Please name 10 advantages of an IDE compared to a simple text editor.

1. Code Autocompletion

- Suggests code as you type (functions, variables, classes)
- Reduces typos and speeds up coding

2. Syntax Highlighting

- Color-coding for keywords, variables, errors
- Makes code easier to read

3. Integrated Debugger

- Step through code line-by-line, set breakpoints, inspect variables
- Much more efficient than manually inserting print()s

4. Error Detection While Typing

- Real-time warnings and error highlighting
- Prevents basic mistakes early before running program

5. Run Automation

- One-click build and run features
- No need to manually compile from the terminal

6. Project and File Management

- Organizes files into projects and folders automatically
- Easier navigation between large amounts of files

7. Version Control Integration

- Built-in Git/SVN support (commit, push, pull inside the IDE)
- No need to switch to the terminal for version control

8. Refactoring Tools

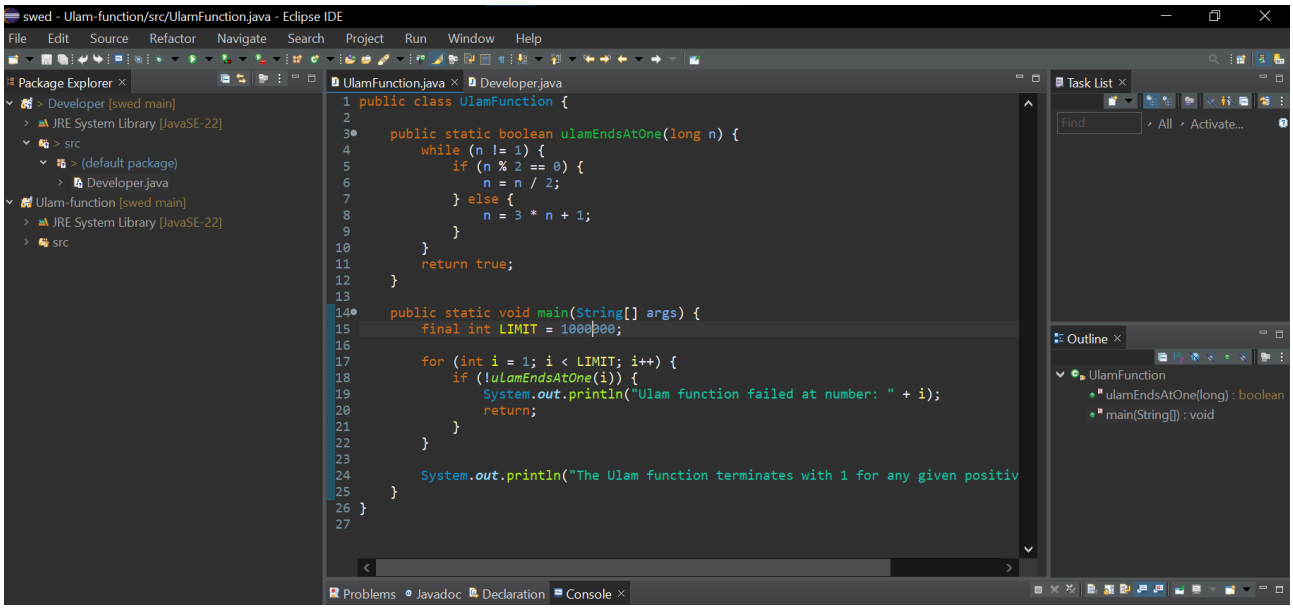
- Easily rename variables, extract methods, reorganize code
- Reduces the risk of breaking code when changing things

9. Code Templates and Snippets

- Predefined code structures (e.g., for loops, class templates)
- Speeds up repetitive tasks

10. Plugin and Extension Support

- Customize your environment with plugins (e.g., for frameworks, languages, testing tools).
- Tailor the IDE exactly to your needs



6. Define the term software design. Explain how this differs from software analysis.

- **Software Design** is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. It is about "how" the system will be built - creating a blueprint or plan for implementation.
- **Software Analysis**, in contrast, is about understanding and specifying *what* the system must do. It involves gathering requirements, analyzing needs, and modeling the problem without yet deciding on the technical details of the solution.

So:

- **Analysis** = Understanding the problem ("what to build").
- **Design** = Planning the solution ("how to build it").

7. Explain why a software design is necessary for a software project. Can you think of a project without this step? What could be the consequences?

Why software design is necessary:

- It provides a clear, structured plan for developers to follow.
- It reduces complexity by breaking the system into manageable parts.
- It ensures that all components fit together well (e.g., UI, database, APIs).
- It helps identify potential problems early, saving cost and time later.
- It makes maintenance and future updates easier.

Consequences of a project without software design:

- Inconsistency: Different parts of the system may not work together.
- Rework: Frequent changes and fixing issues would be needed.
- Scalability problems: System might not handle growth properly.
- Increased costs: Fixing issues during development or after deployment is much more expensive.
- Poor quality: Leads to buggy, hard-to-maintain software.

8. Are the design activities of architectural design, database design, user interface design, and component design independent or interdependent? Using an example, explain why.

They are interdependent, not independent.

- Decisions made during architectural design affect database and UI design.
- Changes in the database (like table structures) may require UI updates.
- Component design depends on how architecture divides the system into modules.

For example designing a social media app:

- Architectural design decides you will have a client-server model with mobile apps and a backend API.
- Database design follows by setting up tables like Users, Posts, and Comments.
- UI design needs to show user profiles, posts, and comments clearly, so it must match what data is stored and accessible.
- Component design defines the API endpoints like `GET /posts`, which both the app and database must support.

If one part changes (say, you add a "Story" feature to the database), you would need to update both UI and components accordingly.