Perfect Guess – Advanced Number Guessing Game

Problem Statement

Design and implement an interactive number guessing game in Python using Object-Oriented Programming principles. The application should generate a random number within a user-defined range and allow the user to guess the number through repeated attempts while providing meaningful feedback and tracking performance.

Functional Requirements

- Generate a random number based on the selected difficulty level.
- Allow the user to choose from multiple difficulty levels.
- Prompt the user repeatedly until the correct number is guessed.
- Indicate whether each guess is too high, too low, or correct.
- Track and display the total number of attempts.
- Store and update the best (minimum) attempts as a high score using file handling.
- Validate user input to prevent runtime errors.
- Allow the user to replay or exit after each game session.

Technical Constraints

- Must be implemented using Object-Oriented Programming.
- Use classes to encapsulate game logic and score handling.
- Use file handling for persistent high score storage.
- Follow clean coding practices and modular design.
- Must run as a command-line application.

Learning Outcomes

- Understanding of random number generation.
- Application of loops and conditional logic.
- Proper use of classes and methods.
- Basic file handling and persistence.
- Designing a complete interactive application.