# Sample Refactoring Documentation for Project “[Bulls-and-Cows-4](https://github.com/Helium-Team/Bulls-and-Cows-4)”

Team “Helium”

1. Redesigned the project structure:
   * Renamed the project to [**BullsAndCows**](https://github.com/Helium-Team/Bulls-and-Cows-4/blob/master/BullsAndCows.sln).
   * Renamed the main class **Program** to **BullsAndCows**.
   * Extracted each class in a separate file with a good name: **CommandReader, ConsolePrinter, GameEngine, NumberGenerator, PlayerCommand, PlayerHelper, ScoreBoard**;
   * Remove existing classes: **Player** and **Score**;
   * Add new classes: **CommandReader, ConsolePrinter, GameEngine, NumberGenerator, PlayerCommand, PlayerHelper, ScoreBoard;**
2. Reformatted the source code:
   * Removed all unneeded empty lines, e.g. in the class ConsolePrinter;
   * Remove variables: attempts and cheats from class GameEngine;
   * Inserted empty lines between the methods.
   * Split the lines containing several statements into several simple lines.
   * Formatted the curly braces **{** and **}** according to the best practices for the C# language.
   * Put **{** and **}** after all conditionals and loops (when missing).
   * Character casing: variables and fields made **camelCase**; types and methods made **PascalCase**.
   * Formatted all other elements of the source code according to the best practices introduced in the course “[High-Quality Programming Code](http://codecourse.telerik.com/)”.
   * Renamed all variables with appropriate English words code according to the best practices introduced in the course “[High-Quality Programming Code](http://codecourse.telerik.com/)”.
3. Introduced constants:
   * **SHOWED\_TOP\_SCORE ;**
   * **WELCOME\_MESSAGE ;**
   * **WRONG\_COMMAND\_MESSAGE;**
   * **MAX\_COW\_VARIATIONS;**
   * **NUMBER\_LENGTH;**
   * **NUMBER\_OF\_DIGITS;**
4. Introduced class **ScoreBoard** and moved all related functionality in it and make it Singleton.
5. Introduced class **PlayerCommand** and moved enum in it..
6. Introduce class **NumberGenerator** and moved all related functionality in it.
7. Introduce class **PlayerHelper** and moved all related functionality in it.
8. Introduce class **ConsolePrinter** and moved all related functionality in it.
9. Split method **CalculateBullsAndCowsCount (playerInput,  generatedNumber,  out bullsCount,  out cowsCount)** in three different methods **CallculateBullsCount(bool[] isBull), CallculateCowsCount(bool[] isBull), CheckPlayerInputForBull;**
10. Make class PlayerHelper and Number generator static.
11. Add method ClearHlep in class PlayerHelper.