# Refactoring Documentation for Project “Game 15”

Team “FRANCIUM”

1. Redesigned the project structure:
   * Renamed the project **igrata15** to **Game15**.
   * Renamed the main class **Program** to **MainMethod**.
   * Extracted classes containing properties, fields, constructors and methods out of static methods. The following classes have been created:
     1. **Board** (all related functionality moved in it)
     2. **ConsoleRenderer** (all related functionality moved in it)
     3. **Engine** (all related functionality moved in it)
     4. **Messages** (all related functionality moved in it)
     5. **Score** (all related functionality moved in it)
     6. **Turn** (all related functionality moved in it)
2. Reformatted the source code:
   * Removed all unneeded empty lines, e.g. in the class **DvoikaImeRezultat(block of 39 empty lines!)**.
   * Inserted empty lines when necessary following the quality code formatting conventions.
   * 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/)”.
3. Renamed variables:

|  |  |
| --- | --- |
| **Refactored data** | **Original data** |
| Board.cs:  - EmptyCellValue  - DirectionRow  - DirectionColumn  - Random  - matrixSizeRows  - matrixSizeColumns  - matrix  - emptyCellRow  - emptyCellColumn | Program.cs: |
| ConsoleRenderer.cs:  *HOLDS ONLY METHODS* | Program.cs: |
| Engine.cs:  - renderer (with property “Renderer”);  - board (with property “Board”);  - score (with property “Score”); | Program.cs:  *NO ANALOG IN THE ORIGINAL CODE*   *NEW VARIABLES CREATED* |
| MainMethod.cs:  *HOLDS THE MAIN METHOD ONLY* | Program.cs: |
| Messages.cs:  *HOLDS ONLY METHODS* | Programs.cs: |
| Score.cs: - name (with property “Name”); - points (with property “Points”); - topScoresMaxCapacity (with property “TopScoresMaxCapacity”); - fileForExternalSave (with property “FileForExternalSave”); - topScoresPersonPattern(with property “TopScoresPersonPattern”); | DvoikaImeRezultat.cs(1) + Program.cs(2):  - name(1);   - score(1);  - TopScoresAmount(2);   - TopScoresFileName(2);   - TopScoresPersonPattern(2); |
| Turn.cs:  - count (with propery “Count”); | Program.cs:  - ??? |

1. Extracted methods:

|  |  |
| --- | --- |
| **Refactored data** | **Original data** |
| Board.cs:  ValidateNextCell(int direction)  CellNumberToDirection(int cellNumber)  MoveCell(int direction)  CheckIfMatrixIsOrderedCorrectly()  InitializeMatrix()  ShuffleMatrix() | Program.cs: |
| ConsoleRenderer.cs:  RenderTopScores(Score scores)  RenderMessage(string message)  RenderMatrix(Board gameField)  GetHorizontalBorder(Board gameField) | Program.cs: |
| Engine.cs:  PlayGame()  LoadBoard()  NextMove(int cellNumber)  GameOver(Score score) | Program.cs: |
| MainMethod.cs:  Main() | Program.cs: |
| Messages.cs:  GetCellDoesNotExistMessage()  GetGoodbye()  GetIllegalCommandMessage()  GetIllegalMoveMessage()  GetNextMoveMessage()  GetWelcomeMessage()  GetCongratsMessage()  GetSorrowMessage()  GetMessageRequestToEnterName() | Programs.cs: |
| Score.cs:  GetTopScoresFromFile()  UpgradeTopScore()  UpgradeTopScoreInFile()  UpgradeTopScorePairs(string[] topScores) | DvoikaImeRezultat.cs(1) + Program.cs(2): |
| Turn.cs:  *DOES NOT HOLD METHODS* | Program.cs: |