# Refactoring Documentation for Project “Game 15”

1. Redesigned the project structure: Team “Game-Fifteen-2”

* Renamed the project to GameFifteen.
* Renamed the main class Program to GameFifteen.
* Extracted classes containing properties, fields, constructors and methods out of static methods.
* The following namespaces have been created:
  + - GameFifteen.Common
    - GameFifteen.UI
    - GameFifteen.Common.Tests
* The following folders have been created:
  + - Contracts
    - Engine
    - GameObjects
    - Utils
* The following classes have been created:
  + - **Command** (all related functionality moved in it)
    - **GameEngine** (all related functionality moved in it)
    - **EqualMatrixChecker** (all related functionality moved in it)
    - **MatrixEmptyCellRandomizator** (all related functionality moved in it)
    - **Player** (all related functionality moved in it)
    - **Point** (all related functionality moved in it)
    - **ScoreBoard** (all related functionality moved in it)
    - **EmptyCell** (all related functionality moved in it)
    - **OutOfMatrixChecker** (all related functionality moved in it)
    - **RandomUtils** (all related functionality moved in it)

2. Reformatted the source code:

* + 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**.
  + All fields were made accessed with **this.**

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 and restructured code:

|  |  |
| --- | --- |
| **Refactored data** | **Original data** |
| Command.cs: -… | GameFifteen.cs  -… |
| EqualMatrixChecker.cs: -… | GameFifteen.cs  -… |
| MatrixEmptyCellRandomizator.cs: | GameFifteen.cs  -… |
| MatrixGenerator.cs: | GameFifteen.cs  -… |
| MatrixRenderer.cs:  Holds only methods | GameFifteen.cs  -… |
| Player.cs: | GameFifteen.cs  -… |
| Point.cs:  - row (with property “Row”); -col (with property “Col”) | GameFifteen.cs  -… |
| Scoreboard.cs: | GameFifteen.cs  -… |
| EmptyCellMover.cs: | GameFifteen.cs  -… |
| OutOfMatrixChecker.cs:  Holds only methods | GameFifteen.cs  -… |
| RandomUtils.cs:  Holds only methods | GameFifteen.cs  -… |

1. Extracted methods:

|  |  |
| --- | --- |
| **Refactored data** | **Original data** |
| Command.cs: -… | GameFifteen.cs  -… |
| EqualMatrixChecker.cs: CheckMatrix(int[,] currentMatrix) | GameFifteen.cs  -… |
| MatrixEmptyCellRandomizator.cs: Point Randomize(int[,] matrix) | GameFifteen.cs  -… |
| MatrixGenerator.cs:  GenerateMatrix() | GameFifteen.cs  -… |
| MatrixRenderer.cs: Render(int[,] matrix) | GameFifteen.cs  -… |
| Player.cs: | GameFifteen.cs  -… |
| Point.cs: | GameFifteen.cs  -… |
| Scoreboard.cs: | GameFifteen.cs  -… |
| EmptyCellMover.cs:  MoveEmptyCell(Point emptyPoint, Point newPoint, int[,] matrix) | GameFifteen.cs  -… |
| OutOfMatrixChecker.cs:  CheckIfOutOfMatrix(Point point, int length) | GameFifteen.cs  -… |
| RandomUtils.cs:  GetRandomNumber()  GetRandomNumber(int max)  GetRandomNumber(int min, int max) | GameFifteen.cs  -… |

1. Constants refactor:

|  |  |
| --- | --- |
| **Refactored data** | **Original data** |
| GAME\_DIRECTIONS\_COUNT  GAME\_BOARD\_SIZE  INIT\_POINT\_POSITION  GAME\_DIRECTIONS\_COUNT  INIT\_POINT\_POSITION  MIN\_RANDOM\_NUMBER  MAX\_RANDOM\_NUMBER  MAX\_RANDOM\_DIRECTION\_NUMBER  INITIAL\_MATRIX\_NUMBER  INITIAL\_EMPTY\_CELL |  |