Instructions

This exercise has to be completed and submit it the day of the next lecture (deadline is 28-02-2023 23:59). To deliver the exercises, upload only the `.java` files containing your solution in Autolab.

It is possible to work in pairs but, in this case, each person has to hand in an individual submission. Additionally, during the next session, you might be asked to explain your solution. In case you are not able to properly explain the solution and answer related questions, the whole exercise will be considered as failed.

Exercise

Write a Java program that fulfills the same requirements of the first assignment:

> Design a simple program to play a simplified version of checker. Specifically, the program has to print the board and then alternatively ask the correct player to insert the old and the new coordinates of the piece to move. The system has to check whether the coordinates refer indeed to a player's owned piece and that the new position fulfills the requirements (i.e., diagonal forward move in an empty cell). The program has to continue until the user terminates it by entering a coordinate as value -1. No jumps (or multiple jumps) and no piece crowning should be implemented.

This time, however, **you have to implement the assignment using object orientation**.

Here is an example of "gameplay" (in blue are the inputs provided by the user):

```
  0 1 2 3 4 5 6 7   <- X axis
 +---------------+
```

```
0 |  1   1   1   1 |
1 |1   1   1   1   |
2 |  1   1   1   1 |
3 |                |
4 |                |
5 |2   2   2   2   |
6 |  2   2   2   2 |
7 |2   2   2   2   |
  +----------------+
   0 1 2 3 4 5 6 7

Turn of player no. 1
Coordinate of piece to move
  Enter X: 1
  Enter Y: 2

Coordinate of new position
  Enter X: 0
  Enter Y: 3
Piece moved!

   0 1 2 3 4 5 6 7   <- X axis
  +----------------+
0 |  1   1   1   1 |
1 |1   1   1   1   |
2 |      1   1   1 |
3 |1               |
4 |                |
5 |2   2   2   2   |
6 |  2   2   2   2 |
7 |2   2   2   2   |
  +----------------+
   0 1 2 3 4 5 6 7

Turn of player no. 2
Coordinate of piece to move
  Enter X: 0
```

```
  Enter Y: 5

Coordinate of new position
  Enter X: 1
  Enter Y: 4
Piece moved!

   0 1 2 3 4 5 6 7   <- X axis
  +----------------+
0 |  1   1   1   1 |
1 |1   1   1   1   |
2 |      1   1   1 |
3 |1               |
4 |  2             |
5 |    2   2   2   |
6 |  2   2   2   2 |
7 |2   2   2   2   |
  +----------------+
   0 1 2 3 4 5 6 7

Turn of player no. 1
Coordinate of piece to move
  Enter X: 2
  Enter Y: 1

Coordinate of new position
  Enter X: 1
  Enter Y: 2
Piece moved!

   0 1 2 3 4 5 6 7   <- X axis
  +----------------+
0 |  1   1   1   1 |
1 |1       1   1   |
2 |  1   1   1   1 |
3 |1               |
4 |  2             |
```

```
5 |    2   2   2   |
6 |  2   2   2   2 |
7 |2   2   2   2   |
  +----------------+
   0 1 2 3 4 5 6 7

Turn of player no. 2
Coordinate of piece to move
  Enter X: -1
```

Initial code

The code you must start from is the following (no package declaration, class name `Checker`). Also, please add all the code in the same file (i.e., add all other classes not as public classes):

```
import java.util.Scanner;

class Piece {
  // ..
}

// ...

public class Checker {



}
```