Due Date: 24.03.2025 23:59

Homework 1
CENG431 – Building Software Systems

In this homework, you are expected to implement a game application called “Duo Card Game” in Java.
You should fulfill the concepts:

- Object Oriented Analysis and Design,
- Object Orientation Fundamentals,
- Inheritance, Polymorphism, Abstract Classes, Interfaces
- Collections,
- UML,
- Mediator Design Pattern,
- CSV file I/O

In this game, 2-4 players sit in a circular arrangement. The number of players is determined randomly at the beginning of the game. The game's objective is to be the first player to get rid of all cards in each round. At the end of each round, the winner (of the round) earns points based on the remaining cards of other players. The first player to reach **500 points** wins the game.
In this game, there is a 109-card card deck. During the game, there are two piles: **Draw Pile** and **Discard Pile**. The card deck is described below.

- **Number Cards (76 cards)**:
  - 19 Blue (0-9), 19 Green (0-9), 19 Red (0-9), 19 Yellow (0-9).
  - Each color has one "0" card and two of each other number (1-9).
- **Action Cards (33 cards)**:
  - **Draw Two (8 cards):** 2 per color. The next player draws 2 cards and loses their turn.
  - **Reverse (8 cards):** 2 per color. Changes the turn order.
  - **Skip (8 cards):** 2 per color. The next player is skipped.
  - **Wild (4 cards):** 1 per color. Allows the player to choose the next color.
  - **Wild Draw Four(4 cards):** 1 per color. Forces the next player to draw 4 cards, but can only be played if the player has no matching color card.
  - **Shuffle Hands (1 card):** All player hands are shuffled and redistributed. The player who played the card chooses the next color.

In the game setup, each player draws a random card from the card deck; the player with the highest value card starts as the dealer. The dealer shuffles the deck (including the cards for determining the dealer) and deals **7 cards** to each player. The dealer draws one card from the deck, gives it to the next player, and does this until each player (including the dealer) has 7 cards. The dealing direction is left. After the dealing, the remaining deck is the **Draw Pile**. The top card from the Draw Pile is drawn, and the **Discard Pile** is started. If the top card drawn from the **Draw Pile** is an **Action Card**, its effect is followed immediately.

During the gameplay, players place a card on the Discard Pile that matches the **color, number, or symbol** of the top card (of the Discard Pile). For example, if the top card on the Discard Pile is a blue 5, the next player must play a blue card or any colored 5. The player will **randomly** decide whether to throw a 5 of any color or a blue card (if all of them are in their hand). If they decide to throw a blue card, they will throw the blue card with **the highest score** in their hand. Alternatively, the player may play a Wild card. If a player cannot play a card, they must draw a card from the Draw Pile. If the drawn card can be played, the player may place it immediately. Otherwise, the turn moves to the next player. The player to the left of the dealer starts the game; therefore, the gameplay direction is left (until a Reverse card is played).

In this game, **Action Cards** introduce special effects:

- **Draw Two:** This card can only be played if it matches the color of the top card in the Discard Pile or if another "Draw Two" card is played. The next player must draw **two cards (from the top of the Draw Pile)** and **skip their turn when a player plays this card**. The same rule applies if this card is drawn at the start of the game.
- **Reverse:** When this card is played, the direction of play is reversed. If the gameplay direction is **left**, it will be now **right**, and vice versa. This card can only be played if it matches the color of the top card in the Discard Pile or if another "Reverse" card is played. If this card is revealed at the start of the game, the dealer plays first, and the game continues in the opposite direction (right instead of left).
- **Skip:** When this card is played, the next player is skipped and does not take their turn. This card can only be played if it matches the color of the top card in the Discard Pile or if another "Skip" card is played. If a "Skip" card is revealed at the beginning of the game, the first player (to the left of the dealer) is skipped, and the next player takes their turn instead.
- **Wild:** This card allows the player to **choose the next color to be played**. The player should select the color they have the most cards of. If they have an equal number of different colors, they may choose a color randomly. The player **can** play this card even if they have other playable cards in their hand. If a Wild Card is revealed at the start of the game, the player to the left of the dealer **chooses the starting color**.
- **Wild Draw Four:** The player who plays this card gets to **choose the next color** for the game. Additionally, the next player must **draw four cards (from the top of the Draw Pile)** and **skip their turn**. However, this card **can only** be played if the player has **no other playable card matching the current color** in the Discard Pile. If this card is revealed at the start of the game, it must be placed back into the Draw Pile, and a new card is drawn to start the Discard Pile.
- **Shuffle Hands:** When this card is played, **all players must place their hands into a Shuffle Pile**; **the player who played the card** shuffles the Shuffle Pile and then redistributes the cards starting from the next player to the **left**, continuing until all cards are dealt. This means some players may end up with more or fewer cards than they originally had. The game then continues in its current turn order. This card also functions as a **Wild Card**, meaning the player who played it gets to **choose the next color**. If this card is revealed at the start of the game, the player to the **left** of the dealer chooses the starting color.

In this game, a round ends when a player plays their last card. If the last card played is a **Draw Two** or **Wild Draw Four**, the next player still draws the required cards, and those points are added to the round's total. If the Draw Pile runs out, shuffle the **Discard Pile** to form a new **Draw Pile**.

At the end of a round, the remaining cards in opponents' hands determine the score of the winner:

- **Number Cards (0-9):** Face value.
- **Draw Two, Reverse, Skip:** 20 points each.
- **Wild, Wild Draw Four:** 50 points each.
- **Shuffle Hands:** 40 points.
- The first player to reach **500 points** wins.

At the end of each round, record the game status in a CSV file similar to below. You can name the players as Player1, Player2, etc.

Round	Player 1	Player 2	Player 3	Player 4
Round 1	0	100	0	0
Round 2	200	0	0	0
Round 3	300	0	0	0
Winner	Player 1			

After you finish your implementation, please draw a UML class diagram of your project and include it in your ZIP file in PDF format.

Important Notes:

1. **Do NOT request inputs in your app**. Printing the following is enough:

- the game setup
  - the dealer,
  - the players and their 7 cards,
  - the starting card of Discard Pile
- the gameplay
  - the game direction
  - Player N
    - Play card Y (if it is an action card show its effect)
    - Draw card X (if ever draw)
  - The round end
- the game-winner
  - Player N won the game
  - Score: 5XY (e.g. 503)

2. You should use relative paths (e.g. Files/sample.csv) instead of absolute paths (e.g. C:\\user\\eclipse-workspace\\MyProject\\Files\\sample.csv).

3. To support **Turkish characters,** you may need to change your project's text file encoding to UTF8: Right-click on your project (in package explorer) → Properties → Text file encoding → Other → UTF8 → Apply.

4. You are expected to write clean, readable, and tester-friendly code. Please try to maximize reusability and prevent redundancy in your methods.

Assignment Rules:

1. In this lecture's homework, cheating is NOT allowed. If any cheating has been detected, they will be graded as **zero,** and there will be no further discussion.

2. You are expected to submit your homework in groups. Therefore, only one of you will be sufficient to submit your homework.
3. Make sure you export your homework as a **Visual Studio Code project**. You can use other IDEs as well; however, you must test if it is supported by VS Code. If the project import is not accomplished in VS Code, you will lose points.
4. Submit your homework through MS Teams.
5. Your exported Java Project should have the following naming format with your assigned group ID (which will be announced on MS Teams) as given below:

   G05_CENG431_HW1

   Also, the zip folder that your project is in should have the same name

   G05_CENG431_HW1.zip
6. Please beware that if you do not follow the assignment rules for exporting and naming conventions, you will lose points.
7. Please be informed that your submissions may be anonymously used in software testing and maintenance research studies. Your names and student IDs will be replaced with non-identifying strings. If you do not want your submissions to be used in research studies, please inform the instructor (Dr. Tuglular) via e-mail.