

Kode tersebut merupakan sebuah kode program yang dibuat dengan menyertakan sebuah konsep *Inheritance*, *Polymorphism*, *Encapsulation*, *Getter and Setter*, *Interface*. Untuk pemahaman lebih lanjut mengenai kode program diatas, berikut adalah penjelasannya :

1. ***Interface Playable:***

- a. ***StartGame()***: Metode ini digunakan untuk memulai permainan. Pada implementasinya, permainan akan terus berlanjut selama permainan belum berakhir.
- b. ***DisplayBoard()***: Metode ini bertanggung jawab untuk menampilkan kondisi terkini papan permainan. Isinya adalah tampilan papan permainan di layar.
- c. ***MakeMove(int row, int col, Player player)***: Metode ini memungkinkan pemain membuat langkah pada posisi tertentu di papan permainan. Posisi langkah ditentukan oleh *row* dan *col*, dan pemain yang melakukan langkah adalah *player*.
- d. ***IsGameOver()***: Metode ini memeriksa apakah permainan sudah berakhir. Permainan dianggap berakhir jika ada pemenang atau papan penuh.

2. ***Abstract class Player:***

- a. ***Class*** ini adalah kelas abstrak yang merepresentasikan pemain dalam permainan. Setiap pemain memiliki nama (*name*) dan simbol (*symbol*) unik.
- b. ***Constructor*** (*public Player(String name, char symbol)*) digunakan untuk menginisialisasi nama dan simbol pemain.
- c. ***Getter*** dan ***Setter*** (*getName()*, *getSymbol()*, *setName(String name)*, *setSymbol(char symbol)*) digunakan untuk mengakses dan memodifikasi atribut pemain.

3. **Kelas TicTacToe:**

- a. ***Konstruktor*** (*public TicTacToe(String player1Name, String player2Name)*) menginisialisasi objek TicTacToe dengan membuat papan

permainan (*board*), dua pemain (*player1* dan *player2*), dan pemain yang sedang giliran (*currentPlayer*).

- b. Metode ***initializeBoard()*** digunakan untuk mengisi papan permainan dengan karakter '-' yang menunjukkan bahwa seluruh papan masih kosong.
- c. Metode ***StartGame()*** adalah inti dari permainan. Pemain secara bergantian membuat langkahnya, dan setiap langkah akan memicu pengecekan kondisi kemenangan atau seri.
- d. Metode ***DisplayBoard()*** menampilkan papan permainan ke layar.
- e. Metode ***MakeMove()*** memungkinkan pemain untuk membuat langkahnya pada posisi yang diinginkan, dengan pengecekan kevalidan langkah.
- f. Metode ***IsGameOver()*** memeriksa apakah permainan sudah berakhir.
- g. Metode ***CheckWin()*** dan ***IsBoardFull()*** digunakan untuk menentukan apakah ada pemain yang menang atau papan penuh.

4. Kelas ***PlayerImpl***:

- a. Mengimplementasikan kelas abstrak ***Player***.
- b. Mewarisi konstruktor dan fungsionalitas dasar dari ***Player***.
- c. Dapat memiliki metode atau perilaku tambahan khusus untuk seorang pemain jika diperlukan.
- d. Dengan kata lain, kelas ini mengimplementasikan kelas abstrak *Player*. Dalam hal ini, ia hanya berfungsi sebagai implementasi konkret dari kelas *Player*.

5. Cara Kerjanya:

- a. Program menginisialisasi permainan Tic-Tac-Toe dengan dua pemain.
- b. Pemain bergantian memasukkan langkah mereka (baris dan kolom) melalui *input* konsol.
- c. Permainan memeriksa apakah ada pemenang atau seri setelah setiap langkah.
- d. Permainan terus berlanjut hingga ada pemenang atau papan penuh.
- e. Hasil akhir (pemenang atau seri) ditampilkan di konsol.

Implementasi ini menggunakan *input/output* konsol dasar dan cocok untuk permainan Tic-Tac-Toe sederhana berbasis baris perintah.