

```
public class Ficha {  
    private int valor1;  
    private int valor2;  
  
    public Ficha(int valor1, int valor2) {  
        this.valor1 = valor1;  
        this.valor2 = valor2;  
    }  
  
    public int getValor1() {  
        return valor1;  
    }  
  
    public int getValor2() {  
        return valor2;  
    }  
  
    public boolean coincideCon(int valor) {  
        return valor1 == valor || valor2 == valor;  
    }  
  
    @Override  
    public String toString() {  
        return "[" + valor1 + "|" + valor2 + "];"  
    }  
}
```

```

import java.util.ArrayList;
import java.util.List;

public class Jugador {
    private String nombre;
    private List<Ficha> fichas;

    public Jugador(String nombre) {
        this.nombre = nombre;
        this.fichas = new ArrayList<>();
    }

    public String getNombre() {
        return nombre;
    }

    public void agregarFicha(Ficha ficha) {
        fichas.add(ficha);
    }

    public boolean puedeJugar(int valor) {
        for (Ficha ficha : fichas) {
            if (ficha.coincideCon(valor)) {
                return true;
            }
        }
        return false;
    }

    public Ficha jugarFicha(int valor) {
        for (Ficha ficha : fichas) {
            if (ficha.coincideCon(valor)) {
                fichas.remove(ficha);
                return ficha;
            }
        }
        return null;
    }

    public boolean tieneFichas() {
        return !fichas.isEmpty();
    }

    @Override
    public String toString() {
        return nombre + " tiene " + fichas.size() + " fichas";
    }
}

```

```

import java.util.Collections;
import java.util.List;
import java.util.ArrayList;
import java.util.Scanner;

public class Domino {
    private List<Ficha> fichasDisponibles;
    private List<Ficha> mesa;
    private Jugador jugador1;
    private Jugador jugador2;
    private int valorMesa;

    public Domino() {
        fichasDisponibles = new ArrayList<>();
        mesa = new ArrayList<>();
        jugador1 = new Jugador("Jugador 1");
        jugador2 = new Jugador("Jugador 2");
        valorMesa = -1; // No hay valor en la mesa al inicio
        generarFichas();
        repartirFichas();
    }

    private void generarFichas() {
        for (int i = 0; i <= 6; i++) {
            for (int j = i; j <= 6; j++) {
                fichasDisponibles.add(new Ficha(i, j));
            }
        }
        Collections.shuffle(fichasDisponibles); // Barajar las fichas
    }

    private void repartirFichas() {
        // Repartir 7 fichas a cada jugador
        for (int i = 0; i < 7; i++) {
            jugador1.agregarFicha(fichasDisponibles.remove(0));
            jugador2.agregarFicha(fichasDisponibles.remove(0));
        }
    }

    private void mostrarMesa() {
        if (mesa.isEmpty()) {
            System.out.println("Mesa vacía.");
        } else {
            System.out.print("Mesa: ");
            for (Ficha ficha : mesa) {
                System.out.print(ficha + " ");
            }
            System.out.println();
        }
    }
}

```

```

    }
    }

    private void turnoJugador(Jugador jugador) {
        System.out.println(jugador.getNombre() + ", es tu turno.");
        mostrarMesa();
        System.out.println(jugador);
        Scanner scanner = new Scanner(System.in);

        if (valorMesa == -1) {
            // Si la mesa está vacía, cualquier ficha puede colocarse
            System.out.println("Elige una ficha para jugar (0-" + (jugador.fichas.size() - 1) + "): ");
            int opcion = scanner.nextInt();
            Ficha fichaElegida = jugador.jugarFicha(opcion);
            mesa.add(fichaElegida);
            valorMesa = fichaElegida.getValor1() == valorMesa ? fichaElegida.getValor2() :
fichaElegida.getValor1();
        } else {
            // Si hay un valor en la mesa, el jugador debe jugar una ficha que coincida
            if (jugador.puedeJugar(valorMesa)) {
                System.out.println("Elige una ficha que coincida con " + valorMesa + ": ");
                int opcion = scanner.nextInt();
                Ficha fichaElegida = jugador.jugarFicha(valorMesa);
                mesa.add(fichaElegida);
                valorMesa = fichaElegida.getValor1() == valorMesa ? fichaElegida.getValor2()
: fichaElegida.getValor1();
            } else {
                System.out.println("No puedes jugar. Pierdes tu turno.");
            }
        }
    }

    public void jugar() {
        while (jugador1.tieneFichas() && jugador2.tieneFichas()) {
            turnoJugador(jugador1);
            if (jugador1.tieneFichas() && jugador2.tieneFichas()) {
                turnoJugador(jugador2);
            }
        }
        if (jugador1.tieneFichas()) {
            System.out.println(jugador1.getNombre() + " gana!");
        } else {
            System.out.println(jugador2.getNombre() + " gana!");
        }
    }

    public static void main(String[] args) {
        Domino domino = new Domino();
    }

```

```
        domino.jugar();  
    }  
}
```