

Laboratorio 1: Clases

Análisis

Funcionalidades:

- Cada jugador lanza los 2 dados y acumula puntos hasta que alcance 100 puntos. Cuando esto ocurre se muestra como el ganador.
- Si un jugador obtiene un valor igual a 1 en alguno de los dados, pierde sus puntos acumulados en ese turno y pasa el turno al siguiente jugador.
- Cada jugador tiene la opción de detener su turno y conservar los puntos acumulados.
- El juego termina cuando un jugador alcanza o supera los 100 puntos, y se muestra el ganador.
- Se debe llevar un seguimiento de los puntos acumulados por cada jugador.

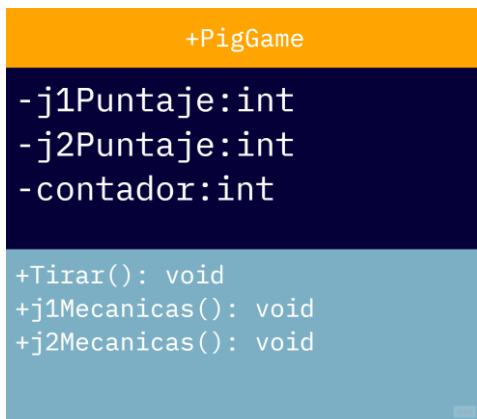
Descripción de clases y atributos:

- PigDiceGame: Clase principal que contiene el método main para ejecutar el juego.
- static Scanner scan: Instancia del escáner para leer la entrada del usuario.
- static Random generator: Instancia del generador de números aleatorios para obtener valores de los dados.
- int j1Puntaje: Variable para almacenar los puntos acumulados por el jugador 1.
- int j2Puntaje: Variable para almacenar los puntos acumulados por el jugador 2.
- int contador: Variable para llevar un seguimiento de los puntos acumulados en el turno actual.

Descripción de métodos:

- Tirar(): Método que simula el lanzamiento de un dado y devuelve un valor aleatorio entre 1 y el número máximo que puede tener la cara del dado.
- j1Mecanicas(): El jugador 1 tiene la opción de lanzar los dados y acumular puntos hasta que obtenga un 1 en algún dado o decida detenerse. Si obtiene un 1, pierde los puntos acumulados en ese turno. Si decide detenerse, los puntos acumulados se suman a su puntuación total. Una vez que alcance o supere los 100 puntos, se muestra que ha ganado el juego y se termina el programa.
- j2Mecanicas(): Similar al método j1Mecanicas() pero para el jugador 2.

Diagramas de Clases UML:



PigDiceGame utiliza los métodos **Tirar()**, **j1Mecanicas()** y **j2Mecanicas()** para implementar la lógica del juego.

Solución al problema y código fuente:

Lab1.codigo

Link de repositorio: <https://github.com/Ikeel04/Lab1.POO.git>