Tenemos las clases Egg, EndGame, FramePrincipal, KeyMonitor, login, main, MyThreadClass, Nodo, Player, ranking, Sentido y Snake.

La única clase que se puede ejecutar es main, y esta lo que hace es crear un nuevo objeto login.

La clase login crea un JOptionPane, que pregunta al usuario por su nombre de jugador, y tiene tres botones: Uno para jugar, otro para acceder al ranking y un último para salir del juego.

El botón de jugar crea un nuevo FramePrincipal, dónde se ejecuta el juego.

El botón de ranking abre una nueva ventana de ranking.

El botón de salir finaliza la ejecución del programa.

FramePrincipal extiende de JFrame, y carga la vista general del juego. Esto incluye el mapa, y los elementos del juego (serpiente y huevos). Las funciones de FramePrincipal són getEgg (permite obtener el objeto huevo que hay actualmente), setEgg (permite sustituir el huevo actual por uno nuevo), checkEgg (controla en todo momento si la serpiente se ha comido al huevo o no, y en caso de hacerlo, otorga al usuario 100 puntos y genera un nuevo huevo, a demás de hacer la serpiente un nodo más grande), y por último la función paint, que dibuja todos los objetos en pantalla.

El constructor de FramePrincipal crea una nueva ventana y genera un keyListener para así poder controlar a la serpiente. Por cada vez que se presione una tecla, la serpiente se moverá un cuadrado en esa dirección y comprobará si se ha comido el huevo o no.

La clase Snake tiene por propiedades el sentido de la serpiente, un ArrayList de nodos, un booleano eatenEgg que indica si la serpiente se ha comido o no el huevo, y 4 variables para almacenar las coordenadas X e Y de la cabeza de la serpiente y la cola.

Snake en su constructor, creará una nueva serpiente, siempre mirando hacia arriba con 6 nodos.

Los métodos de Snake són avanzar, avance, draw, y printSnake.

Avanzar comprueba la dirección de la serpiente, y la hace avanzar en dicha dirección.

Avance hace avanzar una posición a cada uno de los nodos de la serpiente en la dirección que corresponda, para ello primero almacena la posición actual con .setLastPosX/.setLastPosY, y a continuación mueve dicho nodo en la dirección del nodo anterior, o en el caso del primer nodo (la cabeza), en la dirección actual de la serpiente. Esta función también comprueba si la cabeza coincide con alguna posición de uno de los nodos (es decir, la serpiente se ha mordido a si misma), caso en el cuál lanzará una excepción EndGame con el mensaje “Te has mordido. Fin del juego”. También se comprueba al hacer avanzar la cabeza de la serpiente que se mantenga dentro de los límites del mapa, o de lo contrario lanza también la excepción EndGame, pero con el mensaje “Te has salido del tablero. Fin del juego”.

La función draw dibuja cada uno de los nodos de la serpiente

La función printSnake dibuja la animación de cuando nuestra serpiente se mueve.

La excepción EndGame lo que hace es finalizar el juego cerrando la ventana actual, y abre un nuevo ranking, añadiendo el registro del usuario actual.

La clase Ranking se conecta a la base de datos mediante el driver de Java para mysql, y busca en la base de datos “snake”. Dicha clase tiene las funciones getPlayers que devuelve un ArrayList de jugadores obtenidos desde la base de datos, getUsers que se conecta a la base de datos y obtiene los usuarios del juego con sus puntos obtenidos, addPlayer que añade un nuevo usuario con su puntuación final a la base de datos y su constructor, que genera un nuevo MessageDialog con una lista de todos los jugadores con sus respectivos puntos ordenados de mayor a menor puntuación.

Al cerrar dicha ventana, también finaliza la ejecución del programa.

La clase Nodo tiene 4 atributos, posX, posY, lastPosX y lastPosY, que definen la posición actual del nodo y la anterior que tuvieron. Esto permite al siguiente nodo saber dónde debe posicionarse cuando se cree. La clase nodo tiene 2 constructores, uno con parámetros y otro sin. El que no tiene parámetros, genera una posición aleatoria, mientras que el que tiene parámetros genera un nodo en la posición especificada.

La clase Sentido por defecto es hacia abajo, y simplemente indica a la serpiente la dirección en la que moverse. Puede tener 4 valores en total: Derecha, Izquierda, Arriba y Abajo.

La clase Player tiene 2 propiedades, name y pts. Esta clase crea objetos de jugador a partir de un nombre y puntuación de cada uno de los registros en la base de datos y el nombre y puntuación del jugador actual. Tiene setters y getters para las propiedades, y un toString que se utiliza en el ranking para mostrar el nombre y la puntuación de cada jugador.

Finalmente, la clase Egg simplemente tiene por propiedades las posiciones X e Y, con sus correspondientes setters y getters. En su constructor, se genera una posición aleatoria dentro del tablero de juego.