Q_OBJECT

https://doc.qt.io/qt-6/qobject.html#Q_OBJECT

Personajes

#Int vida
#Int velocidad

Jugador

-int energia;
-int municion;

Jugador(int vida1,int municion1,int velocidad1,int energia1)
+int Vida() const
+int Municion() const
+int Velocidad() const
+int Energia() const
+void operator=(const Jugador& nuevoJugador)
+bool operator==(const Jugador& nuevoJugador)

Enemigos

+Enemigos(int vida1,int velocidad1)
+int Vida() const
+int Velocidad() const
+void operator=(const Enemigos& nuevoEnemigo)
+bool operator==(const Enemigos& nuevoEnemigo)

personajemov
-qreal x = 520; -qreal y = 327; -qreal xIn, yIn; -float velIn, theta, dir; -float tiempoTrans; -const int maxInterval = 200; -QSize viewRect; -bool flag = true; -bool meta; -bool caer; - int cont = 0; -int spriteX = 0; -int spriteY = 120; -int spriteWidth = 40; -int spriteHeight = 20; -QPixmap sprite; -QPixmap spriteSheet; -QTimer *timer; -QTimer *timer1; -QSet<int> keysPressed; -QElapsedTimer TiempoA; -QElapsedTimer TiempoD; -QElapsedTimer TiempoW;
PersonajeMov(QGraphicsView *view, float velIn, float +theta, QGraphicsItem* im = nullptr); +void keyPressEvent(QKeyEvent *event) override; +void moveBy(int dx, int dy); +void moveKy(int dx, int dy); +void keyReleaseEvent(QKeyEvent *event); +void movParabolico(float *dt); +int getPosicionX(); +int getPosicionY(); +bool getMeta(); +bool getCaer(); +void setSprite(int dir); +bool isOnGround();

MainWindow

-QGraphicsScene * scene1, * scene2, * scene3, * scene4;
-QGraphicsPixmapItem * fig1, * fig2, ... , * fig21;
-QGraphicsPixmapItem * balaEnSprite;
-QGraphicsPixmapItem * enemigo1, * enemigo2, ... , * enemigo9;
-QGraphicsPixmapItem * bala1, * bala2, ... , * bala9;
 -QGraphicsLineItem * l1, * l2;
-QPointF center, balaDirection;
-QString nombre;
-QTimer * misilTimer, * launchTimer, * moveTimer, * balaTimer, * terminarLvl2, *jumpTimer, movOndularT, fig20Timer, enemyShootTimer;
-double reductionStep;
-int tiempoTotal ,misilCount ,lvl , intentos,cantidadEne , puntaje;
-double angle ,radius;
-qreal initialVelocity, gravity, jumpTime, startX, startY, velocityX, velocityY;
-bool pierde, canLaunch, isJumping, balaEnMovimiento, Municion;
-bool bala1Existe, bala2Existe, ... , bala9Existe;
-QList<QGraphicsRectItem> obst;
 -QList<QGraphicsRectItem> muros;
-Jugador PP;
-PersonajeMov jug1;

-void launchMisil();
-void launchBala();
-void setupScene1();
-void setupScene2();
-void setupScene3();
-void setupScene4();
-void loadCurrentScene();
-void resetScene1();
-void resetScene2();
 -void resetScene3();
 -void checkCollisions();
-void startJump();
-void updateJump();
-void updateBala();