## class main window

•

• Ui::MainWindow \*ui

•

• GameScene \*gameScene

- o explicit MainWindow(QWidget \*parent = nullptr); → Constructor
  - o  $\sim$ MainWindow();  $\rightarrow$  Destructor
- o void startGame();
  - void exitGame();
  - void on\_startGameButton\_clicked();
    - o void on\_exitGameButton\_clicked();

## class obstacle

None explicitly defined; relies on QGraphicsPixmapItem for attributes such as position, size, and pixmap.

explicit Obstacle(QGraphicsItem \*parent =
nullptr); → Constructor

## class Gamescene:public Qmainwindow

QGraphicsScene \*scene,QGraphicsView
\*view,QGraphicsTextItem
\*scoreText,QGraphicsRectItem \*lifeBar, QTimer
\*movementTimer,QTimer \*scoreTimer,Mario
\*mario,QList<Obstacle \*>
obstacles,QList<QGraphicsPixmapItem \*>
backgrounds,int score,int lives

- explicit GameScene(QWidget \*parent = nullptr); → Constructor
- ~GameScene(); → Destructor
- void createObstacles();
- void updateScore(int points);
- void updateLives();
- void centerViewOnMario();
- void finishLevel();

## class Mario : public QObject, public QGraphicsPixmapItem

•

- float velocityX
- float velocityY
- bool onGround
- QTimer \*physicsTimer
- explicit Mario(QGraphicsItem \*parent = nullptr); → Constructor
- o void move();
- void marioHitObstacle();
- void keyPressEvent(QKeyEvent \*event) override;
- void keyReleaseEvent(QKeyEvent \*event) override;
- void jump();
- void checkCollisions();
- void applyGravity();
- void updatePosition();

class MovingObstacle : public QObject, public QGraphicsPixmapItem

•

• QTimer \*movementTime, int direction, int speed, int range, int currentDistance

- o explicit MovingObstacle(QGraphicsItem \*parent =
  - nullptr); → Constructor
- o void setSpeed(int speed);
  o void setRange(int range);
- void obstacleOutOfBounds();
- void move();