**Set the gameState width and height values to the Level’s width and height values.**

**package** sonar.gamestates.states.levels;

**import** sonar.gamestates.Screen;

**import** sonar.gamestates.states.levels.stages.entities.animations.tiles.Tile;

**import** sonar.gamestates.states.levels.stages.entities.animations.tiles.TileManager;

**public** **class** Level

{

//Level retrieves all the tiles that build our level.

**private** LM lm;

**private** LevelBuilder buildLevel;

**protected** Level(LevelBuilder buildLevel, LM lm)

{

**this**.buildLevel = buildLevel;

**this**.lm = lm;

lm.getStage().setHeight(buildLevel.getHeight());

lm.getStage().setWidth(buildLevel.getWidth());

//Must set the width, height, and tiles

//of the GameState class with the Level's

//width, height and tiles.

}

**void** update()

{

}

**void** render(**int** xScroll, **int** yScroll, Screen screen)

{

screen.setOffset(xScroll, yScroll);

lm.getStage().drawTiles(xScroll, yScroll, screen);

//Call the gameState class function here!

}

LM getLm(){**return** lm;}

LevelBuilder getBuildLevel(){**return** buildLevel;}

}