

CMPE 160 PROJECT 2 REPORT

Name: Eren AKÇİN

Student ID: 2021400060

Section: 01

CODE EXPLANATION:

The project has 5 classes for game objects and 1 class(Environment) for combining other classes. Here is a short explanation of the classes.

Player.class:



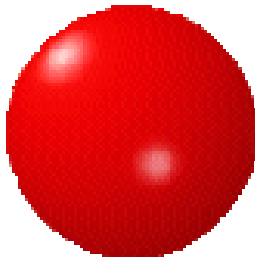
Contains the movement and drawing functions of the player. For movement functions it takes “**Left and right**” buttons as inputs.

Arrow.class:



Contains the movement and drawing functions of the arrow. For movement functions it takes “**Space Bar**” for input if the arrow is passive (not triggered) at that moment.

Ball.class:



This class has movement, drawing, and splitting functions for the ball. For movement, this class uses the time-dependent free fall formula. While moving, the movement method also checks the boundaries of the canvas -left, right, and bottom sides. The splitting function is called in arrow-ball collision situations. It splits the ball if it has a level higher than 1, or it will just disappear.

Bar.class:

This class creates a rectangle bar at the bottom of the screen which has an initial color of yellow. As time passes, the bar starts to shorten and change its color to a reddish color. If the bar's length becomes zero, the game ends.

Environment.class:

That is the class where all the game objects are combined. We check game-ending and game-winning situations in this class. For example Ball-Arrow and Ball-Player collisions. Also, it has a game loop that draws the objects, frame by frame, and a game restarter responsible for replaying or quitting.

GAMEPLAY VIDEO:

<https://drive.google.com/file/d/11c6mawslI6yguM4sy0iGT42O7i5POszk/view?usp=sharing>

In the loss situation caused by a ball-player collision, In the end-game screen, I left the ball and player object on the screen for a better display of the cause of loss **on purpose**. I know that is not the same as the gameplay video you shared with us but I think this implementation looks better and I kept the code like that.