

CET 350 Technical Computing Using Java  
Group Program 5 Bouncing ball with obstructions Applet

Using an IDE, write a java applet that will draw a screen box for the bouncing ball to move within.  
Include two scrollbars: speed and size.  
Include two labels for the scrollbars: speed and size.  
Include three buttons: pause, run, and quit.  
The control area will contain the scrollbars, buttons, and labels.  
The ball will move on a diagonal until it touches a side of the box.  
When the ball touches a side of the box it will change direction based on the side that was touched.  
The pause run buttons will alternately pause and resume the ball movement.  
The quit button will terminate the applet.  
The scrollbars will change the values of the speed and size of the ball.  
The mouse will be used to drag and place rectangles into the screen area.  
The rectangles that are completely covered by another rectangle will not be included.  
When the new rectangle completely covers existing rectangles, remove all covered rectangles.  
The rectangles will not exceed the screen box.  
The rectangle will not be placed if the ball is inside it.  
When the ball touches a side of a rectangle it will change direction based on the side that was touched.  
The size of the ball will be restricted to its current allowable space.  
Use a ball object with a paint routine.  
The ball should not destroy the boundary of the box or the rectangles.  
When the mouse is clicked inside a rectangle, all rectangles will be removed that contain the mouse click position.

Supply the Java source code.

Labeled with:

- a. Your name(s)
- b. CET 350
- c. Group number
- d. Your e-mail address.

Name your program Bounce2.java

At the top of the program, place a comment header containing:

- a. Your name(s)
- b. CET 350
- c. Group number
- d. Your e-mail address

Properly comment the program.

Only turn in a program that will compile without errors. A program that has compile errors will be returned unchecked. Late points will continue to accumulate until a program is turned in that compiles without errors.

