Practical 4

**Write a java application to demonstrate 5 bouncing balls of different colors using threads.**

**Filename: BouncingBalls.java**

**Code:**

**import** javax.swing.\*;

**import** java.awt.\*;

**import** java.util.Random;

**public** **class** BouncingBalls **extends** JFrame {

**private** **static** **final** **int** ***WIDTH*** = 800;

**private** **static** **final** **int** ***HEIGHT*** = 600;

**private** **static** **final** **int** ***BALL\_RADIUS*** = 20;

**private** **static** **final** **int** ***BALL\_SPEED*** = 5;

**private** Ball[] balls;

**public** BouncingBalls() {

setTitle("Bouncing Balls");

setSize(***WIDTH***, ***HEIGHT***);

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

setResizable(**false**);

setLocationRelativeTo(**null**);

balls = **new** Ball[5];

**for** (**int** i = 0; i < balls.length; i++) {

balls[i] = **new** Ball(getRandomColor());

**new** Thread(balls[i]).start();

}

BallPanel ballPanel = **new** BallPanel();

add(ballPanel);

setVisible(**true**);

}

**private** Color getRandomColor() {

Random random = **new** Random();

**return** **new** Color(random.nextInt(256), random.nextInt(256), random.nextInt(256));

}

**private** **class** BallPanel **extends** JPanel {

@Override

**protected** **void** paintComponent(Graphics g) {

**super**.paintComponent(g);

**for** (Ball ball : balls) {

g.setColor(ball.getColor());

g.fillOval(ball.getX(), ball.getY(), ***BALL\_RADIUS*** \* 2, ***BALL\_RADIUS*** \* 2);

}

}

}

**private** **class** Ball **implements** Runnable {

**private** Color color;

**private** **int** x;

**private** **int** y;

**private** **int** dx;

**private** **int** dy;

**public** Ball(Color color) {

**this**.color = color;

x = **new** Random().nextInt(***WIDTH*** - ***BALL\_RADIUS*** \* 2);

y = **new** Random().nextInt(***HEIGHT*** - ***BALL\_RADIUS*** \* 2);

dx = **new** Random().nextBoolean() ? ***BALL\_SPEED*** : -***BALL\_SPEED***;

dy = **new** Random().nextBoolean() ? ***BALL\_SPEED*** : -***BALL\_SPEED***;

}

**public** Color getColor() {

**return** color;

}

**public** **int** getX() {

**return** x;

}

**public** **int** getY() {

**return** y;

}

@Override

**public** **void** run() {

**while** (**true**) {

move();

repaint();

**try** {

Thread.*sleep*(20);

} **catch** (InterruptedException e) {

e.printStackTrace();

}

}

}

**private** **void** move() {

**if** (x + dx < 0 || x + dx > ***WIDTH*** - ***BALL\_RADIUS*** \* 2) {

dx = -dx;

}

**if** (y + dy < 0 || y + dy > ***HEIGHT*** - ***BALL\_RADIUS*** \* 2) {

dy = -dy;

}

x += dx;

y += dy;

}

}

**public** **static** **void** main(String[] args) {

SwingUtilities.*invokeLater*(BouncingBalls::**new**);

}

}

**Output:**

****