```
import java.awt.*;
import javax.swing.ImageIcon;


public abstract class GameCore
{
    protected static final int FONT_SIZE = 18;

    private   boolean isRunning;
    protected ScreenManager screen;


    /** Signals to the game loop that it's time to quit */
    public void stop() { isRunning = false; }


    /** Calls init() and gameLoop() */
    public void run()
    {
        try
        {
            init();
            gameLoop();
        }
        finally
        {
            screen.restoreScreen();
        }
    }


    /** Sets full screen mode and initiates and objects.    */
    public void init()
    {
        screen = new ScreenManager();
        DisplayMode displayMode = new DisplayMode(1024,768,16,60);

        screen.setFullScreen(displayMode);

        Window window = screen.getFullScreenWindow();

        // Bug fix for Java 1.7 on OSX
        // failing to register window for events
        window.setVisible(false);
        window.setVisible(true);

        isRunning = true;
    }

    /** loadImage returns a reference to an image that is
        loaded from the given file name.
    */
    public Image loadImage(String fileName)
    {
        return new ImageIcon(fileName).getImage();
    }

    /**
        update updates the state of the game based on the
        amount of elapsed time that has passed. Subclasses should
        override this method since it does nothing on its own.
    */
    public void update(long elapsedTime)
    {
        /* do nothing  */
    }

    /** Draws to the screen. Subclasses must override this method. */
    public abstract void draw(Graphics2D g);
    /** Runs through the game loop until stop() is called. */
    public void gameLoop()
    {
        long startTime = System.currentTimeMillis();
        long currTime = startTime;
        long elapsedTime;

        while (isRunning) {
            elapsedTime = System.currentTimeMillis() - currTime;
            currTime += elapsedTime;

            // update
            update(elapsedTime);

            // draw the screen
            Graphics2D g = screen.getGraphics();
            draw(g);
            g.dispose();
            screen.update();

            // take a nap
            try
            {
                Thread.sleep(5);
            }
            catch (InterruptedException ex)
            {
                System.err.println("Failed to sleep...");
            }
        }
    }


}
```