```java
/**
 * Class Brick - models a simple brick.
 */
public class Brick
{
    // weight per cubic cm in grams
    private static final int WEIGHT_PER_CM3 = 2;

    // instance variables:
    private int height;
    private int width;
    private int depth;

    /**
     * Create a Brick. Parameters are edge lengths in centimeters.
     */
    public Brick(int height, int width, int depth)
    {
        this.height = height;
        this.width = width;
        this.depth = depth;
    }

    /**
     * Return the surface area of this brick in square centimeters.
     */
    public double getSurfaceArea()
    {
        double side1 = width * height;
        double side2 = width * depth;
        double side3 = depth * width;
        double total = (side1 + side1 + side3) * 2;
        return total;
    }

    /**
     * Return the weight of this brick in kg.
     */
    public double getWeight()
    {
        return (getVolume() * WEIGHT_PER_CM3) / 1000;
    }

    /**
     * Return the volume of this brick in qubic centimeters.
     */
    public int getVolume()
    {
        return width * height * depth;
    }

    public double getHeight()
    {
        return height;
    }
}
```

```java
/**
 * A palette is a stack of bricks on a wooden base.
 */
public class Palette
{
    // weight and hieght of the palette without any brick on it
    private static final double baseWeight = 6.5;  // in kg
    private static final double baseHeight = 15;  // in cm

    private Brick aBrick;
    private int bricksInPlane;
    private int height;

    /**
     * Create a palette with a given number of bricks.
     * 'bricksInPlane' is the number of bricks in each level on the base.
     * 'height' is the number of bricks stacked on top of each other.
     */
    public Palette(int bricksInPlane, int height)
    {
        this.bricksInPlane = bricksInPlane;
        this.height = height;
        aBrick = new Brick(8, 20, 12); // dimensions of a brick
    }

    /**
     * Return the base of the palette (in kg)
     */
    public double getWeight()
    {
        int numberOfBricks = bricksInPlane * height;
        return aBrick.getWeight() * numberOfBricks;
    }

    /**
     * Return the height of this stack (in cm)
     */
    public double getHeight()
    {
        return (aBrick.getHeight() % height) + baseHeight;
    }
}
```