```java
import sheffield.*;
/**
 * Class to implement Assignment2 for COM1003
 * @author Euan Rochester (ACA15ER)
 */
public class Picture{

  enum Tree{
    Background,
    Leaf,
    Trunk
  }

  final static int WIDTH = 1800;
  final static int HEIGHT = 150;
  final static int PICTURE_WIDTH = 300;
  final static int PICTURE_HEIGHT = 150;
  final static int NUM_TREES = 30;
  //trees can appear anywhere on the x axis
  final static double X_TRANS_SCALE = WIDTH;
  //less randomness scaling on y as it looks odd otherwise
  final static double Y_TRANS_SCALE = HEIGHT/5.0;

  public static void main(String[] args){
    //load the picture
    Tree[][] picture = new Tree[PICTURE_WIDTH][PICTURE_HEIGHT];
    EasyReader data = new EasyReader("picture.txt");
    for(int y=0;y<PICTURE_HEIGHT;y++){
      for(int x=0;x<PICTURE_WIDTH;x++){
        char val = data.readChar();
        //treeCel: like pixel, but for trees
        Tree treeCel;
        switch(val){
          case '0':
            treeCel = Tree.Background;
            break;
          case '1':
            treeCel = Tree.Leaf;
            break;
          case '2':
            treeCel = Tree.Trunk;
            break;
          default:
            System.err.println("Found bad input character '"+val+"' in data val");
            return;
        }
        picture[x][y] = treeCel;
      }
    }

    EasyGraphics gfx = new EasyGraphics(WIDTH,HEIGHT);
```

```
    //set background color
    gfx.setColor(0,64,128);
    //and fill with it
    gfx.fillRectangle(0,0,WIDTH,HEIGHT);

    for(int n=0;n<NUM_TREES;n++){
      //subtract picture width/2 to allow trunk (as opposed to the left edge of the tree image)
      //to be drawn all the way from the very left to the very right edge
      int xTrans = (int) (Math.random()*X_TRANS_SCALE) - PICTURE_WIDTH/2;
      int yTrans = (int) (Math.random()*Y_TRANS_SCALE);
      for(int y=0;y<PICTURE_HEIGHT;y++){
        for(int x=0;x<PICTURE_WIDTH;x++){
          int r = 0,g = 0,b = 0;
          //picture y is from top, canvas y from bottom so use HEIGHT-y to make it from top
          //and -1 to avoid off by 1 error
          switch(picture[x][PICTURE_HEIGHT-y-1]){
            case Background:
              continue;
            case Leaf:
              g = 128;
              break;
            case Trunk:
              break;
          }
          gfx.setColor(r,g,b);
          gfx.plot(x+xTrans,y+yTrans);
        }
      }
    }

  }
}
```