

*Happiness is having a large, loving, caring family in another city.*  
— George Burns (1896–1996).

**Notice that this assignment is due on Thursday afternoon! Start early!**

This is a team assignment, to be done in your newly assigned partnership. Your mission is to (1) gain more experience with writing classes, and (2) draw pretty pictures of a city’s skyline.

Create a directory called `skyline` to store your work. When you are finished, use `hsp` to submit your skyline directory. Indicate in program comments the full names of both authors (you and your partner), as well as roughly what fraction of the time each team member spent “driving” at the keyboard. You should aim to give all team members approximately the same amount of time driving. Good luck, and have fun! Remember that lab assistants are available in the evenings in CMC 306 to help out if you need it.

Many complicated movie scenes that you see involve lots of individually generated objects, all assembled together on one screen. A Pixar movie scene of a city—such as in *The Incredibles*—involves superimposing a number of buildings of different shapes and sizes. For this assignment, you will create a city landscape by drawing a number of buildings of varied sizes.

1. Copy `Canvas.java` into your `skyline` directory. Write a class called `Building` to represent a building in a cityscape. Specifically, include the following methods:

- a constructor
- `public void setHeight(int h)`  
(sets the height of the building to `h`)
- `public void setWidth(int w)`  
(sets the width of the building to `w`)
- `public void setColor(Color col)`  
(sets the color of the building to `col`)
- `public void setHorizontalLocation(int loc)`  
(sets how many pixels over from the left of the canvas the building begins)
- `public void draw(Canvas canvas, int canvasHeight)`  
(given a canvas and its height, draw the building)

For now, draw your building by drawing a rectangle of the appropriate color. (Putting a black border around it will make it look a lot more like a building.)

You should add on your own any private object variables that you need.

Test your code as you go along. In other words, create another class called `Skyline` that creates and draws some buildings. As you write methods in `Building`, test them in `Skyline`.

2. Once you’ve gotten the above working, add the following methods to your `Building` class:

- `public void setHeightRandom(int maxHeight)`  
(sets the height to a random value from 0 to `maxHeight`, inclusive)
- `public void setWidthRandom(int maxWidth)`  
(sets the height to a random value from 0 to `maxWidth`, inclusive)

- `public void setLocationRandom(int maxLoc)`  
(sets the horizontal location to a random value from 0 to maxLoc, inclusive)
- `public void setColorRandom()`  
(sets the building color randomly)
- `public int area()`  
(returns the area of the building—that is, width times height)

When you are completely finished with this question, test your code against the `Skyline.java` that's available on the webpage. It uses techniques that we're talking about this week, so you should be able to understand how it works by the time you submit your solutions. If you've done everything correctly, this Skyline program should give you a nice cityscape to admire.

3. Once all of the above is working (with good program style!), you'll have 95% of the credit for the assignment. For the last 5%, work to make your buildings look a little more like buildings. For example, you could add an antenna to the top of the building whose height is proportional to the rest of the building, or you could add vertical lines running down and/or across the building. You could also add windows—if you read ahead to the end of Chapter 5, you can have a different number of windows based on the height of the building; otherwise, you can have windows whose size scales with the size of the building.