public class CarRadio {
    // We decided that we need 12 buttons, each of which can remember an AM station and an FM station.
    // Since the FM stations are decimals, we'll use float for all.
    private final int BANDS = 2;  //
    private final int BUTTONS = 12;  //

    // What the buttons remember
    float [ ] [ ] frequency = {{0,0,0,0,0,0,0,0,0,0,0,0},  //AM band
                                {0,0,0,0,0,0,0,0,0,0,0,0}};  //FM band

    private boolean isOn;  // Do we pay attention to changes in state?
    private boolean isFM;  // When we select or change station, is it FM?

    // Data to check any supposed frequencies against those allowed.
    private final float buttonAMallowed [] = new float[ ];  //
    private final float buttonFMallowed [] = new float[ ];  //</p> 

    // ***********************[ Methods ]******************************/
    /** Select station to play from one of BUTTONS number of stations.
    The row is not needed, because it’s remembered by isFM.
    pre:  0 ≤ col ≤ BUTTONS
    post: the frequency in that button, a value in buttonAMallowed ∪ buttonFMallowed */
    float pressButton (int col){ ... }
    /** etc. */
    holdButton(int col) {... }
}
/*
We could have said
float [ ] [ ] frequency;
and then in the CarRadio constructor say
for (int row = 0; row < BANDS; row++)
{
    for (int col = 0; col < BUTTONS; col++)
    {
        frequency [row][col] = 0;
    }
}
*/

and while we’re at it, here’s some more constructor code—
for (int col = 0, freq = 50; freq <= 1610; col++, freq +=10)
{
    buttonAMallowed [col] = 530 + col * 10;
} 
for (int col =0, float freq=87.9; freq <= 107.9; col++, freq += 0.2)
{
    buttonFMallowed [col] = 87.9 + col * 0.2;
} */