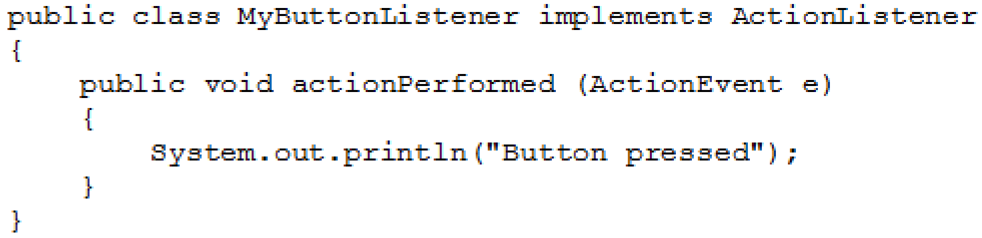
Students do, student exercises

1. Write a loop that it iterates (steps through the numbers) 1 – 8 (1, 2, 3…8)
2. Modify this program so that it instead iterates through the sequence 5 – 8 (5, 6, 7, 8)
3. Modify this program so that it prompts the user for two numbers, you can assume now that the first number entered by the user will always be less than or equal to the second number):
    - The first number will be the start of the number sequence e.g. 12
    - The second number will be the last number in the sequence e.g. 20
    - The program will iterate through this sequence e.g. 12, 13, 14…20
    - Question: what will happen if the user enters two identical numbers



```
public class MyButtonListener implements ActionListener
{
    public void actionPerformed (ActionEvent e)
    {
        System.out.println("Button pressed");
    }
}
```

4. Modify the program further so that when it prompts the user for the two numbers it will automatically determine which number is lower and that lower will number will be the start of the sequence. The other number will be the end of the sequence.
    - E.g. the user first enters 15 and then enters 3. The program will then iterate through the sequence 3, 4, 5…15
5. Modify this program even further so that it doesn’t increase the count by just one e.g. the previous example started at 3 and then increased the count by one so the next number in the sequence is 4. The program will prompt the user by the amount that count should be increased.
    - E.g. the user first enters 15 and then enters 3. As before the first number in the sequence will be 3 and the last number will be 15.
        - If in this version of the program (and with the previous two inputs) the user enters 3 for the count, then the program will iterate through the sequence 3, 6, 9, 12, 15
        - If in this version of the program (and with the previous two inputs) the user enters 5 for the count, then the program will iterate through the sequence 3, 8, 13. (Since 13 + 5 = 18 which exceeds the maximum bound in the sequence the last number that the program will count through is 13).
6. Write a new program that will prompt the user for the current year and error check the input.
    - The program prompts the user for the current year in the form of a numeric value e.g. 2017.
    - If the year is not one that is within the 21st century (2000 – 2099) then the program will display an appropriate and helpful error message (e.g. “Year must be in the range of 2000-2099”) and set the year to the default year of 2035).
    - The program will then display the current value for the year (either the value entered by the user if the year that was entered was valid or the default value if the year was not valid).
7. Modify the previous program (asks the user for the year).

- It will still prompt the user for a year and display an error message if the year is not within the range of 2000 – 2099.
- Instead of setting invalid values to the default value the program will instead **repeatedly ask** (using a do-while loop) for the year until a valid value has been entered.
    - E.g.
        - The user enters 1999 for the year. An error message will be displayed and the program again prompt the user for the year.
        - The user then enters 2135 for the year. Again an error message will be displayed and the program re-prompts the user for the year.
        - This time the user enters 2000 for the year. Since this value is within the valid range the program will stop prompting the user for the year and display the year entered.