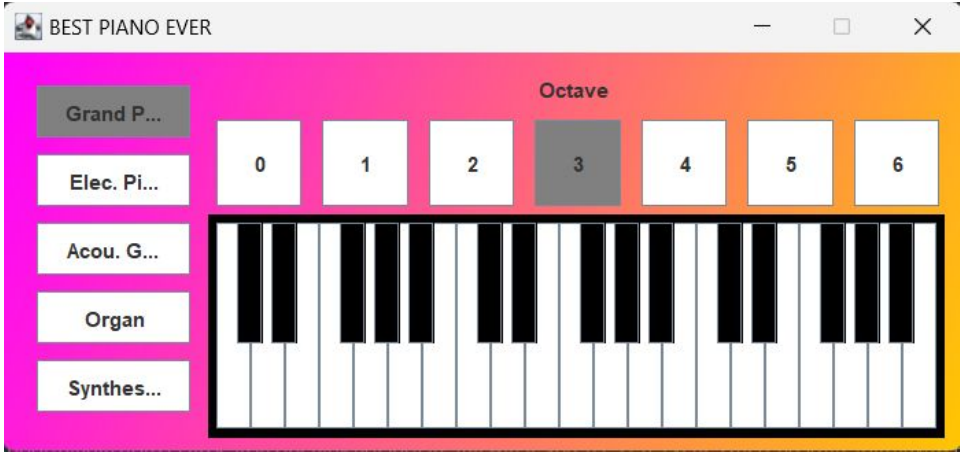
Virtual Piano

Ivan Faucett and Saad Chaudry

So what is this?

- A GUI that resembles a keyboard
- It has options to change the octave and the instrument currently being played
- Inspiration found from VSTs and a youtube video showcasing a piano built in swing



Packages

- Javax.swing
  - The main GUI package used
  - All of the components were from Swing
  - Swing is not thread safe and must be run under a new thread
- Java.awt
  - Color
  - ActionListener
- Javax.sound.midi
  - Used to provide sound for the instrument using MidiChannel,. MidiSystem, and Synthesizer.

Swing and AWT

- All the components were Swing based such as JFrame, JLayeredPane, and JButtons
- Swing is considered not thread-safe
- AWT was used only for its Color class
- Used for the background to be painted in a gradient color and coloring the buttons

JLayeredPane

- Traditionally many JPanels are created and placed in either a grid or some other type of layout in a JFrame
- Making a keyboard out of JButtons made it so JPanels could not be used
- JLayeredPane allows absolute positioning of components through (x, y) coordinates
- It also allows components to be stacked on top of each other.
- An issue that arose was the JLayeredPane took up the entire JFrame and as such all components had to be added to the JLayeredPane instead of adding them to different JPanels

JButton and ActionListener

- Each component except for one JLabel is a button
- Each button needed a new ActionListener to define what happens when clicking that button
- Each keyboard button is connected to a specific note
- Octave buttons change what notes the keyboard is on
- The Instrument buttons change what instrument patch the synthesizer is running

MIDI Implementation

- MIDI lets instruments, computers, and various other audio devices to all connect to one another
- The javax.sound.midi package allows for the ability to connect external devices and create virtual instruments using the Musical Instrument Digital Interface (MIDI) standard
- Our program uses MIDI messages to change pitch and timbre of the instrument
- Primarily, the program sends MIDI note messages to establish the note being played, and MIDI Program Change (PC) messages to change the sound of the instrument
- We also are utilizing the General MIDI specification, which adds standardized song files to produce the various sounds of the piano

javax.sound.midi Use

- When a key is pressed, the MIDI note value is sent to a method that plays the note,
- The playNote method first calls the startSynthesizer method, which creates a synthesizer object and establishes the synthesizer’s program change value
- The playNote method then calls the noteOn method in the MidiChannel interface (using the MIDI note value that was sent) to produce a sound at the correct pitch.
- The octave select buttons simply shift the MIDI note value up or down a value of 12, with each change of 1 for the MIDI note value representing a semitone
- The timbre select buttons on the left of the panel simply changes the PC value according to the instrument’s General MIDI PC value (i.e. the PC value for a grand piano is 1, for an electric piano is 5, etc.)