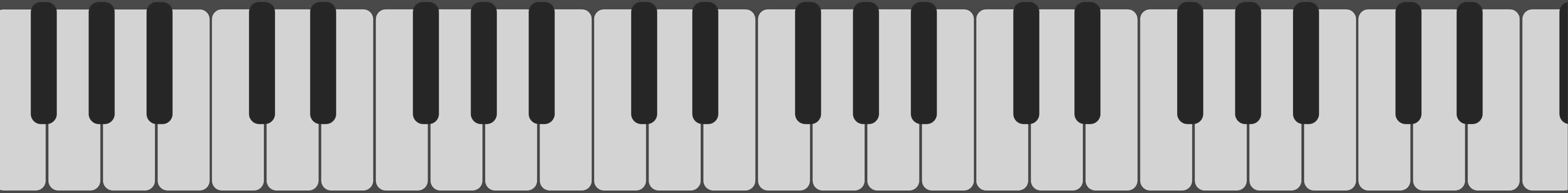
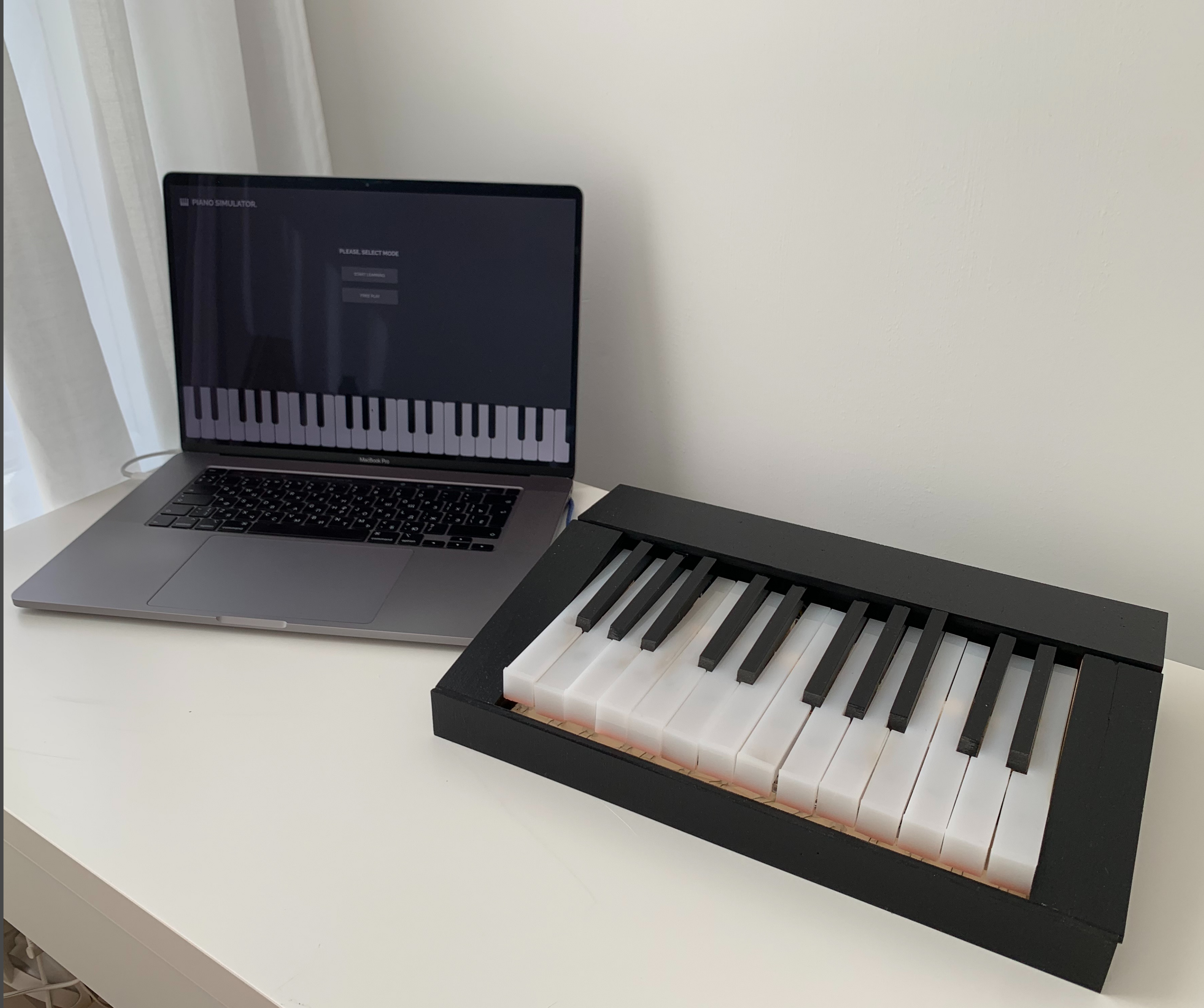
PIANO SIMULATOR

Alyona Radkevich



PIANO SIMULATOR

Piano Simulator is a 24-key piano keyboard connected to the software. It is aimed to teach users how to play basic piano keys, where they are located, and how to read notes.



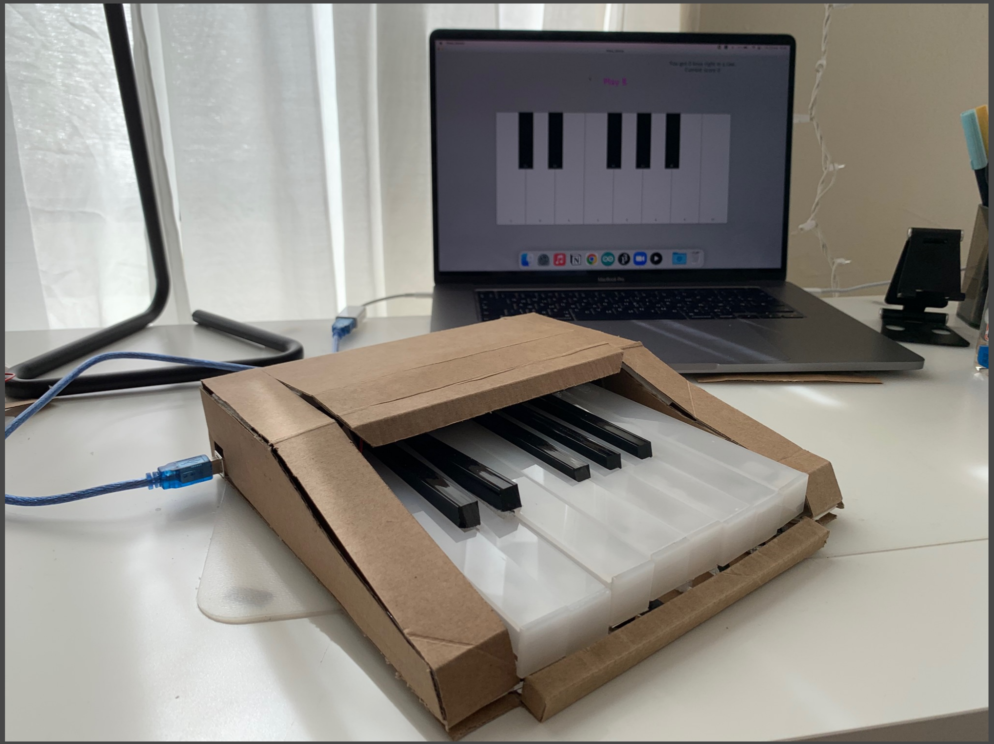
PIANO SIMULATOR

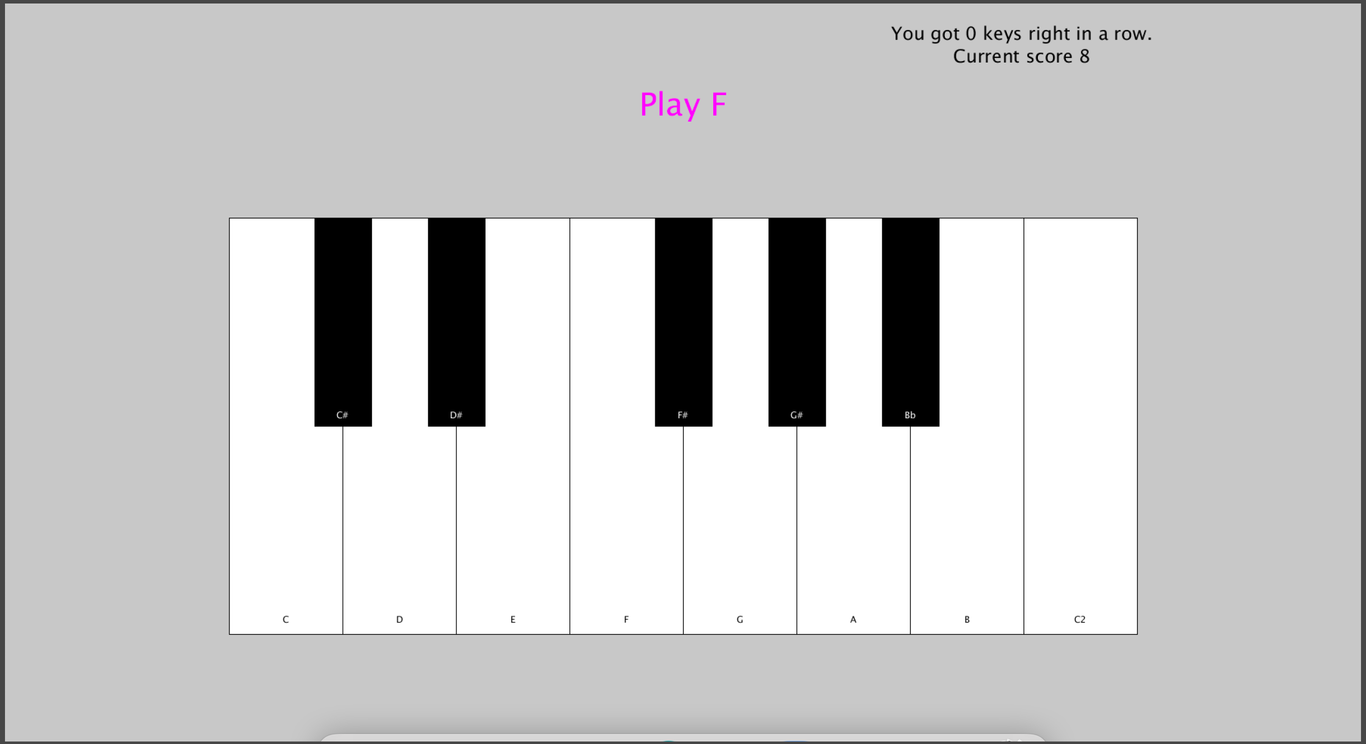
LED illuminated interface makes learning more engaging while the application guides users through short lessons.

PIANO SIMULATOR INSPIRATION

The idea came from the final project I did in MTEC 2280 class in spring 2022.

I thought there was so much more I could have done to make the project better.

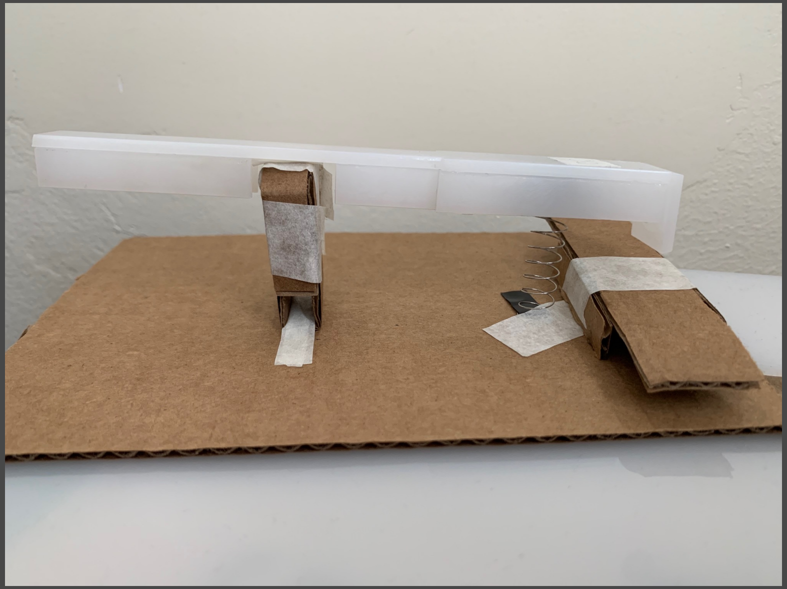


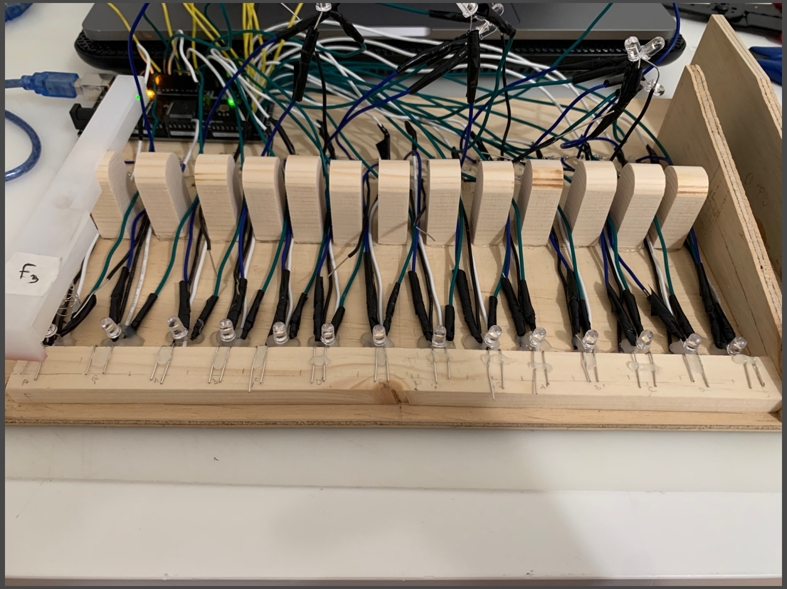


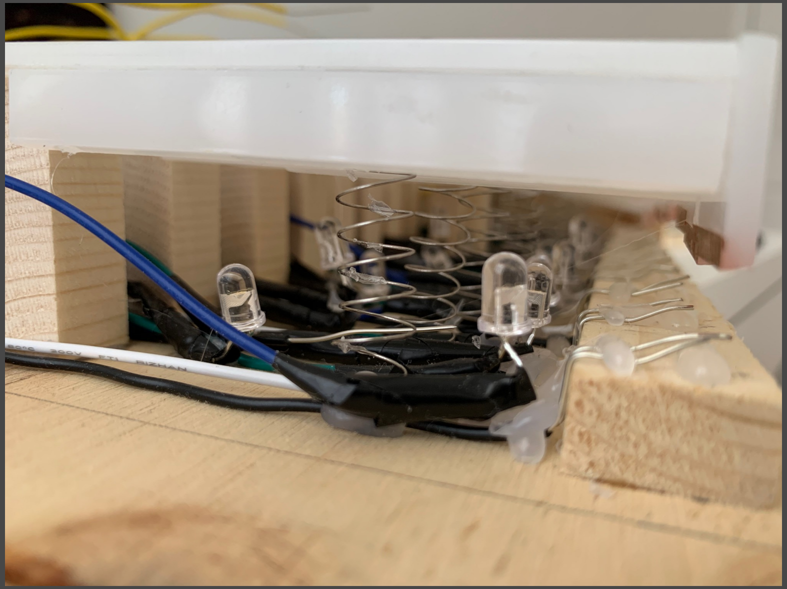
You got 0 keys right in a row.
Current score 8

Play F

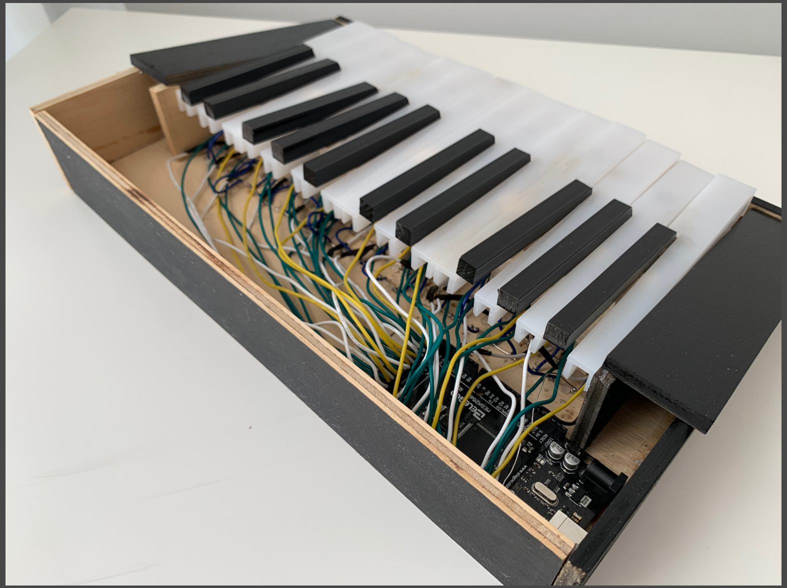
PIANO SIMULATOR
PIANO



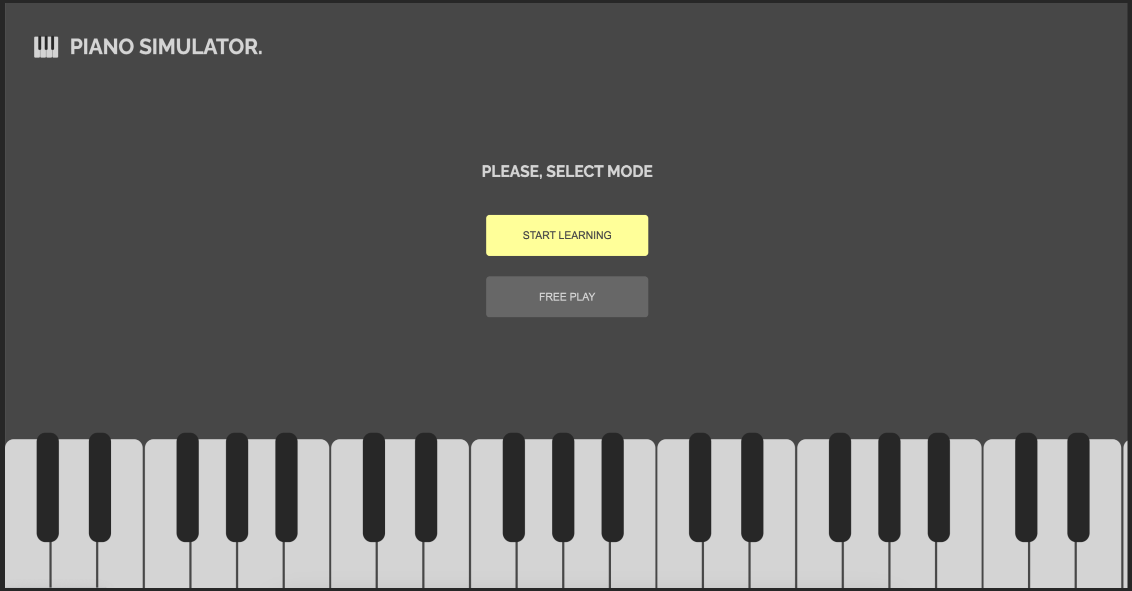


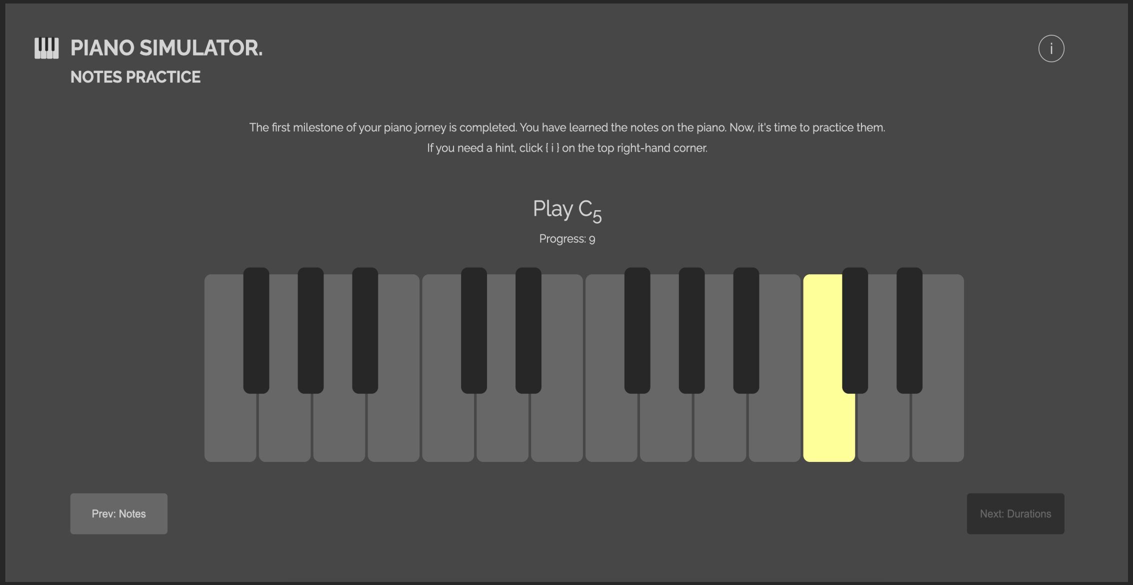


PIANO SIMULATOR
PIANO



PIANO SIMULATOR
WEB APPLICATION

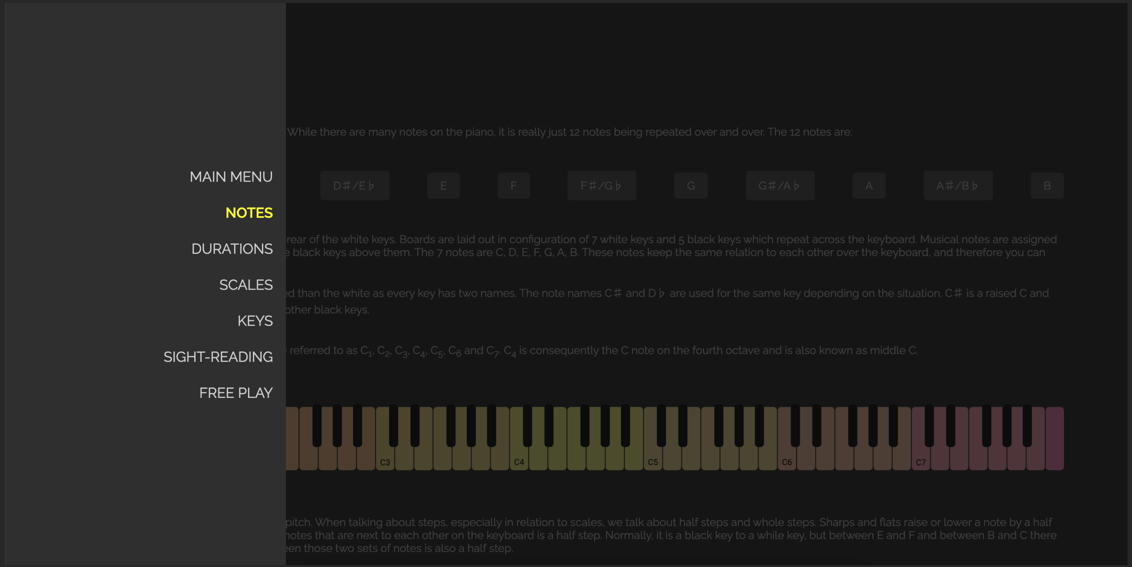




The first milestone of your piano jorney is completed. You have learned the notes on the piano. Now, it's time to practice them.
If you need a hint, click [i] on the top right-hand corner.

Play C_5

Progress: 9



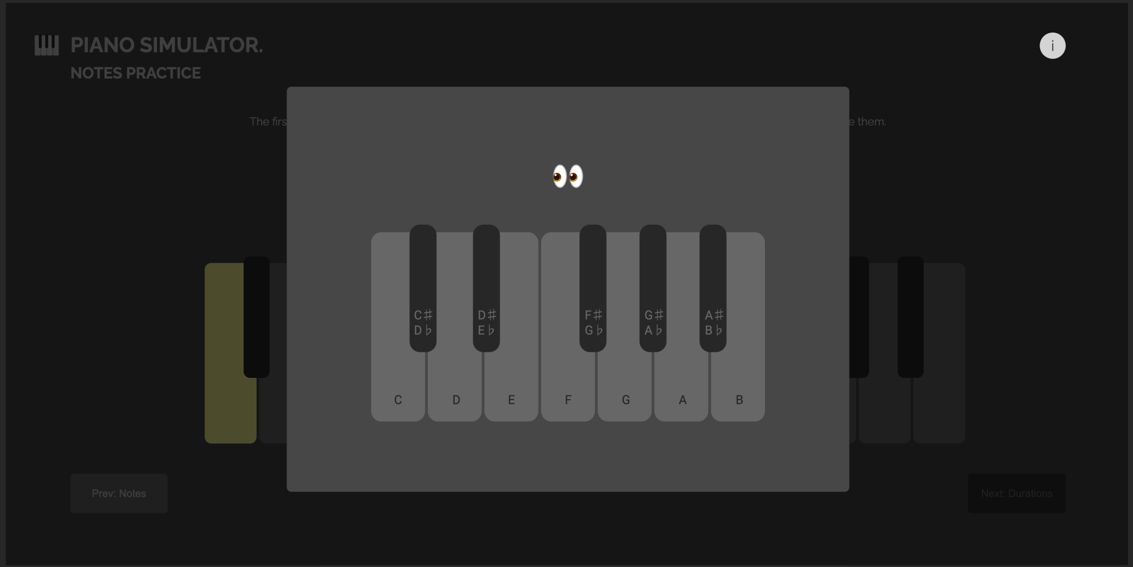
While there are many notes on the piano, it is really just 12 notes being repeated over and over. The 12 notes are:

rear of the white keys. Boards are laid out in configuration of 7 white keys and 5 black keys which repeat across the keyboard. Musical notes are assigned
e black keys above them. The 7 notes are C, D, E, F, G, A, B. These notes keep the same relation to each other over the keyboard, and therefore you can

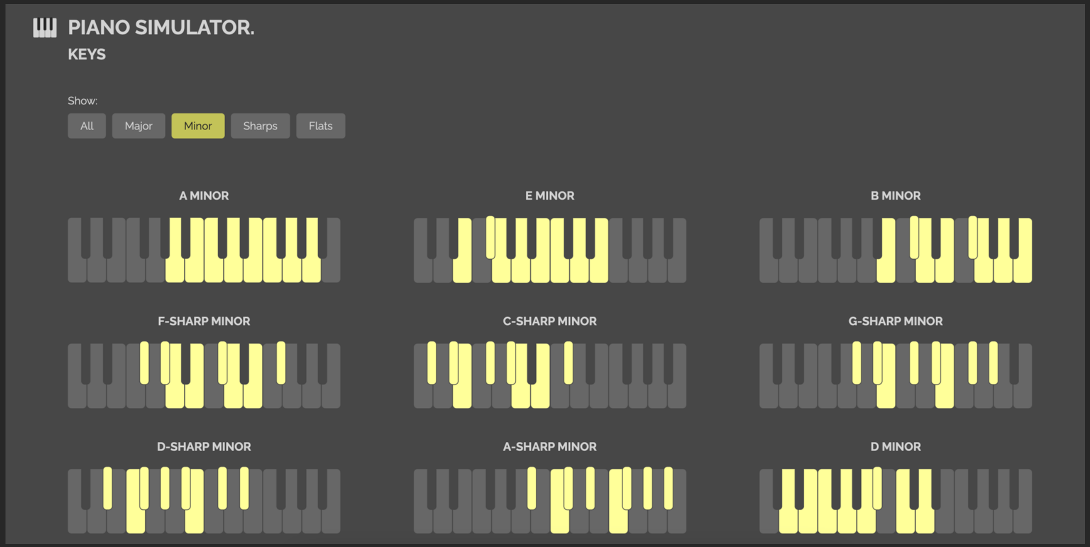
d than the white as every key has two names. The note names C♯ and D♭ are used for the same key depending on the situation. C♯ is a raised C and
other black keys.

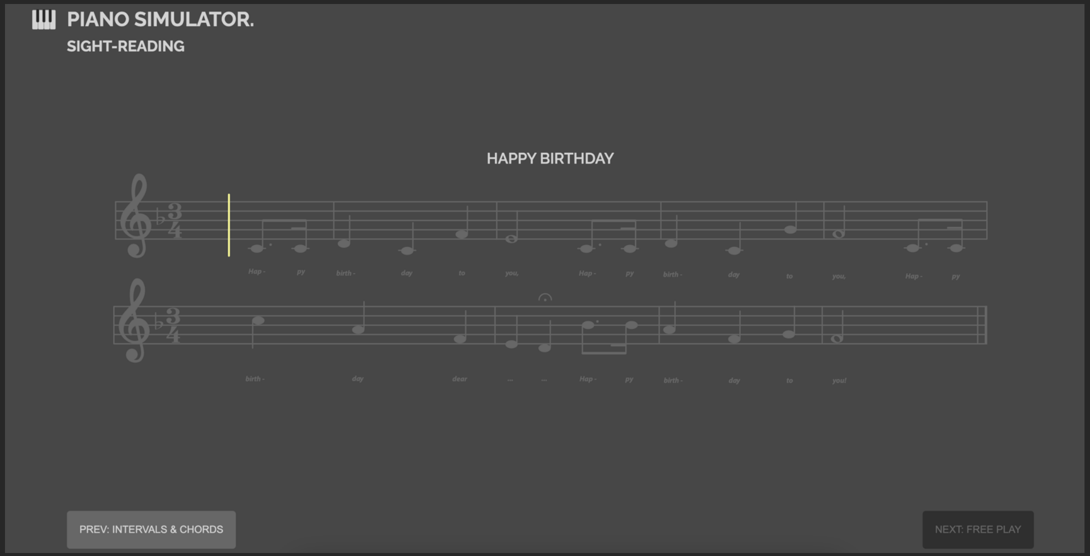
referred to as C_1, C_2, C_3, C_4, C_5, C_6 and C_7. C_4 is consequently the C note on the fourth octave and is also known as middle C.

pitch. When talking about steps, especially in relation to scales, we talk about half steps and whole steps. Sharps and flats raise or lower a note by a half
notes that are next to each other on the keyboard is a half step. Normally, it is a black key to a while key, but between E and F and between B and C there
een those two sets of notes is also a half step.

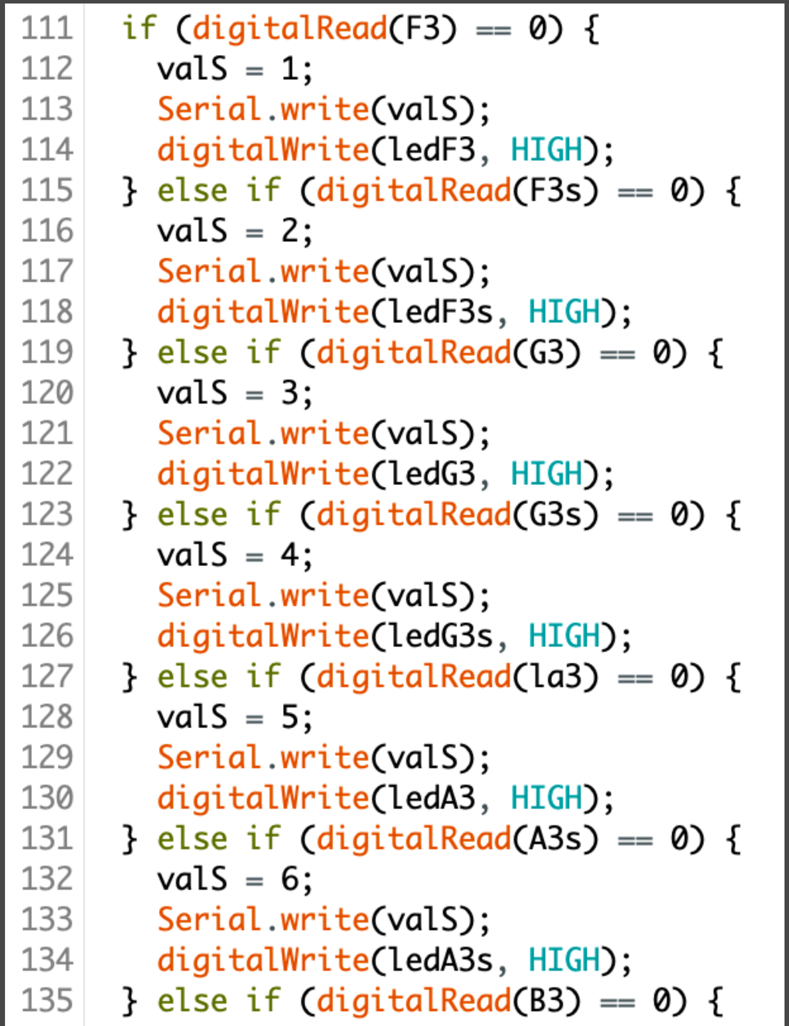


PIANO SIMULATOR
WEB APPLICATION

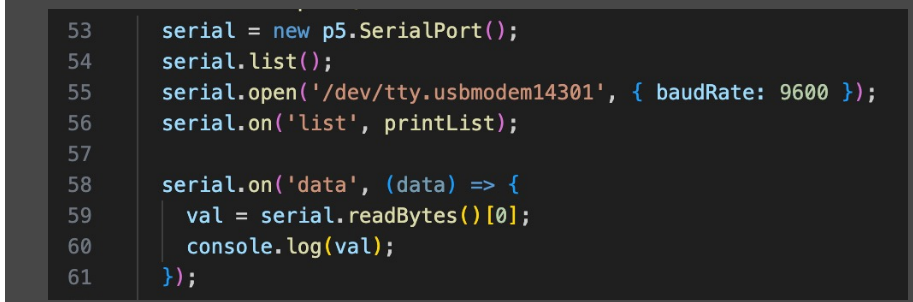



PIANO SIMULATOR.
SIGHT-READING
HAPPY BIRTHDAY
Hap - py birth - day to you, Hap - py birth - day to you, Hap - py
birth - day dear ... ... Hap - py birth - day to you!
PREV: INTERVALS & CHORDS
NEXT: FREE PLAY

PIANO SIMULATOR
SERIAL COMMUNICATION

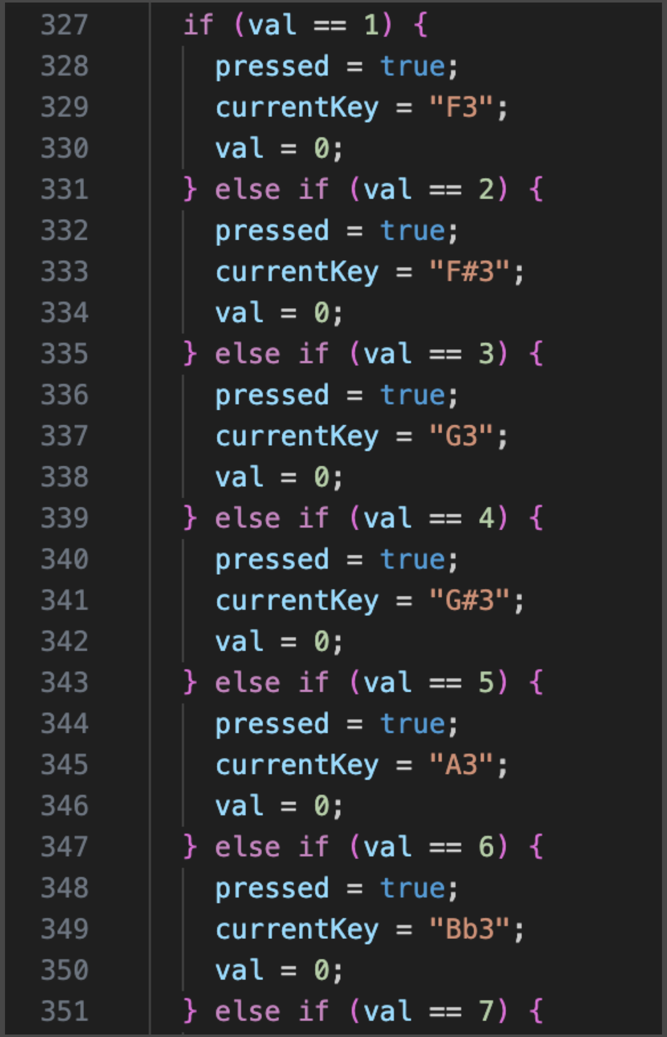


```
if (digitalRead(F3) == 0) {
  valS = 1;
  Serial.write(valS);
  digitalWrite(ledF3, HIGH);
} else if (digitalRead(F3s) == 0) {
  valS = 2;
  Serial.write(valS);
  digitalWrite(ledF3s, HIGH);
} else if (digitalRead(G3) == 0) {
  valS = 3;
  Serial.write(valS);
  digitalWrite(ledG3, HIGH);
} else if (digitalRead(G3s) == 0) {
  valS = 4;
  Serial.write(valS);
  digitalWrite(ledG3s, HIGH);
} else if (digitalRead(la3) == 0) {
  valS = 5;
  Serial.write(valS);
  digitalWrite(ledA3, HIGH);
} else if (digitalRead(A3s) == 0) {
  valS = 6;
  Serial.write(valS);
  digitalWrite(ledA3s, HIGH);
} else if (digitalRead(B3) == 0) {
```



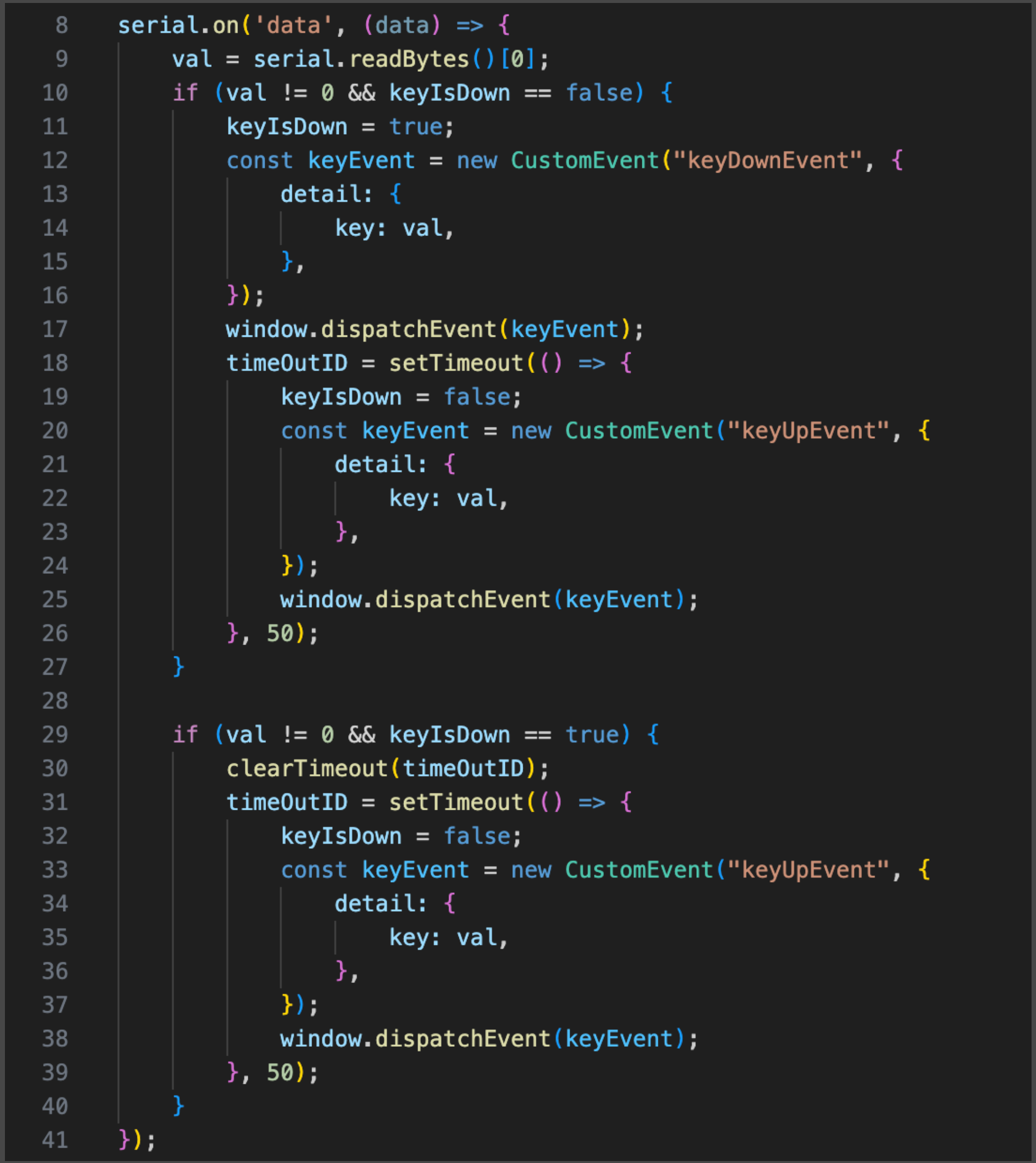
```
serial = new p5.SerialPort();
serial.list();
serial.open('/dev/tty.usbmodem14301', { baudRate: 9600 });
serial.on('list', printList);

serial.on('data', (data) => {
  val = serial.readBytes()[0];
  console.log(val);
});
```



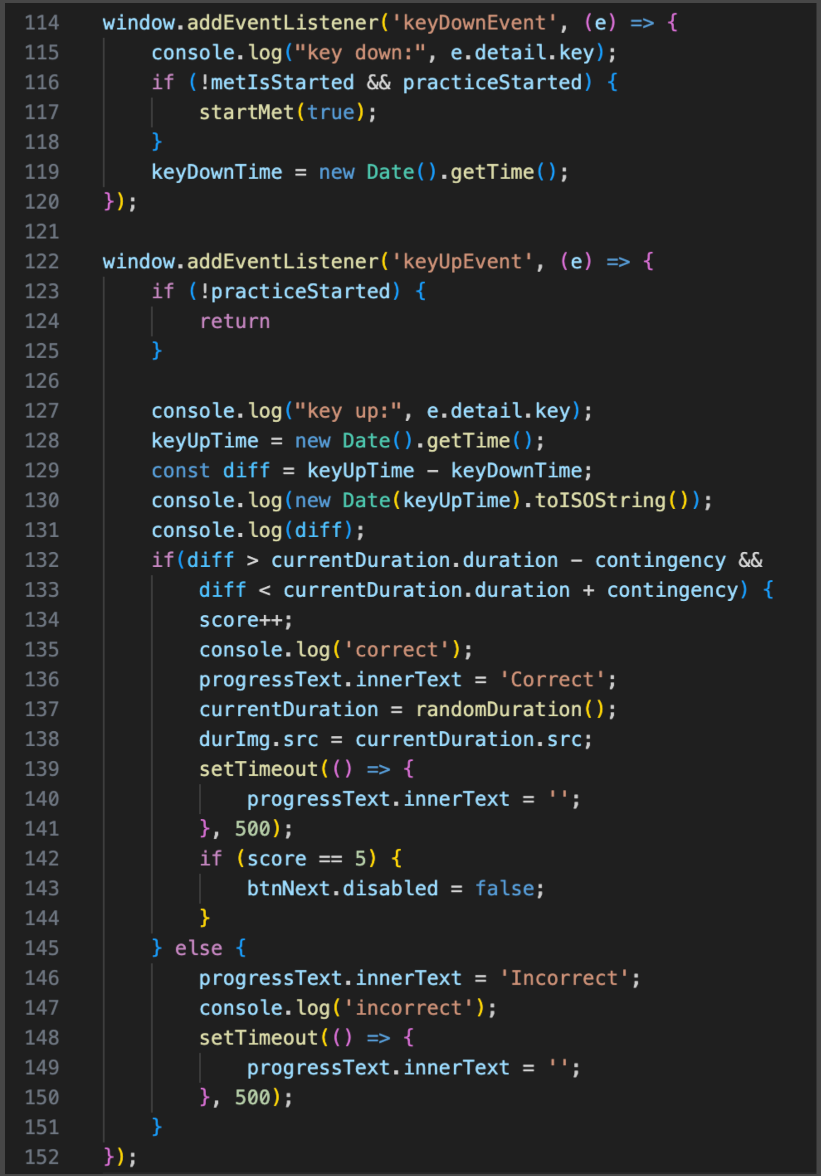
```
if (val == 1) {
  pressed = true;
  currentKey = "F3";
  val = 0;
} else if (val == 2) {
  pressed = true;
  currentKey = "F#3";
  val = 0;
} else if (val == 3) {
  pressed = true;
  currentKey = "G3";
  val = 0;
} else if (val == 4) {
  pressed = true;
  currentKey = "G#3";
  val = 0;
} else if (val == 5) {
  pressed = true;
  currentKey = "A3";
  val = 0;
} else if (val == 6) {
  pressed = true;
  currentKey = "Bb3";
  val = 0;
} else if (val == 7) {
```

PIANO SIMULATOR
CUSTOM EVENT



```javascript
serial.on('data', (data) => {
    val = serial.readBytes()[0];
    if (val != 0 && keyIsDown == false) {
        keyIsDown = true;
        const keyEvent = new CustomEvent("keyDownEvent", {
            detail: {
                key: val,
            },
        });
        window.dispatchEvent(keyEvent);
        timeOutID = setTimeout(() => {
            keyIsDown = false;
            const keyEvent = new CustomEvent("keyUpEvent", {
                detail: {
                    key: val,
                },
            });
            window.dispatchEvent(keyEvent);
        }, 50);
    }

    if (val != 0 && keyIsDown == true) {
        clearTimeout(timeOutID);
        timeOutID = setTimeout(() => {
            keyIsDown = false;
            const keyEvent = new CustomEvent("keyUpEvent", {
                detail: {
                    key: val,
                },
            });
            window.dispatchEvent(keyEvent);
        }, 50);
    }
});
```

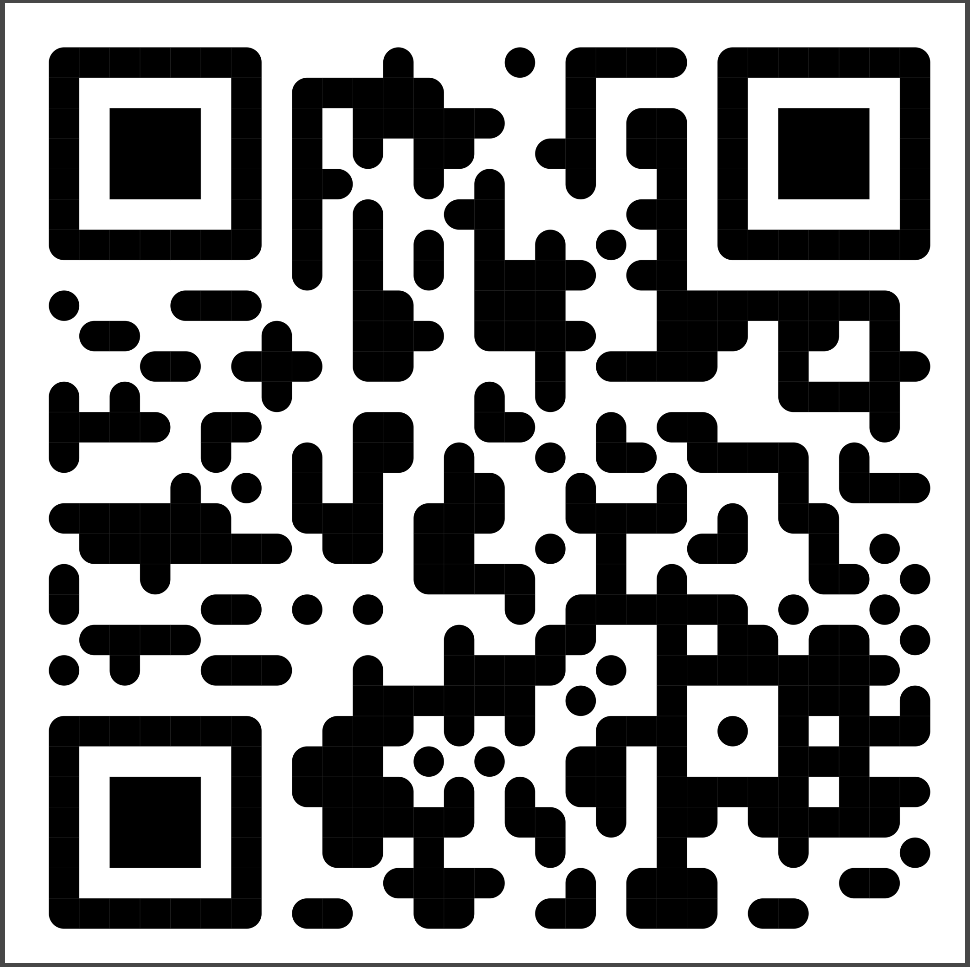


```javascript
window.addEventListener('keyDownEvent', (e) => {
    console.log("key down:", e.detail.key);
    if (!metIsStarted && practiceStarted) {
        startMet(true);
    }
    keyDownTime = new Date().getTime();
});

window.addEventListener('keyUpEvent', (e) => {
    if (!practiceStarted) {
        return
    }

    console.log("key up:", e.detail.key);
    keyUpTime = new Date().getTime();
    const diff = keyUpTime - keyDownTime;
    console.log(new Date(keyUpTime).toISOString());
    console.log(diff);
    if(diff > currentDuration.duration - contingency &&
        diff < currentDuration.duration + contingency) {
        score++;
        console.log('correct');
        progressText.innerText = 'Correct';
        currentDuration = randomDuration();
        durImg.src = currentDuration.src;
        setTimeout(() => {
            progressText.innerText = '';
        }, 500);
        if (score == 5) {
            btnNext.disabled = false;
        }
    } else {
        progressText.innerText = 'Incorrect';
        console.log('incorrect');
        setTimeout(() => {
            progressText.innerText = '';
        }, 500);
    }
});
```

PIANO SIMULATOR
VIDEO



PIANO SIMULATOR
BIG THANK YOU

🤍 I would like to thank **ALLISON BERKOY** for the guidance, feedback, support, and assistance throughout the project.

🤍 I would like to thank **ADAM WILSON** for his expertise, efforts, and helpful suggestions.

🤍. I would like to thank **JOSHUA CORN** for the advice and feedback he was happy to give me.

🤍. I would like to thank **RUDY GUERRERO** for the help with wood cutting and for letting me do some as well.

PIANO SIMULATOR

Alyona Radkevich