CS 101
Sound

Lecture 23
Sound

- Can play sound in Processing!
- What is sound?
Sound

- Can play sound in Processing!
- What is sound?
- **(Physics):** sound is a vibration that typically propagates as an audible wave of pressure, through a transmission medium such as a gas, liquid or solid (wp)
Sound

- Can play sound in Processing!
- What is sound?
- **(Physics):** sound is a vibration that typically propagates as an audible wave of pressure, through a transmission medium such as a gas, liquid or solid (wp)
- **(Perception):** The reception of such waves and their perception by the brain. Humans can hear sound waves with frequencies between about 20Hz and 20 kHz. Sound above 20 kHz is ultrasound and below 20 Hz is infrasound. (wp)
Types of Sounds

- Can either play pre-recorded audio files, or create custom sounds
- Using pre-recorded audio files is generally easier, but gives less flexibility and customization
- Analogous to drawing with images vs shapes and color
Sound library

- To play sound, must download library via processing
- To use in your program, import like so

```java
import processing.sound.*;
```

- Many possibilities with this library:
  - [https://processing.org/reference/libraries/sound/index.html](https://processing.org/reference/libraries/sound/index.html)
Play a sound file

- As with images, sound files come in many formats
Play a sound file

- As with images, sound files come in many formats
- Processing supports common ones: mp3, wav, aif
How to use

- Make sure to: `import processing.sound.*;`
- Find a supported music file, and place it into a directory named `data` in your sketch directory
- Create variable for the sound: `SoundFile music;`
- Load the sound: `music = new SoundFile(this, "music.mp3");`
- Play: `music.play();`
import processing.sound.*;
SoundFile music;

... 

music = new SoundFile(this, "music.mp3");

... 

music.play();
Play a sound file

- As with images, sound files come in many formats
- Processing supports common ones: mp3, wav, aif
- To load and play sound files, use the `SoundFile` type
- Put sound files in data directory
  - Let’s practice by:
    - Writing a program that plays some music
Play some music

- Go to the class website and download music.mp3
- Write a program that simply plays this file!
Play a sound file

- As with images, sound files come in many formats
- Processing supports common ones: mp3, wav, aif
- To load and play sound files, use the `SoundFile` type
- Put sound files in data directory
  - Let’s practice by:
    - Writing a program that plays some music
    - Adding sound to `spaceship_target_practice`
Spaceship_target_practice with sound

- Download `spaceship_target_practice` from the class site
- Download the sounds from the lecture page
- Change the program so that:
  - When a shot is taken, a sound plays
Spaceship_target_practice with sound

- Download `spaceship_target_practice` from the class site
- Download the sounds from the lecture page
- Change the program so that:
  - When a shot is taken, a sound plays
  - When the target is hit, a sound plays
Spaceship_target_practice with sound

- Download `spaceship_target_practice` from the class site
- Download the sounds from the lecture page
- Change the program so that:
  - When a shot is taken, a sound plays
  - When the target is hit, a sound plays
  - Background music
Generating custom sound

- Processing has several functions for creating custom sounds
- Specifically, the “Oscillators” functions
The Pulse oscillator

import processing.sound.*;
Pulse pulse;

void setup() {
   pulse = new Pulse(this);
   pulse.play();
}

void draw() {
}

https://processing.org/reference/libraries/sound/Pulse.html
The Pulse oscillator

```java
import processing.sound.*;
Pulse pulse;

void setup()
{
  pulse = new Pulse(this);
  pulse.play();
}

void draw()
{
}
```
The Pulse oscillator

```java
import processing.sound.*;
Pulse pulse;

void setup() {
    pulse = new Pulse(this);
    pulse.play();
    pulse.freq(100);
}

void draw() {
}
```

Change the frequency
Try other numbers!
The Pulse oscillator

import processing.sound.*;
Pulse pulse;

void setup() {
    pulse = new Pulse(this);
    pulse.play();
    pulse.freq(100);
    pulse.amp(0.5);
}

void draw() {
}

Change the amplitude (volume)
A float between 0.0 and 1.0
What sound(s) does this produce?

```java
import processing.sound.*;
Pulse pulse;

void setup() {
  pulse = new Pulse(this);
  pulse.play();
}

void draw() {
  if (frameCount % 60 > 0 && frameCount % 60 < 30) {
    pulse.freq(100);
  } else {
    pulse.freq(200);
  }
}
```
What sound(s) does this produce?

```java
import processing.sound.*;
 Pulse pulse;

void setup() {
    pulse = new Pulse(this);
    pulse.play();
    pulse.freq(50);
}

void draw() {
    float volume = frameCount / 200.0;
    pulse.amp(volume);
}
```
Halo Health

- Use the pulse oscillator to make the halo health recharge sound
  - [https://www.youtube.com/watch?v=Dluhm2g5OOU](https://www.youtube.com/watch?v=Dluhm2g5OOU)
Halo Health

```java
import processing.sound.*;

Pulse pulse;

void setup() {
    size(300, 150);
    background(100);
    frameRate(30);
    pulse = new Pulse(this);
    pulse.play();
}

int freq = 1;
float volume = 1.0;

void draw() {
    if (freq < 300) {
        pulse.amp(volume);
        pulse.freq(freq);
        freq += 3;
        volume -= 0.01;
    } else {
        pulse.stop();
    }
}
```
Shepard Tone

- Use the pulse oscillator to create a shepard tone
  - [https://www.youtube.com/watch?v=LVWTQcZbLgY](https://www.youtube.com/watch?v=LVWTQcZbLgY)
Shepard Tone

import processing.sound.*;
Pulse pulse1;
Pulse pulse2;
Pulse pulse3;

void setup() {
  size(640, 360);
  background(255);
  frameRate(10);
  pulse1 = new Pulse(this);
  pulse2 = new Pulse(this);
  pulse3 = new Pulse(this);
  pulse2.amp(0.25);
  pulse1.play();
  pulse2.play();
  pulse3.play();
}

int shepF1 = 400;
int shepF2 = 300;
int shepF3 = 200;
float shepV1 = 100;
float shepV3 = 0;
int inc = 5;

void draw() {
  shepV1 = shepV1 <= 0 ? 100 : shepV1 - inc;
  shepV3 = shepV3 >= 100 ? 0 : shepV3 + inc;
  shepF1 = shepF1 >= 500 ? 400 : shepF1 + inc;
  shepF2 = shepF2 >= 400 ? 300 : shepF2 + inc;
  shepF3 = shepF3 >= 300 ? 200 : shepF3 + inc;

  pulse1.freq(shepF1);
  pulse1.amp(shepV1/100.0 / 4.0);
  pulse3.freq(shepF3);
  pulse3.amp(shepV3/100.0 / 4.0);
  pulse2.freq(shepF2);
}
Audio input

- Can also playback and modify audio input from a microphone
- Use the `AudioDevice` and `AudioInput` types in the library
- [https://processing.org/reference/libraries/sound/](https://processing.org/reference/libraries/sound/)
Audio input

- Can also playback and modify audio input from a microphone
- Use the `AudioDevice` and `AudioInput` types in the library
- [https://processing.org/reference/libraries/sound/](https://processing.org/reference/libraries/sound/)
- Let’s practice by:
  - Playing back audio from mic
Audio input

- Can also playback and modify audio input from a microphone
- Use the `AudioDevice` and `AudioInput` types in the library
- [https://processing.org/reference/libraries/sound/](https://processing.org/reference/libraries/sound/)
- Let’s practice by:
  - Playing back audio from mic
  - Writing a program in which the user can control pan and volume of output using mouse position
Materials

• Required Materials
  ○ https://processing.org/reference/libraries/sound/index.html