Computational Music Mixing with EarSketch: Engaging Students With Python and Music

ISTE 2015: Session B318

Tuesday, June 30th

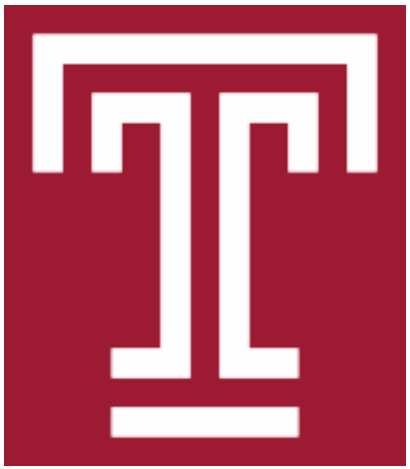
Mr. Christopher Michaud

michaudc@marist.com

www.nebomusic.net

Marist School / Georgia Institute of Technology

Who am I?



GEORGE MASON UNIVERSITY

Websites for Presentation

nebomusic.net

earsketch.gatech.edu/earsketch2

Why Music and Programming?

- *A Philosophy of Music Education:* Bennett Reimer

Music allows humans to encode, refine, reflect, and share non-verbal thought in a dynamic moving system that changes over time.

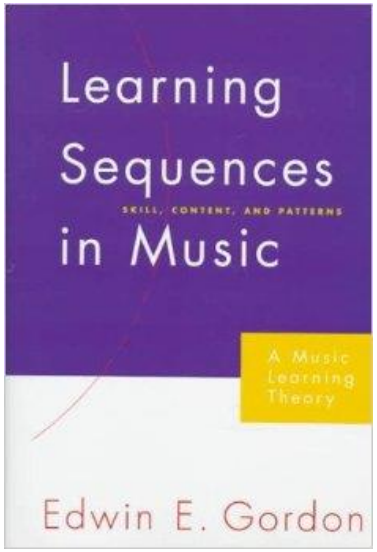
What is Programming?

- Allows humans to encode, refine, reflect, and share logical thought in a dynamic moving system that changes over time.

- Music and Programming (Computational Thinking) are the same Human Skill expressed in two different domains: non-verbal feelings in art/music and logical / motion thought in Programming or Computational Thinking

Another Book . . .

Learning Sequences in Music:
Edwin Gordon



Concept of 'Audiation' - The inner thought of music.

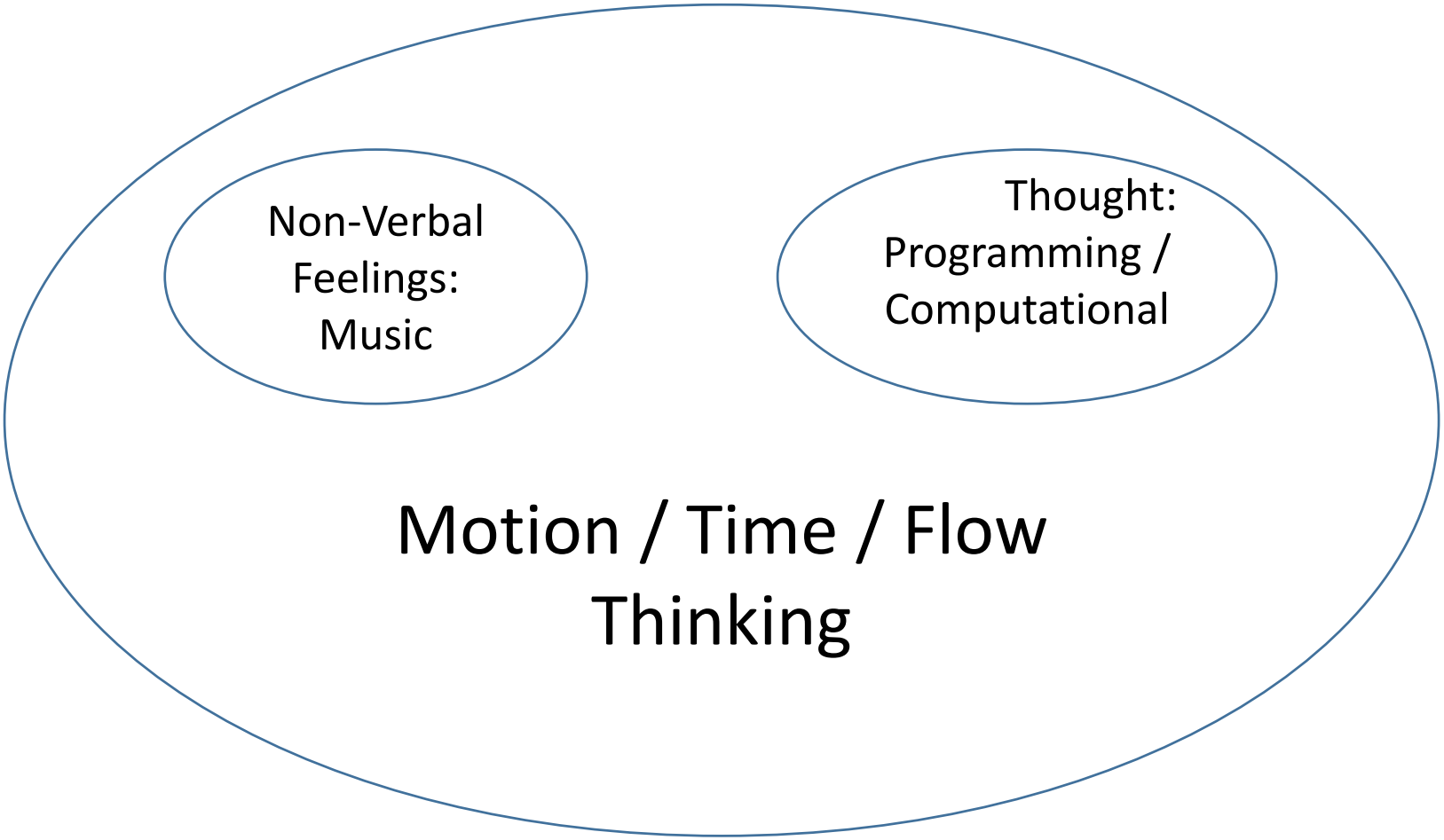
Table 2. Types of Audiation (Gordon, 2007b, p. 15).

Type 1	listening to	familiar or unfamiliar music
Type 2	reading	familiar or unfamiliar music
Type 3	writing	familiar or unfamiliar music from dictation
Type 4	recalling and performing	familiar music from memory
Type 5	recalling and writing	familiar music from memory
Type 6	creating and improvising	unfamiliar music while performing or in silence
Type 7	creating and improvising	unfamiliar music while reading
Type 8	creating and improvising	unfamiliar music while writing

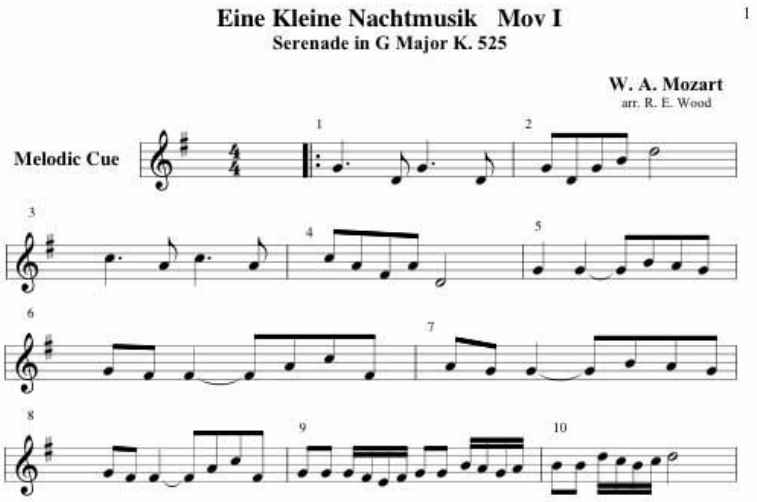
Table 3. Stages of Audiation (Gordon, 2007b, p. 20).

Stage 1	momentary retention
Stage 2	imitating and audiating tonal patterns and rhythm patterns and recognizing and identifying a tonal center and macrobeats
Stage 3	establishing objective or subjective tonality and meter
Stage 4	retaining in audiation tonal patterns and rhythm patterns that have been organized
Stage 5	recalling tonal patterns and rhythm patterns organized and audiated in other pieces of music
Stage 6	anticipating and predicting tonal patterns and rhythm patterns

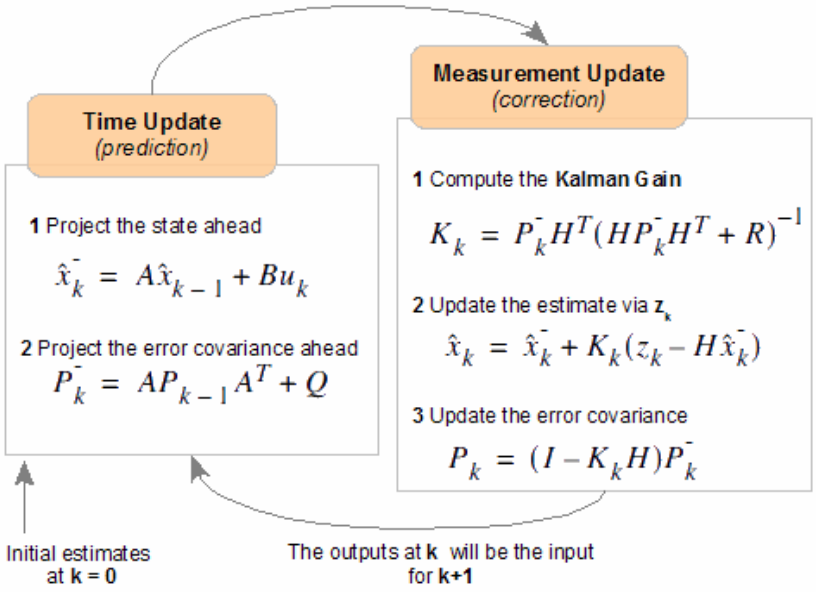
Audiation is to music as *Computational Thinking* is to Computer Science



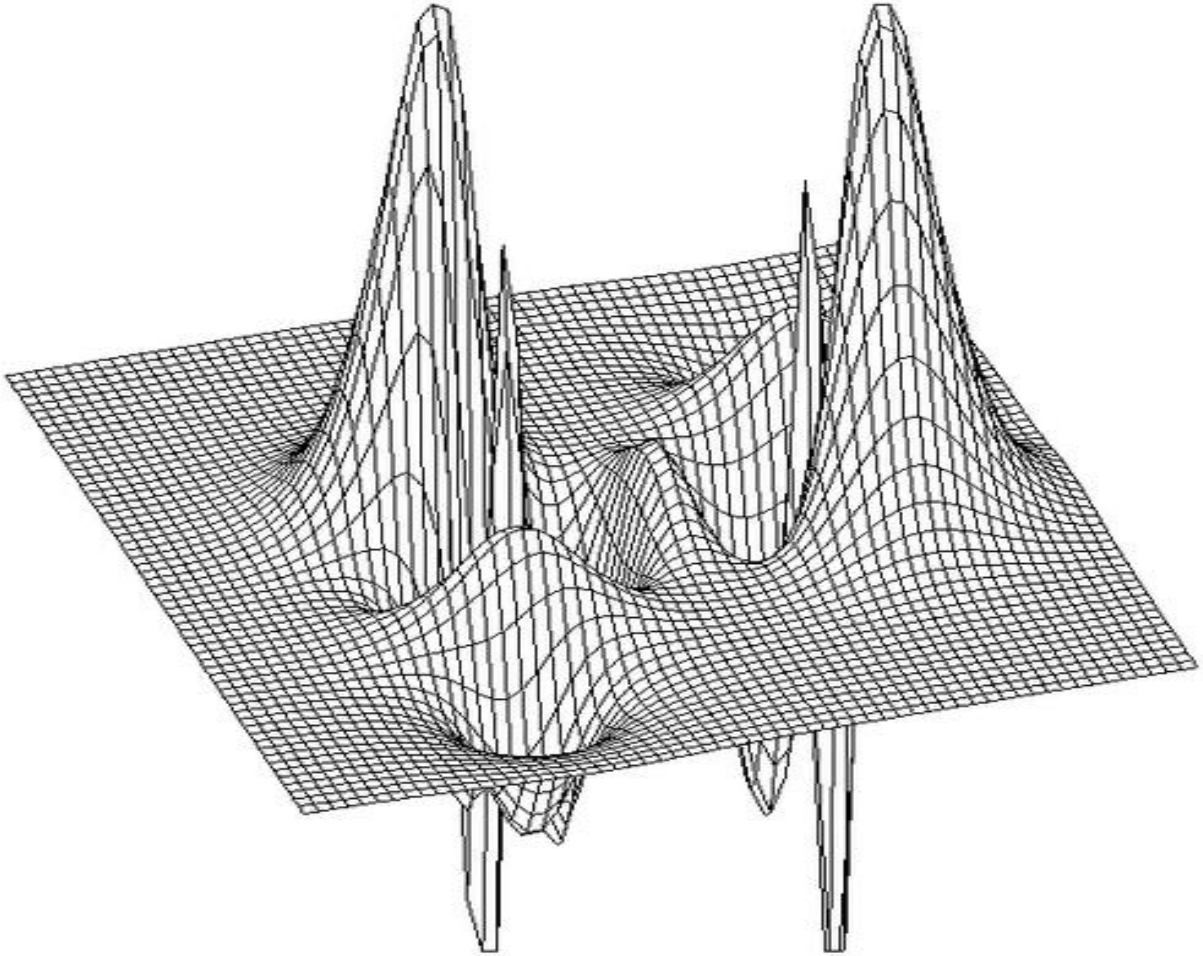
To put this another way . . .

Eine Kliene

Encoding the Non-Verbal

Kalman Filter



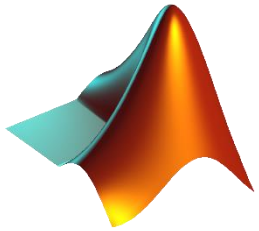
Encoding the Logic



Motivation

- Large population of students through which music is the primary vehicle for personal expression.

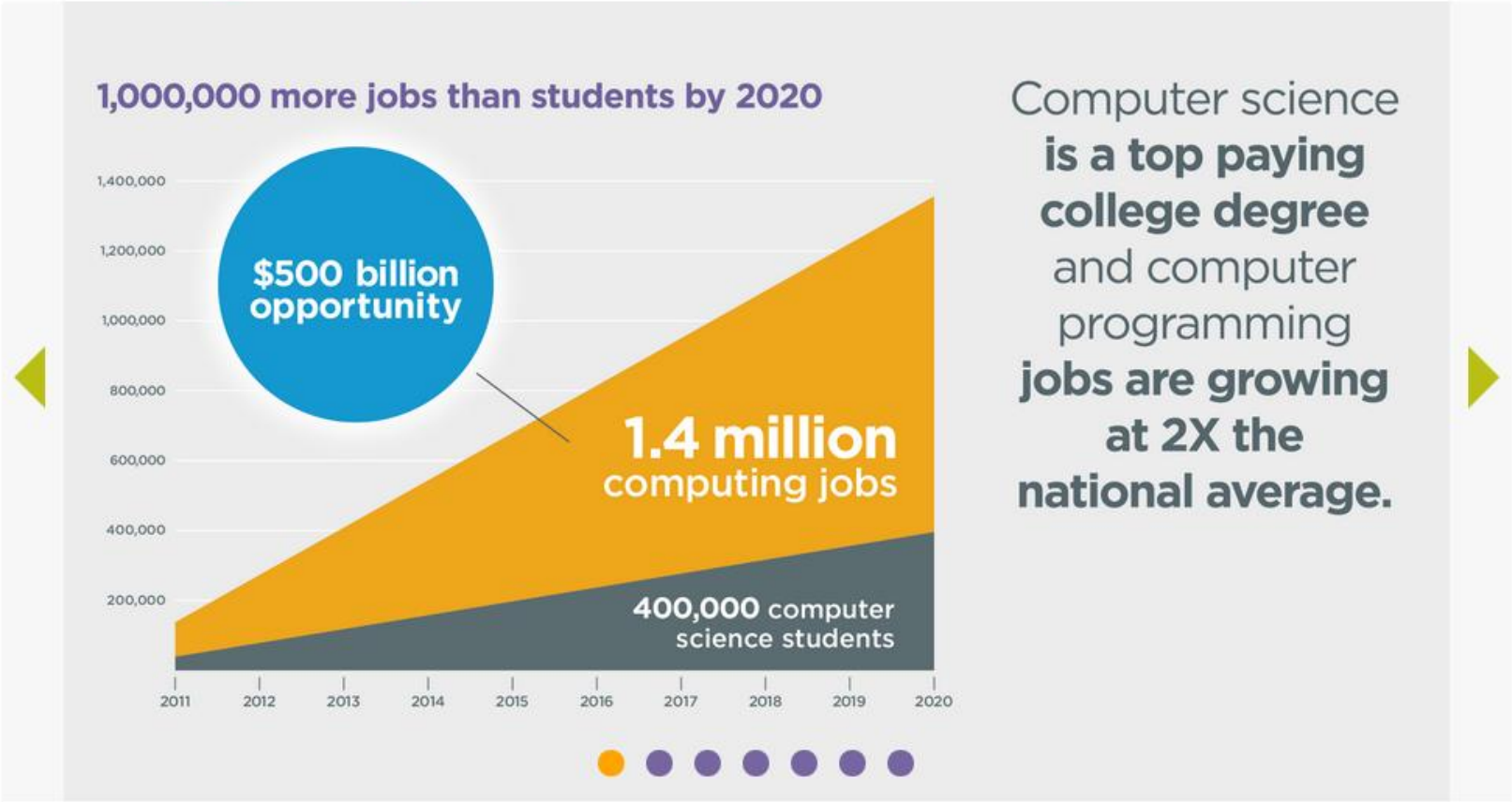
- In CS – we tend to focus on:
  - Graphics / Games / Visual
  - Text Processing / Database
  - Logic / Algorithmic / Mathematical



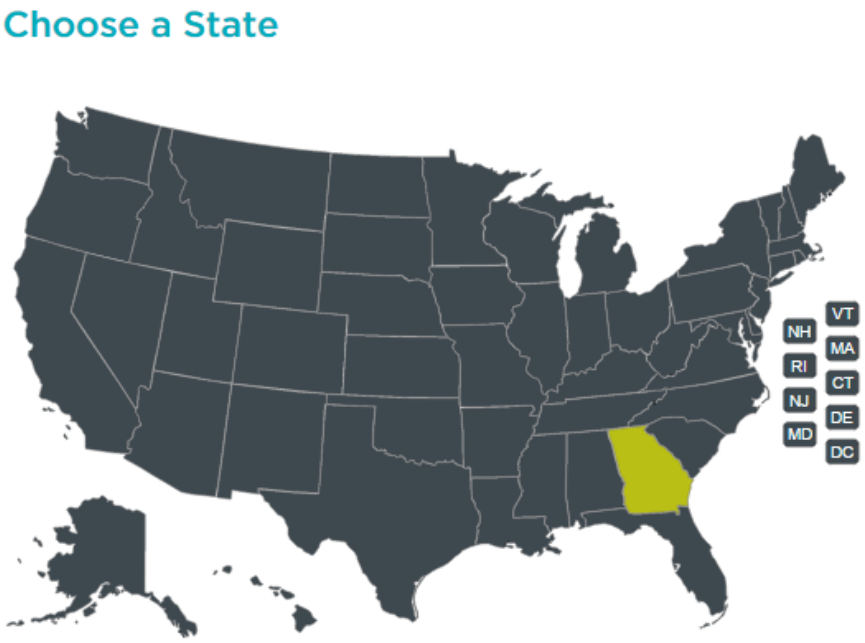
- ***EarSketch*** seeks to engage students in CS through the art of music.

Why Learn to Program?

What's wrong with this picture? Share these stats.



Why Learn to Program?



Georgia

20,371 open computing jobs
(growing at 4.8x the state average)

1,836 computer science graduates

133 schools teach computer science

- CS counts as math or science credit
- Clear certification pathways for CS teachers
- CS curriculum standards

Georgia fact-sheet | **More advocacy tools**

Georgia is one of 25 states where students **can** count computer science for credit towards high school graduation!

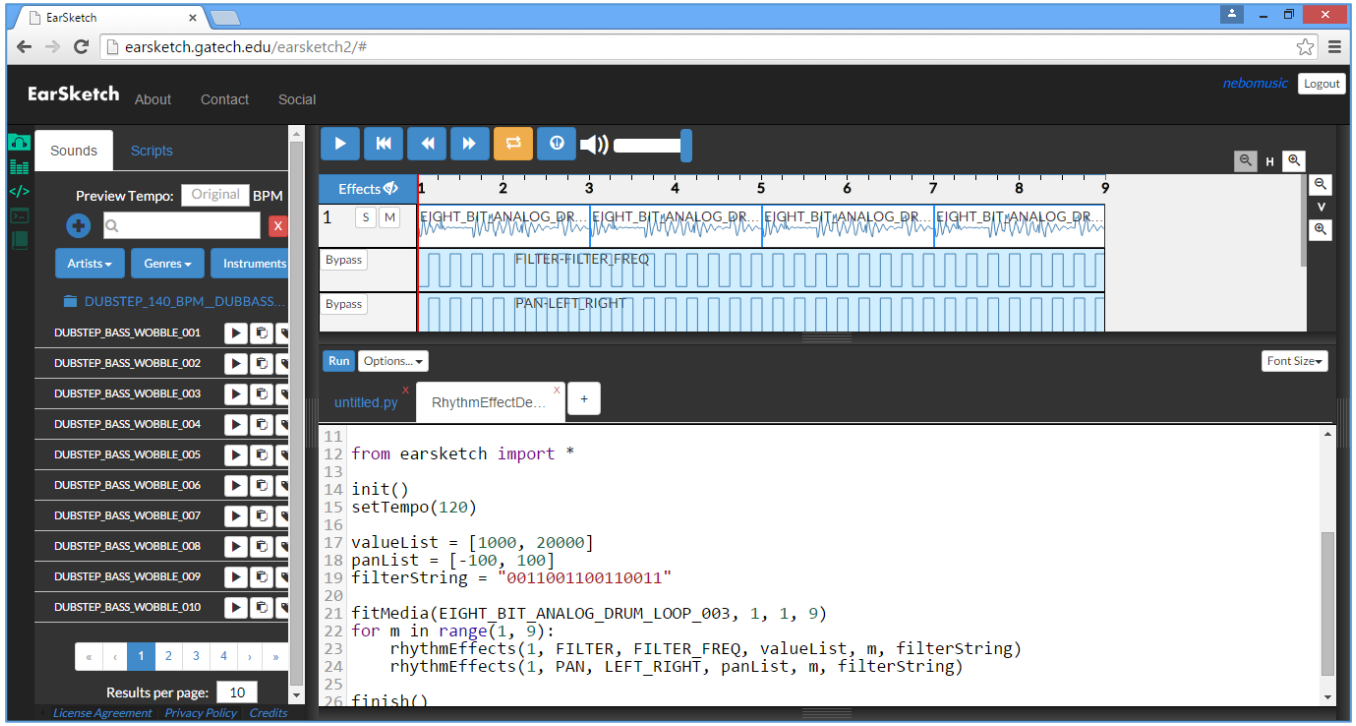
Take Action

Sources: **The Conference Board, National Science Foundation, Code.org database. More info.**

https://code.org/promote

What is EarSketch?

- Online Programming and Music Mixing Workstation



-Used at Georgia Tech to teach Digital Music Mixing.

-Tool for Programming

-Free!

-Web based

Description of EarSketch

- Programming Environment
- Python Based
- Web Based App or Installed System of Software
- API built in Python for Music Mixing
- NSF Funded project to encourage computational interest through the mixing and sharing of music.
- Curriculum and Social Media Site

Authentic

Relevant

Expressive

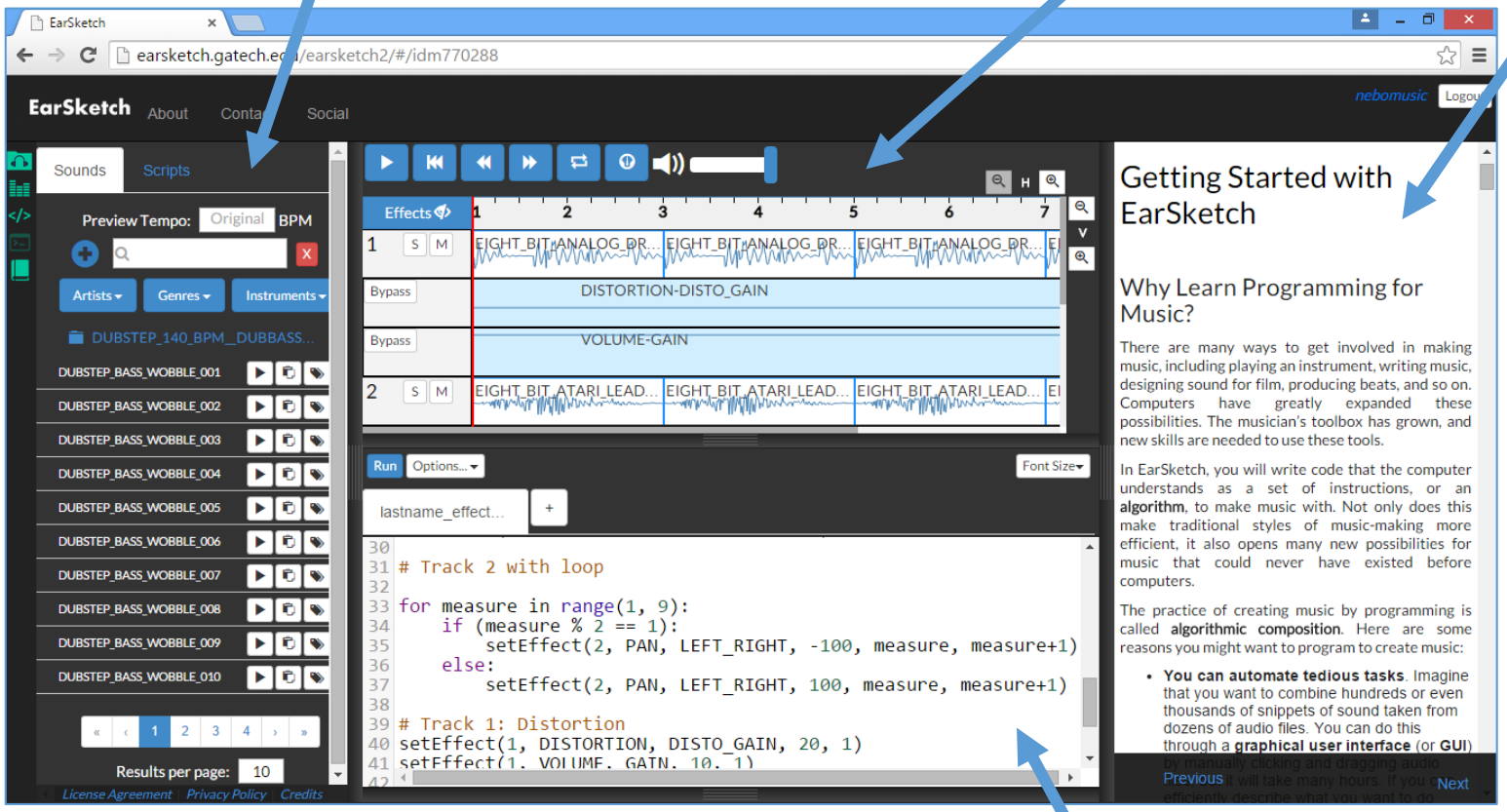
EarSketch Website: Web based IDE and DAW

Sound Library

Music View: Display and Playback

Documentation and Curriculum

Toggle Views



Coding Window: Programming

What is Python?

- General purpose Computer Programming Language
- Web development: Google and Yahoo
- Game Development, Science, Graphics
- In Business: National Weather Service, NASA, IBM, Disney, and Nokia
- Used in Development of the Google Car and other Robotic Systems

Sample EarSketch Program

sample.py +

```
1  #       python code
2  #
3  #       script_name:  sample.py
4  #
5  #       author: Christopher Michaud
6  #
7  #       description: Demo of EarSketch Sections
8  #
9  #
10 #
11
12 from earsketch import *
13
14 init()
15 setTempo(120)
16
17 # A Section - Three Tracks
18
19 fitMedia(TECHNO_LOOP_PART_003, 1, 1, 5)
20 fitMedia(TECHNO_CLUBLEAD_001, 2, 1, 5)
21 fitMedia(TECHNO_CLUBSFX_001, 3, 1, 5)
22
23
24
25 finish()
26
```

} Comments

} Setup Section

} Music Section

} Finish Section

Essential Elements we will use in Python:

- **Comments**
  ```
  # This is a comment – meant for Humans
  ```
- **Includes** – loading preset methods or data
  ```
  from earsketch import *
  ```
- **Functions** – telling the computer "what to do"
  ```
  fitMedia(drums, 1, 1, 5)
  ```
- **Variables and data types** – Names for information stored by program
  ```
  Beat1 = "0+++0+++0+0+0+++"
  ```
- **Tabs:** Enclose code in sections
- **Lists:** Groups variables into one data structure

EarSketch Python Functions

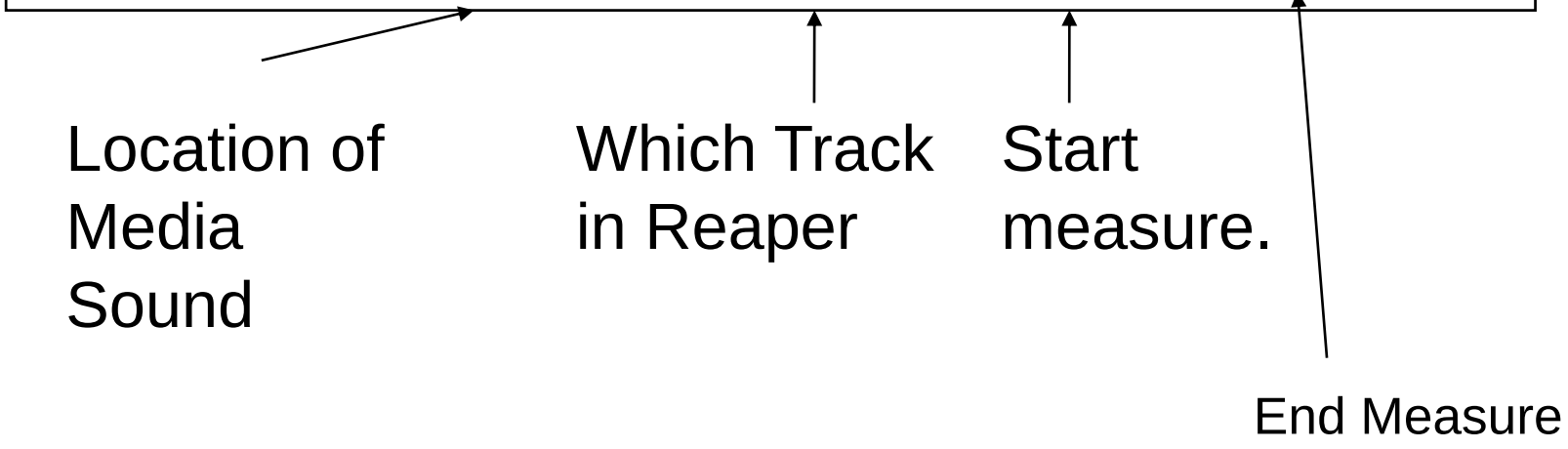
EarSketch Function	Description
`init()`	Start new Earsketch Mix and clear the DAW
`setTempo(120)`	Sets the beats per minute (speed) of the Mix
`println("Hello")`	Prints message to console

EarSketch Python Functions

Music Mixing Functions
`fitMedia(file, track, start, end)`
`makeBeat(file, track, measure, beatString)`
`setEffect(track, effect, parameter, sV, sM, eV, eM)`
`rhythmEffects(track, effect, parameter, list, measure)`

“fitMedia” Function

```
fitMedia(file, track, start, end)
```



Example:

```
fitMedia(HIP_HOP_DRUMS1_2M, 1, 1, 9)
```

Setting Volume Effects

```
setEffect(track, VOLUME, GAIN, level, start,
level2, end)
```

- Example

```
setEffect(1, VOLUME, GAIN, -40, 1, 10, 5)
```

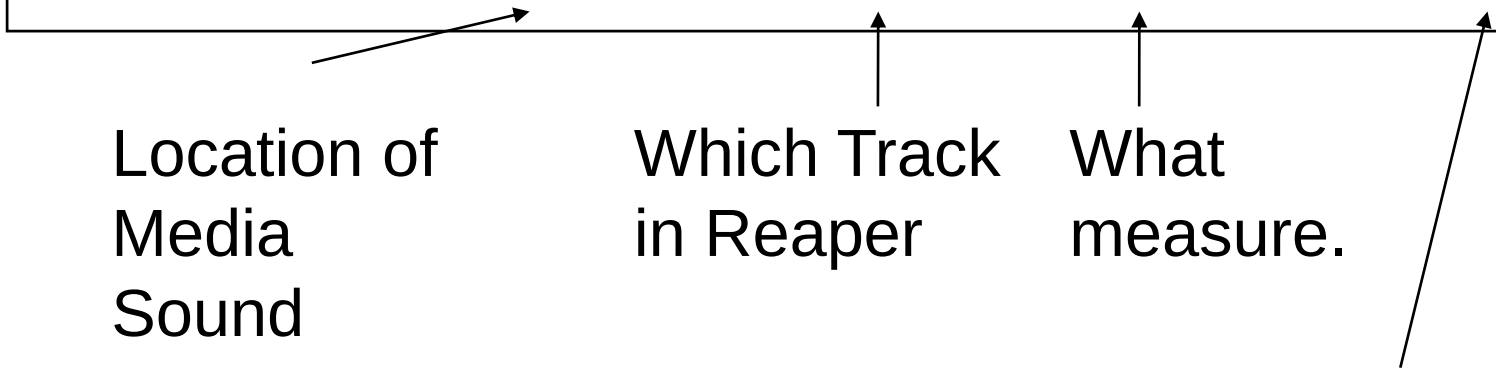
Selected List of Effects and Parameters

Effect	Parameter	Min to Max Values
VOLUME	GAIN	-60 to 12
DELAY	DELAY_TIME	0 to 300.0
CHORUS	CHORUS_LENGTH	1.0 to 15.0
CHORUS	CHORUS_NUMVOICES	1.0 to 8.0
DISTORTION	DISTO_GAIN	0.0 to 50.0
FILTER	FILTER_FREQ	20.0 to 20000.0
PAN	LEFT_RIGHT	-100 to 100 (Left to Right)

Complete Effect list at:
http://earsketch.gatech.edu/category/learning/reference/every-effect-explained

“makeBeat” Method

```
makeBeat(file, track, measure, BeatString)
```



Example:

```
makeBeat(drums, 1, 1, “0+0+0+++00-00+++”)
```

Beat String notation

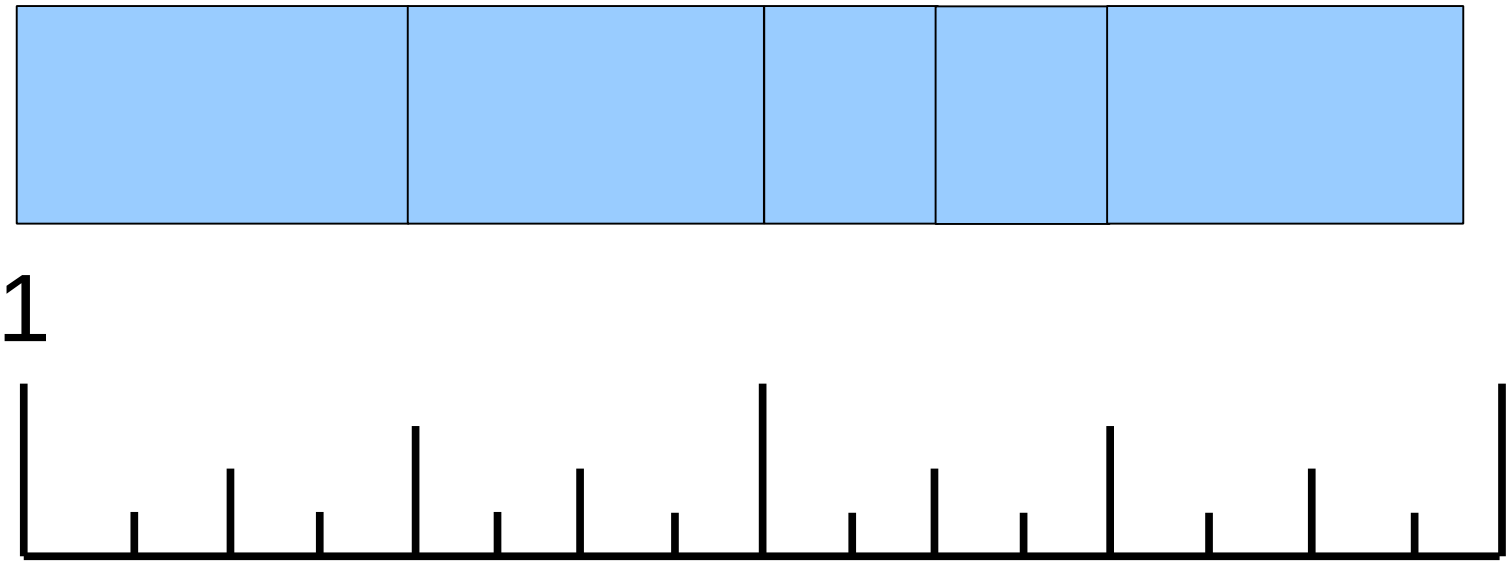
"0, 1, 2, 3 . . . " = Which Media Sound you want for the segment of beat. Correspond to placement in a List that is one based.

Note: 0 will refer to a sound if it is the only media file in the argument.

"+" Means extend or loop the Media sound $1/16^{th}$ of a measure.

"-" Means $1/16^{th}$ measure of rest.

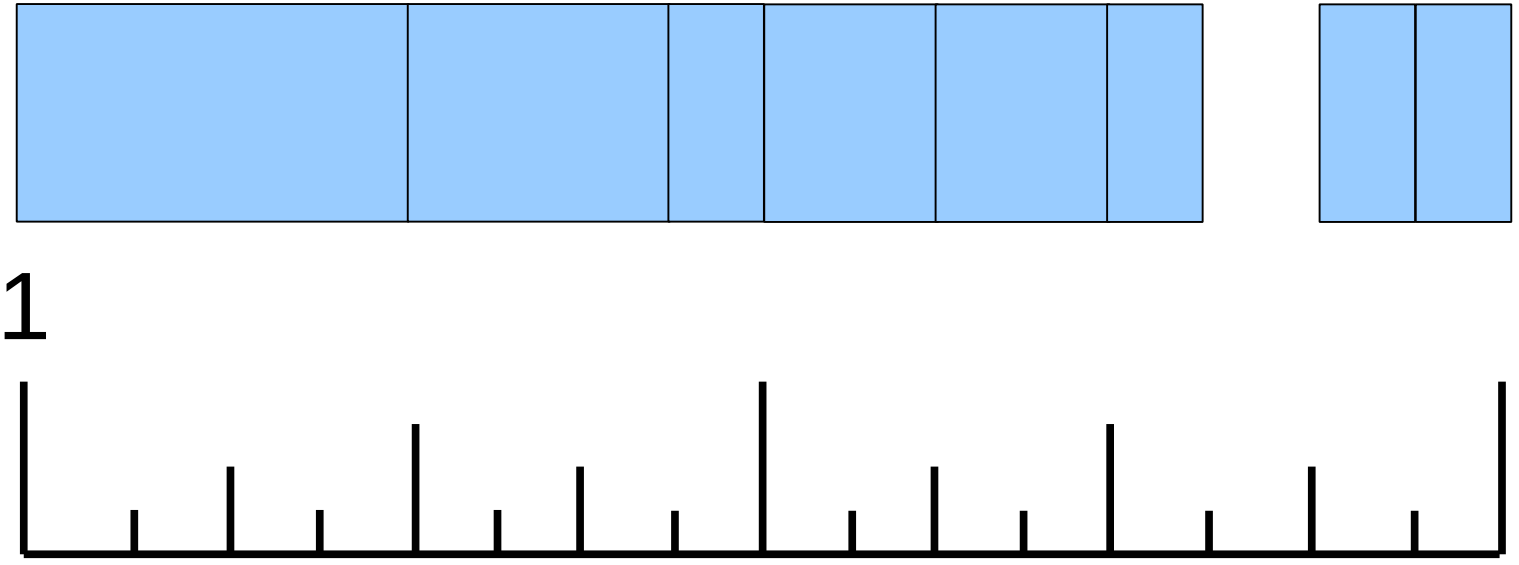
“0+++0+++0+0+0+++”



1


```
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1,
“0+++0+++0+0+0+++”)
```

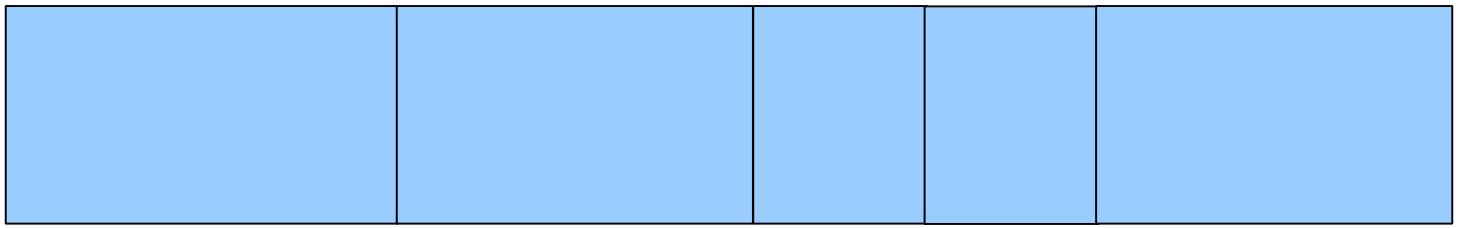
“0+++0++00+0+0-00”



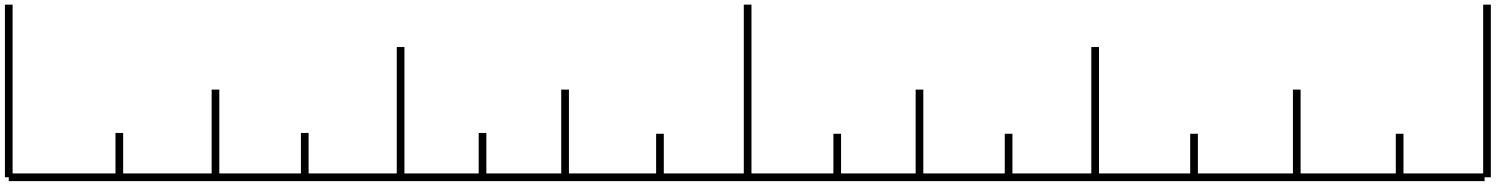
1


```
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1, “0+++0++00+0+0-00”)
```

“0+++0+++0+0+0+++”

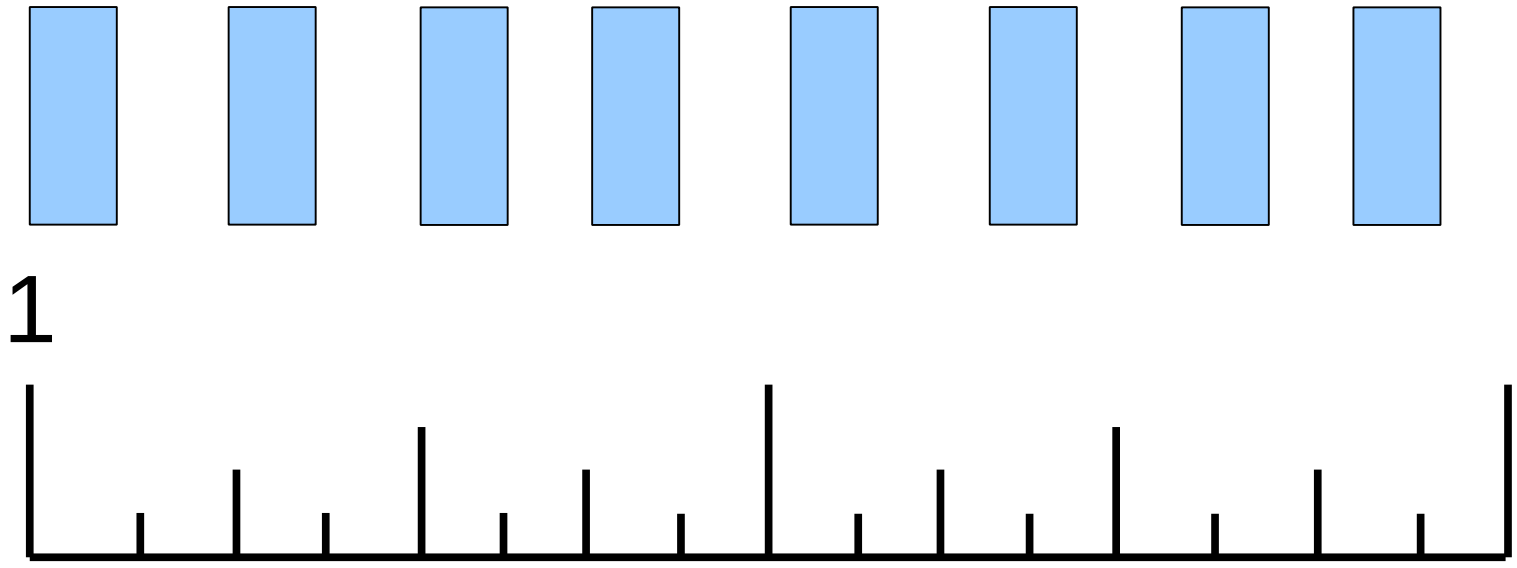


1



```
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1, “0+++0+++0+0+0+++”)
```

“0-0-0-0-0-0-0-0-”



1


```
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1, “0-0-0-0-0-0-0-0-”)
```

For Loops: Skip Counting

```
fillDrum = HIP_HOP_SYNTHDRUMS2_2M
beat = "0+++0+++0-000+00"

for measure in range(1, 9, 2):
    makeBeat(fillDrum, 1, measure, beat)
```

measure is the "index variable" = assigned values from the range()

(1, 9, 2) means start counting at 1,
end before 9 [meaning 8] and skip count by 2:

(1, 3, 5, 7)

Functions: Recycle and Reuse!

```
def sectionA(start, end):
    stompDrums = HIPHOP_STOMP_BEAT_002
    bongoDrums = HIPHOP_DUSTYPERCUSSION_002
    keys = HIPHOP_SOLOMOOGLEAD_001
    scratch = ELECTRO_SFX_WHITENOISE_SCATTER_002

    fitMedia(stompDrums, 1, start, end)
    fitMedia(bongoDrums, 2, start, end)
    fitMedia(keys, 3, start, end)

    for measure in range(start, end):
        if measure % 2 == 0:
            fitMedia(scratch, 4, measure, measure+1)
```

Now I can use this section anywhere!

```
sectionA(1, 9)
sectionA(17, 25)
```

Creating a Function

1. Definition:

```
def sectionA(start, end):
```

2. Decide on Variables for Music

```
stompDrums = HIPHOP_STOMP_BEAT_002
bongoDrums = HIPHOP_DUSTYPERCUSSION_002
keys = HIPHOP_SOLOMOOGLEAD_001
scratch = ELECTRO_SFX_WHITENOISE_SCATTER_002
```

3. Write fitMedia() calls

```
fitMedia(stompDrums, 1, start, end)
fitMedia(bongoDrums, 2, start, end)
fitMedia(keys, 3, start, end)
```

4. Write any For Loops

```
for measure in range(start, end):
    if measure % 2 == 0:
        fitMedia(scratch, 4, measure, measure+1)
```

5. Set Effects

Rhythm Effects

```
# Define List of Values for Effects
valueList = [1000, 20000]
panList = [-100, 100]

# Define BeatString for Effects
filterString = "0011001100110011"

# Music for Track
fitMedia(EIGHT_BIT_ANALOG_DRUM_LOOP_003, 1, 1, 9)

# For Loop to call Effects
for m in range(1, 9):
    rhythmEffects(1, FILTER, FILTER_FREQ, valueList, m, filterString)
    rhythmEffects(1, PAN, LEFT_RIGHT, panList, m, filterString)
```

Exercises for Classroom

- Create EarSketch Account
- Mix1: (AB Section Exercise)
  - Use fitMedia and setEffect
  - Music in sections
- Mix2: (makeBeat and For Loop Exercise)
  - makeBeat
  - For Loop Structure
- Mix3: (Defining Functions Exercise)
- Final Mix

Final Mix Project Goal

- Define Three Functions
  - sectionA(start, end)
  - sectionB(start, end)
  - sectionC(start, end)
- Each function will have at least 3 musical clips
- At least one function will use a for loop and makeBeat
- At least one function will use effects

- Call your functions to create a music mix
  - ABABCBB
  - At least 64 Measures

ISTE Standard S.1.a: Create original works as a means of personal or group expression

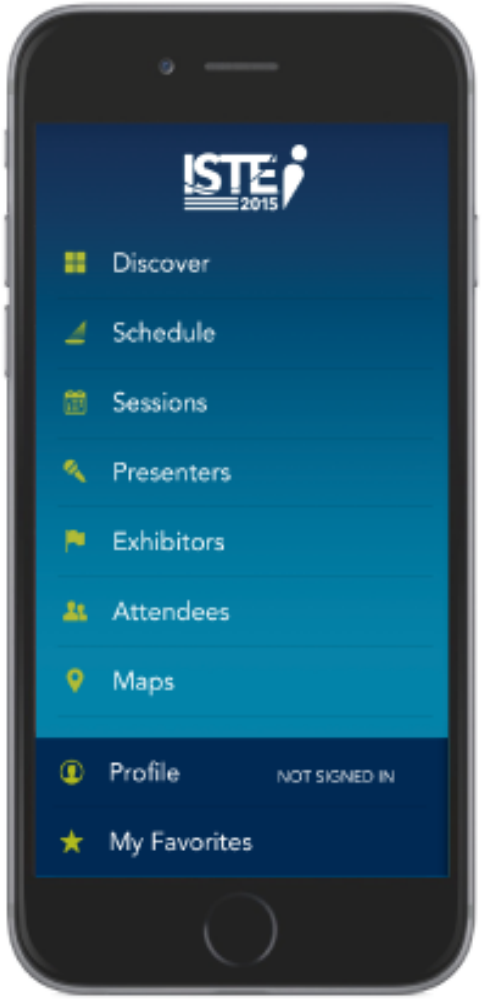
- EarSketch is a tool that equips students generate musical mixes and create functions programmatically to manipulate sound recordings. The result is an artifact (sound file and textual code) that can be shared with others or modified in group work setting.

ISTE Standard T.1.d: Model collaborative knowledge construction

- The EarSketch online tool and Social media site allows student and teachers to reflect on each other's work and collaborate and reflect on student approach and process in creating music mix projects. The EarSketch curriculum fosters remixing of computational and musical ideas through the models of pair programming.

ISTE Standard T.2

- Section a: EarSketch features a library of culturally relevant musical sounds that allow students and instructors entrance into the world of mixing without needing extensive musical training



Tell us what you think!

There are two ways to provide feedback on this session:

- ISTE 2015 mobile app
- isteconference.org/feedback

My Email:
michaudc@marist.com