Programming In Python
Music Mixing with EarSketch

Mr. Christopher Michaud
michaudc@marist.com
www.nebomusic.net
Marist School / Georgia Institute of Technology
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
What do all Human Languages Have?

• Nouns
• Verbs
• Adverbs
• Adjectives
• Clauses
• Pronouns
• Subject and Object
All Programming Languages Have:

- Data Types
- Data Structures
- Variables
- Operators
- Control Structures
- Functions
  - Call
  - Define
- Class Structures
## Data Types (Many more than on this list)

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Definition and Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>integer</td>
<td>Whole Numbers (-9, -4, 0, 1, 200, 500, ...)</td>
</tr>
<tr>
<td>float</td>
<td>Decimal Numbers (-23.45, -11.2, 0.0001, 1.25, 5.67 ...)</td>
</tr>
<tr>
<td>string</td>
<td>Characters ‘strung together’ into worlds (“Hello”, “Mr. Michaud”, “0+++0+0+0---0+++”)</td>
</tr>
<tr>
<td>boolean</td>
<td>True or False</td>
</tr>
</tbody>
</table>
Variables: Store Data

my_age = 15

bill = 17.25

music = “C:\MyMusic\song.mp3”

lives = 3

name = “Mr. Michaud

alive = True
Data Structures: Group Data together
In Python, these are called “Lists”

students = [
    “Rebecca”, “Joshua”, “Carter”,
    “Kelley”, “Jose”
]

ages = [16, 12, 13, 13, 14]

images = [“dog.png”, “bird.png”, “cat.png”]

### Operators: Perform Changes or Measurements with Data

<table>
<thead>
<tr>
<th>Operator</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Adds two values together ( 5 + 4 )</td>
</tr>
<tr>
<td>-</td>
<td>Subtracts one value from another ( 5 - 4 )</td>
</tr>
<tr>
<td>*</td>
<td>Multiplies two values together ( 5 \times 4 )</td>
</tr>
<tr>
<td>/</td>
<td>Divides one value by another ( 5 / 4 )</td>
</tr>
<tr>
<td>%</td>
<td>Modulo: Returns the remainder after division ( 5 % 4 = 1 )</td>
</tr>
<tr>
<td>=</td>
<td>Assigns a value to a variable</td>
</tr>
<tr>
<td>==</td>
<td>Compares two values</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater Than</td>
</tr>
<tr>
<td>!</td>
<td>Not</td>
</tr>
<tr>
<td>and</td>
<td>And</td>
</tr>
<tr>
<td>or</td>
<td>or</td>
</tr>
</tbody>
</table>
Control Structures: Control flow of the Program

<table>
<thead>
<tr>
<th>Type</th>
<th>Example</th>
</tr>
</thead>
</table>
| Conditional: If a statement is true | if (age == 16):  
  canDrive = True |
| Loop: Repeat Code for a number of times | for count in range(1, 10):  
  print count |
| While Loop: Repeat while a condition is true | while (lives > 3):  
  x = x + 1 |
| If Else Conditional: | if (age == 16):  
  canDrive = True  
else:  
  canDrive = False |
Functions: Group Commands together

• Define Functions

```python
def printName(first, last):
    name = first + " " + last
    print name
```

• Call Functions

```python
printName(“Rebecca”, “Michaud”)
```
Classes: Model Objects in Programming: Store Properties and Actions of Object

```
class robot:

    def __init__(self, x = 0.0, y = 0.0, heading = 0.0):
        self.x = x
        self.y = y
        self.heading = heading
        self.turning = 0
        self.distance = distance

    def move(self, turning, distance):
        # Execute motion
        self.heading += turning
        self.x += distance * cos(self.heading)
        self.y += distance * sin(self.heading)
```
Example: Login Program (Not secure!)

```python
# Sample Python Program - Login
# Mr. Michaud
# www.nebomusic.net

# Set User name
userName = "nebomusic"

# Set password - not really secure in this example!
password = "1234"

# Function to prompt user for password
def login():
    enterName = raw_input("Please enter username: ")
    enterPassword = raw_input("Password: ")
    if (enterName == userName and enterPassword == password):
        print "Welcome " + userName +"!"
    else:
        print "User name or password not correct. Goodbye"

# run program
login()
```
Example Program: Drawing with Turtle

```python
# Sample Python Program: Drawing
# Mr. Michaud
# www.nebomusic.net

# import Turtle Library
from turtle import *

# Create a window for a Turtle
window = Screen()

# Create a turtle
rebecca = Turtle()

def drawSquare(size):
    for edge in range(4):
        rebecca.forward(size)
        rebecca.left(90)

# Draw a Design
for index in range(54):
    drawSquare(100)
    rebecca.left(25)

done()
```
Example Program: Mad Libs

```python
# Mad libs
# Mr. Michaud
# www.nebomusic.net

from random import randint

# Lists of words
nouns = ["car", "cat", "dog", "peach", "sousaphone"]
verbs = ["ran", "ate", "dropped", "stepped", "flew", "walked"]

def printMadLib():
    # Pick random words from Lists
    n1 = nouns[randint(0, len(nouns)-1)]
    n2 = nouns[randint(0, len(nouns)-1)]
    v = verbs[randint(0, len(verbs)-1)]

    sentence = "The " + n1 + " " + v + " on a " + n2 + "."
    print sentence

# Print 10 sentences
for index in range(10):
    printMadLib()
```
Online Python Tutorial: Codecademy

• Go to: http://www.codecademy.com/
• Create an account and Log in
• Go to the Python Tutorial
• Goal for Week: Finish Python Language Tutorial Sections
  • Python Syntax
  • Strings and Console Output
  • Conditionals and Control Flow
  • Functions
  • Lists and Dictionaries
  • Lists and Functions
  • Loops
What is EarSketch?

• Online Programming and Music Mixing Workstation

http://earsketch.gatech.edu/earsketch2/

- 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
EarSketch Website: Web based IDE and DAW

- Sound Library
- Music View: Display and Playback
- Documentation and Curriculum

Toggle Views

Coding Window: Programming

Getting Started with EarSketch

Why Learn Programming for Music?

There are many ways to get involved in making music, including playing an instrument, writing music, designing sound for film, producing beats, and so on. Computers have greatly expanded these possibilities. The musician’s toolbox has grown, and new skills are needed to use these tools.

In EarSketch, you will write code that the computer understands as a set of instructions, or an algorithm, to make music with. Not only does this make traditional styles of music-making more efficient, it also opens many new possibilities for music that could never have existed before computers.

The practice of creating music by programming is called algorithmic composition. Here are some reasons you might want to program to create music:

- You can automate tedious tasks. Imagine that you want to combine hundreds or even thousands of snippets of sound taken from dozens of audio files. You can do this through a graphical user interface (GUI).

License Agreement | Privacy Policy | Credits

Previous | Next
Sample EarSketch Program

```python
# python code
# script_name: sample.py
# author: Christopher Michaud
# description: Demo of EarSketch Sections

from earsketch import *

init()
setTempo(120)

# A Section - Three Tracks
fitMedia(TECHNO_LOOP_PART_003, 1, 1, 5)
fitMedia(TECHNO_CLUBLEAD_001, 2, 1, 5)
fitMedia(TECHNO_CLUBSFX_001, 3, 1, 5)

finish()
```

{ 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
How is Music Organized? How do we encode musical clips?

**Layers:** Music exists through time

**Tracks:** Vertical Layers of Music
# EarSketch Python Functions

<table>
<thead>
<tr>
<th>EarSketch Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>init()</code></td>
<td>Start new EarSketch Mix and clear the DAW</td>
</tr>
<tr>
<td><code>setTempo(120)</code></td>
<td>Sets the beats per minute (speed) of the Mix</td>
</tr>
<tr>
<td><code>println(&quot;Hello&quot;)</code></td>
<td>Prints message to console</td>
</tr>
</tbody>
</table>
# EarSketch Python Functions

## Music Mixing Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Signature</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>fitMedia</code></td>
<td><code>fitMedia(file, track, start, end)</code></td>
</tr>
<tr>
<td><code>makeBeat</code></td>
<td><code>makeBeat(file, track, measure, beatString)</code></td>
</tr>
<tr>
<td><code>setEffect</code></td>
<td><code>setEffect(track, effect, parameter, sV, sM, eV, eM)</code></td>
</tr>
<tr>
<td><code>rhythmEffects</code></td>
<td><code>rhythmEffects(track, effect, parameter, list, measure)</code></td>
</tr>
</tbody>
</table>
“fitMedia” Function

\[
\text{fitMedia}(\text{file}, \text{track}, \text{start}, \text{end})
\]

- Location of Media Sound
- Which Track in Reaper
- Start measure.
- End Measure

Example:

\[
\text{fitMedia(\text{HIP\_HOP\_DRUMS1\_2M}, 1, 1, 9)}
\]
Setting Volume Effects

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

- Example

```plaintext
setEffect(1, VOLUME, GAIN, -40, 1, 10, 5)
```
## Selected List of Effects and Parameters

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

Complete Effect list at:  
[http://earsketch.gatech.edu/category/learning/reference/every-effect-explained](http://earsketch.gatech.edu/category/learning/reference/every-effect-explained)
“makeBeat” Method

makeBeat(file, track, measure, BeatString)

- Location of Media Sound
- Which Track in Reaper
- What measure.

Example: “0+++0+++0+0+0+++”

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 1/16th of a measure.

“-” Means 1/16th measure of rest.
```
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1,
"0+++0+++0+0+0+++"")
```
"0+++0++00+0+0−00"

makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1, "0+++0++00+0+0−00")
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1, "0+++0+++0+0+0+++")
```
makeBeat(ELEKTRO_HOUSE_DRUMS3_2M, 1, 1, "0-0-0-0-0-0-0-0-0-"
```
For Loops: Skip Counting

```python
code here
```

(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!

```python
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:

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

2. Decide on Variables for Music

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

3. Write fitMedia() calls

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

4. Write any For Loops

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