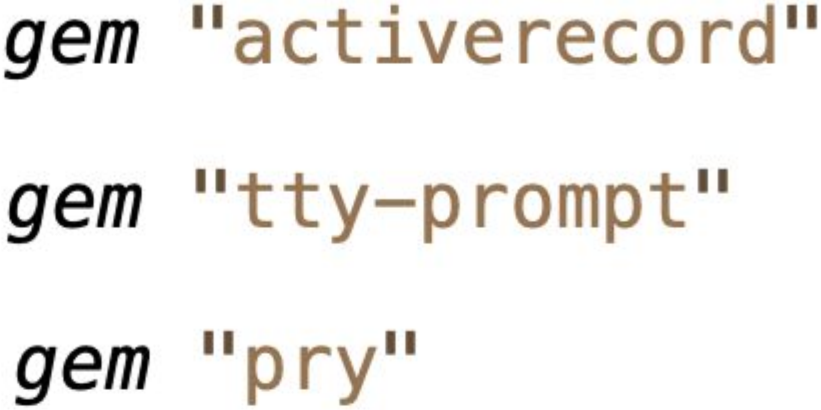
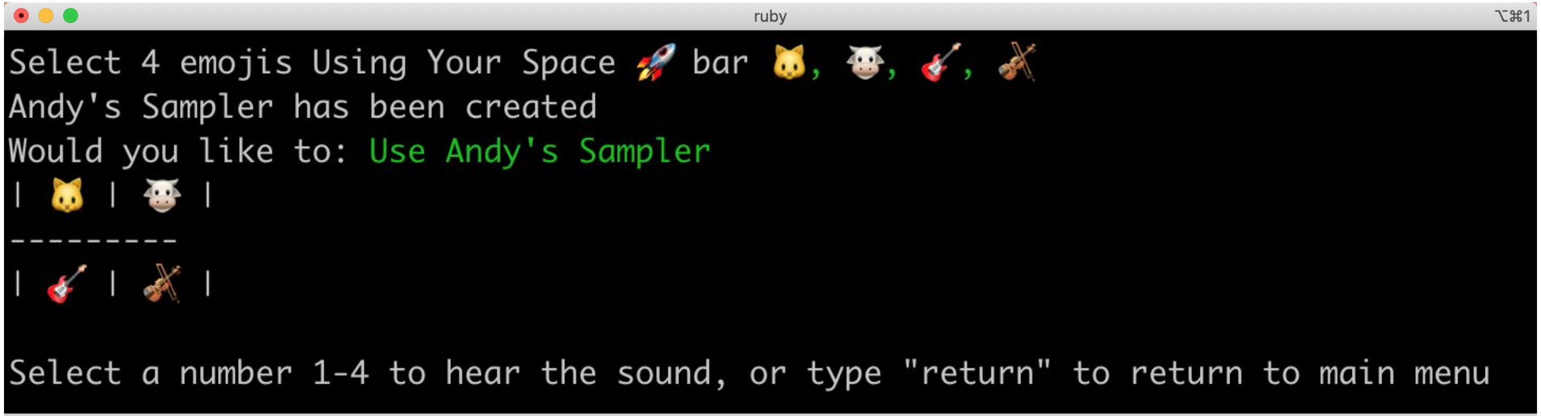
SamplerSounds

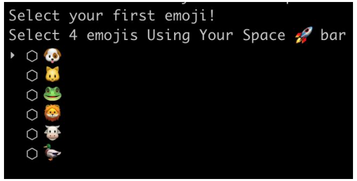


```
gem "activerecord"

gem "tty-prompt"

gem "pry"
```



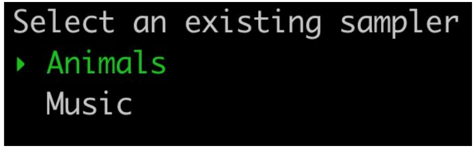
```
Select 4 emojis Using Your Space 🚀 bar 🐱, 🐮, 🎸, 🎻
Andy's Sampler has been created
Would you like to: Use Andy's Sampler
| 🐱 | 🐮 |
---------
| 🎸 | 🎻 |

Select a number 1-4 to hear the sound, or type "return" to return to main menu
```

1 😺 *"meow"*



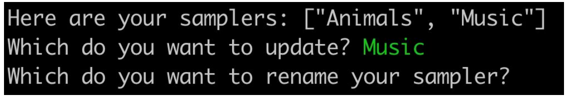
```
Select your first emoji!
Select 4 emojis Using Your Space 🚀 bar
‣ ⬡ 🐶
  ⬡ 🐱
  ⬡ 🐸
  ⬡ 🦁
  ⬡ 🐮
  ⬡ 🦆
```

Create a sampler



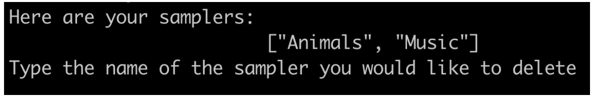
```
Select an existing sampler
‣ Animals
  Music
```

Read/Select an existing sampler



```
Here are your samplers: ["Animals", "Music"]
Which do you want to update? Music
Which do you want to rename your sampler?
```

Update a sampler



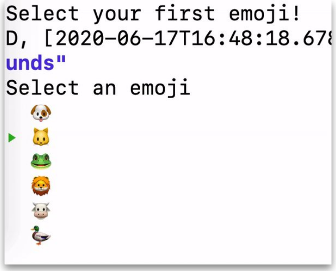
```
Here are your samplers:
               ["Animals", "Music"]
Type the name of the sampler you would like to delete
```

Delete a sampler

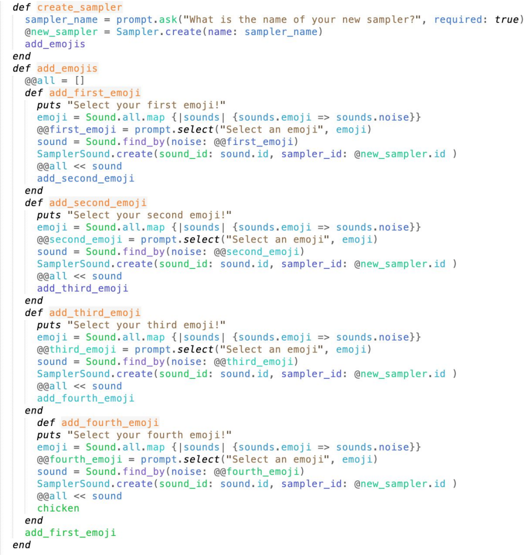
👯 **Group project != pair programming**

💡 **Ask for help when you need it**

📦 **Start small, think big**

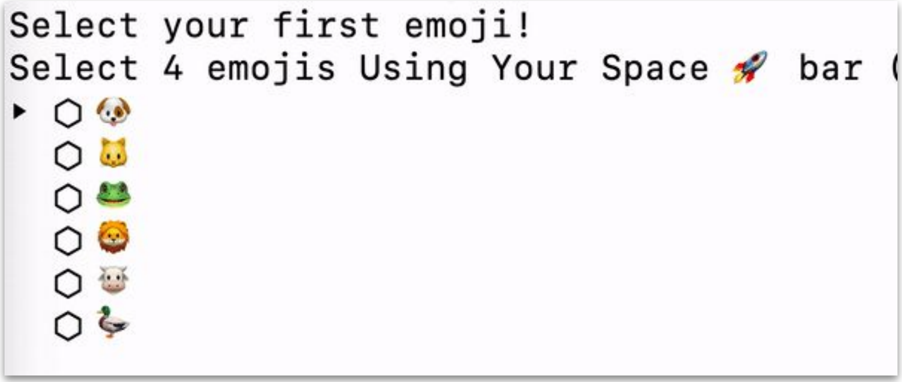


```
Select your first emoji!
D, [2020-06-17T16:48:18.678
unds"
Select an emoji
   🐶
 ▶ 🐱
   🐸
   🦁
   🐮
   🦆
```

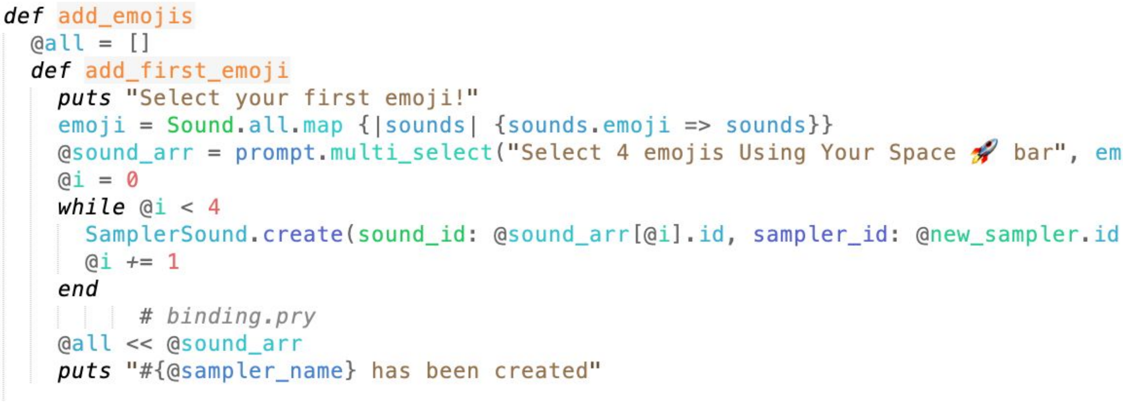


```ruby
def create_sampler
  sampler_name = prompt.ask("What is the name of your new sampler?", required: true)
  @new_sampler = Sampler.create(name: sampler_name)
  add_emojis
end
def add_emojis
  @@all = []
  def add_first_emoji
    puts "Select your first emoji!"
    emoji = Sound.all.map {|sounds| {sounds.emoji => sounds.noise}}
    @@first_emoji = prompt.select("Select an emoji", emoji)
    sound = Sound.find_by(noise: @@first_emoji)
    SamplerSound.create(sound_id: sound.id, sampler_id: @new_sampler.id )
    @@all << sound
    add_second_emoji
  end
  def add_second_emoji
    puts "Select your second emoji!"
    emoji = Sound.all.map {|sounds| {sounds.emoji => sounds.noise}}
    @@second_emoji = prompt.select("Select an emoji", emoji)
    sound = Sound.find_by(noise: @@second_emoji)
    SamplerSound.create(sound_id: sound.id, sampler_id: @new_sampler.id )
    @@all << sound
    add_third_emoji
  end
  def add_third_emoji
    puts "Select your third emoji!"
    emoji = Sound.all.map {|sounds| {sounds.emoji => sounds.noise}}
    @@third_emoji = prompt.select("Select an emoji", emoji)
    sound = Sound.find_by(noise: @@third_emoji)
    SamplerSound.create(sound_id: sound.id, sampler_id: @new_sampler.id )
    @@all << sound
    add_fourth_emoji
  end
    def add_fourth_emoji
    puts "Select your fourth emoji!"
    emoji = Sound.all.map {|sounds| {sounds.emoji => sounds.noise}}
    @@fourth_emoji = prompt.select("Select an emoji", emoji)
    sound = Sound.find_by(noise: @@fourth_emoji)
    SamplerSound.create(sound_id: sound.id, sampler_id: @new_sampler.id )
    @@all << sound
    chicken
  end
  add_first_emoji
end
```

Before



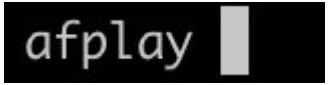
```
Select your first emoji!
Select 4 emojis Using Your Space 🚀 bar (
▶ ⬡ 🐶
  ⬡ 🐱
  ⬡ 🐸
  ⬡ 🦁
  ⬡ 🐮
  ⬡ 🦆
```



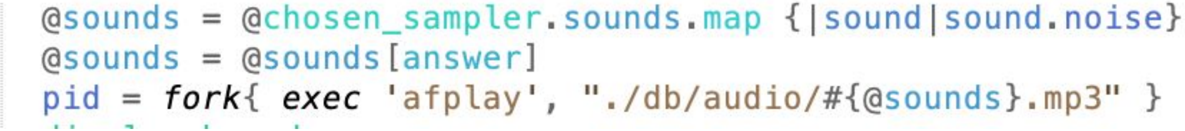
```ruby
def add_emojis
  @all = []
  def add_first_emoji
    puts "Select your first emoji!"
    emoji = Sound.all.map {|sounds| {sounds.emoji => sounds}}
    @sound_arr = prompt.multi_select("Select 4 emojis Using Your Space 🚀 bar", em
    @i = 0
    while @i < 4
      SamplerSound.create(sound_id: @sound_arr[@i].id, sampler_id: @new_sampler.id
      @i += 1
    end
          # binding.pry
    @all << @sound_arr
    puts "#{@sampler_name} has been created"
```

After

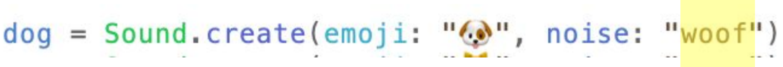
Implementing sound



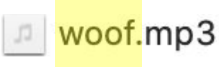
```
afplay
```



```
@sounds = @chosen_sampler.sounds.map {|sound|sound.noise}
@sounds = @sounds[answer]
pid = fork{ exec 'afplay', "./db/audio/#{@sounds}.mp3" }
```



```
dog = Sound.create(emoji: "🐶", noise: "woof")
```



woof.mp3

Fork me on GitHub

https://github.com/travisprol0/Mod-1-Final-Project

[redacted]

In the future…✨

1️⃣ More Sounds

2️⃣ Prettier sampler

3️⃣ Beat maker (that loops over time)

4️⃣ Categorize emojis for selection