T1A3 - Terminal Application Assignment

Created by Karla Tolentino

“Barber’s Paradise”

What is “Barber’s Paradise?”

This app was designed for hairdressing clients with barbers/hairdressers in mind!

Going to see a new barber? Or is this your first time? Do you feel pressured or awkward when the hairdresser asks you what haircut you want and you can’t seem to remember those one or two different styles you like on the spot?

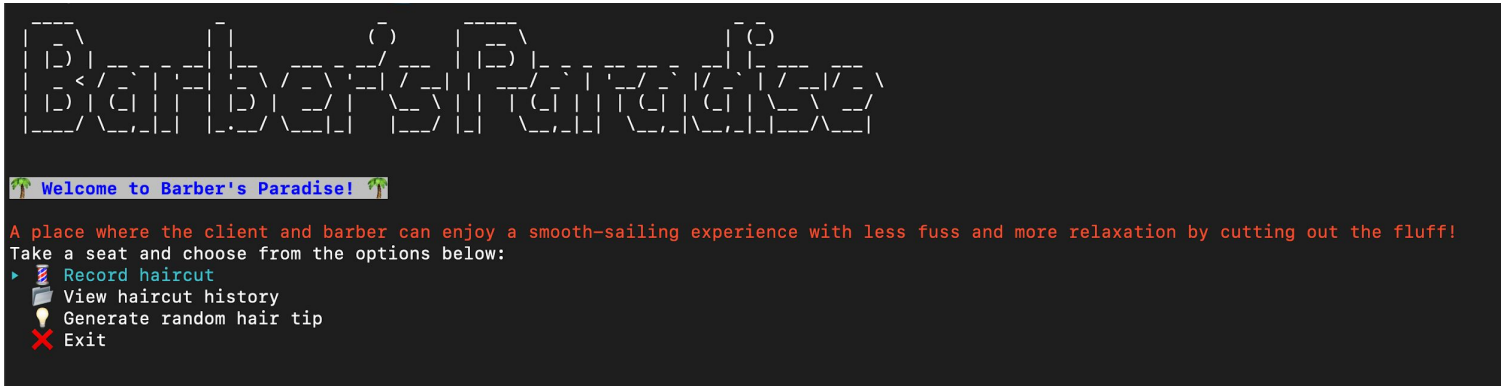
No worries, mate! Take a seat, crack open a beverage and relax as we take you to Barber’s Paradise! The place where you cut out the fluff in your haircut appointments and take note of all of your haircut experiences!

Barber’s Paradise: Main Features

1. Menu option
2. Record haircut
3. View haircut history
4. Generate random hair tip

Main Features: 1. Menu Option

- Operates on user selection via on-screen menu on app
- Option selection takes you to that feature



Main Features: 1. Menu Option

- Uses ‘tty-prompt’ gem
- Control flow structure used to implement menu option in code.

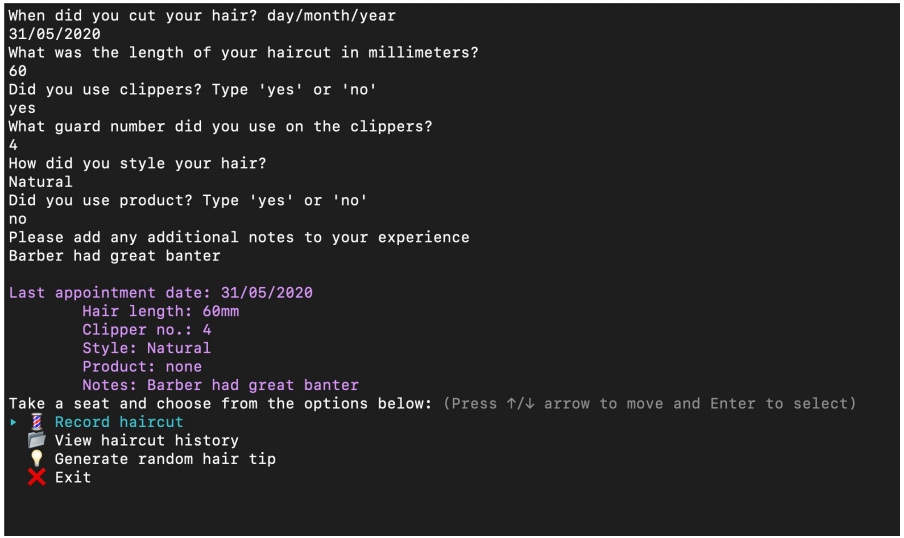
```ruby
while true
    choice = prompt.select(
        "Take a seat and choose from the options below:",
        [
            "💈 Record haircut",
            "📂 View haircut history",
            "💡 Generate random hair tip",
            "❌ Exit"
        ])
```

```ruby
elsif choice == "📂 View haircut history"
    get_history
elsif choice == "💡 Generate random hair tip"
    random_tip
elsif choice == "❌ Exit"
    print "\n"
    puts "See you at your next appointment!".colorize(:white).on_light_blue.bold
    exit
else
    puts "Invalid choice"
end
nd
```

Main Features: 2. Record Haircut

- Takes user input
- Prompts user for information on hair length, date of haircut, clipper guard used, product/styling procedures and additional notes on the experience
- Outputs data in formatted description to user
- Stores input in separate file for later use

Main Features: 2. Record Haircut



Main Features: 2. Record Haircut

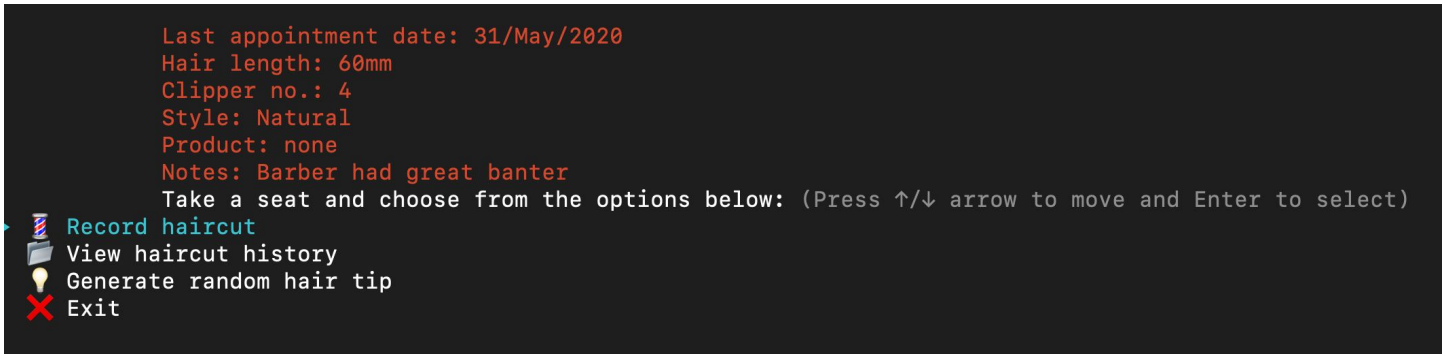
- Uses prompts and case statements to handle user input
- Pushes user input into data_hash array in JSON file for later use

```ruby
if choice == "2 Record haircut"
    prompt1 = "> "
    puts "When did you cut your hair? day/month/year"
    date = gets.chomp.to_s
    puts "What was the length of your haircut in millimeters?"
    length = gets.to_i
    puts "Did you use clippers? Type \'yes' or 'no\'"
    while clipper = gets.chomp.to_s
        case clipper
        when 'yes'
            puts "What guard number did you use on the clippers?"
            clipper = gets.chomp.to_i
            break
        when 'no'
            clipper = "none"
            break
        else
            puts "Invalid input. Please enter \'yes' or 'no\'."
            print prompt1
        end
```

```ruby
data_hash['history'] << "
    Last appointment date: #{date}
    Hair length: #{length}mm
    Clipper no.: #{clipper}
    Style: #{style}
    Product: #{product}
    Notes: #{notes}
    "
File.write('hair.json', JSON.dump(data_hash))
```

Main Features: 3. View Haircut History

- Pulls recorded user input from JSON data file
- Displays hair record history in order of input



Main Features: 3. View Haircut History

- Pulls user data from previous input from the JSON data_hash file
- If data is empty, it will input message

```
if data_hash['history'].empty?
        puts "No history recorded. Please choose another option.".colorize(:light_blue).bold
        print "\n"
    else
        print data_hash['history'].colorize(:red)
    end
```

Main Feature: 4. Generate Random Hair Tip

- Outputs a random hair tip to display to user
- Can always rely on some handy tips from your hairdresser or barber!

Main Feature: 4. Generate Random Hair Tip

- Requires ‘json’ dependency
- Pulls random hair tips at random using .sample function from JSON data_hash file

```
print data_hash['advice'].sample.colorize(:red).bold
```

Review & Challenges

- Personal circumstances affecting my learning
- Mental struggles
- Underestimated the time needed for each checklist to be completed
- Getting my head wrapped around RSPEC
- Error handling - need to revise more

More features or sprinkles I would have liked to add..

- Hair quiz that generates an appropriate hairstyle based on the answers
- A gem that displays an image result in the hair quiz!
- Added more scripts
- Added more gems to glamify and customise “barbershop” vibe to app

The End