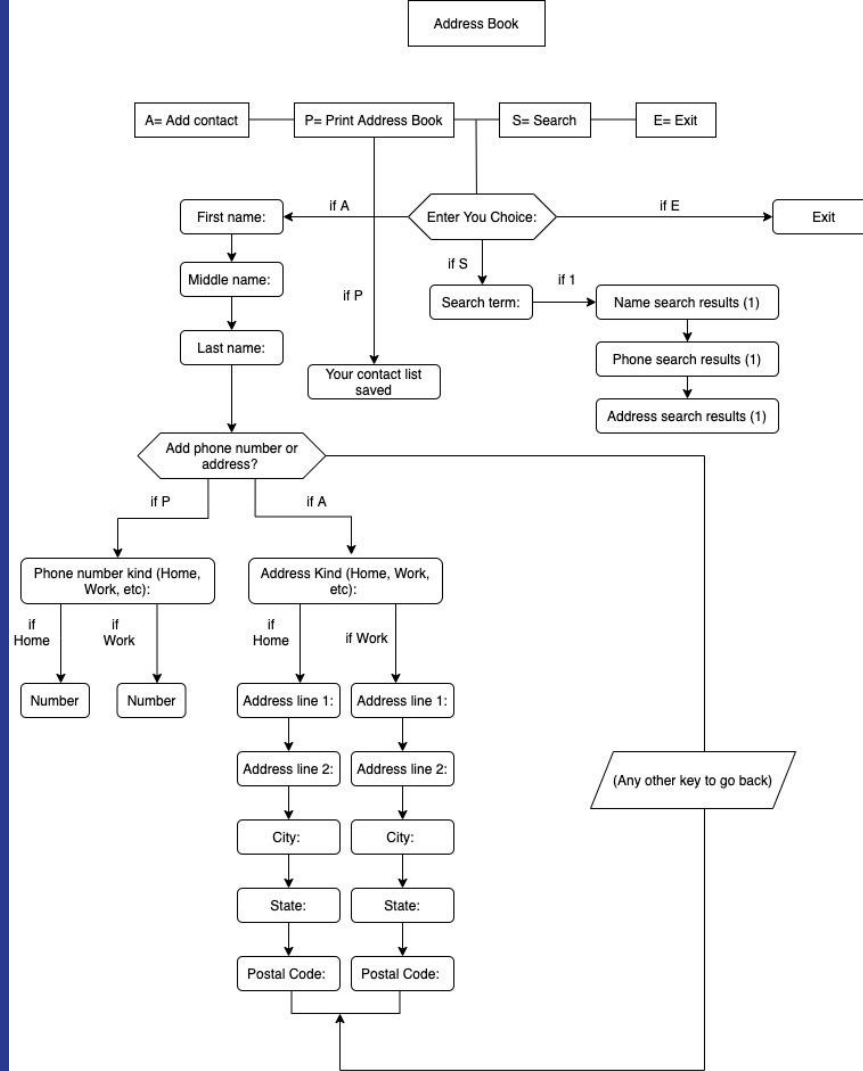
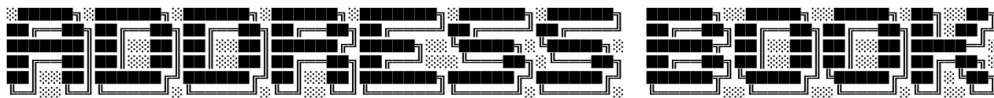


Welcome to my Address book Terminal app

By Shabnam Louie

Flow chart





Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: a

First name: Shabnam

Middle name:

Last name: Louie

Add phone number or address?

p: Add phone number

a: Add address

(Any other key to go back)

a

Address Kind (Home, Work, etc): home

Address line 1: 30

Address line 2: 2 sarah street

City: sydney

State: nsw

Postal Code: 2000

Add phone number or address?

p: Add phone number

a: Add address

(Any other key to go back)

p

Phone number kind (Home, Work, etc): work

Number: 0456789432

Add phone number or address?

p: Add phone number

a: Add address

(Any other key to go back)

o

Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: p

Your contact list saved

Louie, Shabnam .

Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: s

Search term: s

Name search results (s)

Shabnam Louie

Phone Numbers

work: 0456789432

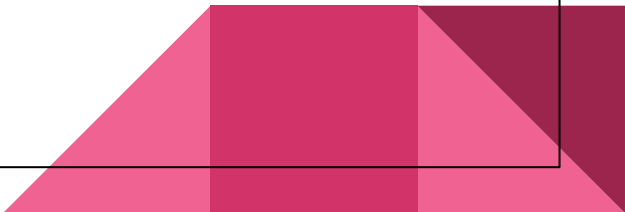
Addresses

Main features:

- Adding contacts to address book list
- Printing (save all of information when user typed)
- Searching (find a contact name or letter or address from address book list according to previous information that user typed)
- Exit



I built an Address Book terminal app with Ruby in the command-line program to let the user put in the names and information of all of their contacts. For this app I used:

- variables
 - classes
 - loops
 - control structures (statements)
 - definition
 - error handling
 - input and output
 - using symbols
 - try to not repeat my codes (DRY)
 - using Ruby Gem (Colorizer)
- 

Contact file:

- Create an attribute writers.
- Add methods and variables to read `first_name`, `middle_name`, `last_name`.
- Made another method called `full_name` to show my full name together.
- Make string method of full name
- Use a case statement to see what the format is.
- By default, return first and last name and did not include middle name.



EXPLORER

ADDRESSBOOK

docs

ppt

spec

src

image

spec

.rspec

address_book.rb

address.rb

contact.rb

contacts.yaml

error_handling.rb

Gemfile

Gemfile.lock

phone_number.rb

README.md

run_address_book.sh

contact.rb

src > contact.rb > ...

1 require "../phone_number"

2 require "../address"

3

4 class Contact

5 attr_writer :first_name, :middle_name, :last_name

6 attr_reader :phone_numbers, :addresses

7

8 def initialize

9 @phone_numbers = []

10 @addresses = []

11 end

12

13 def add_phone_number(kind, number)

14 phone_number = PhoneNumber.new

15 phone_number.kind = kind

16 phone_number.number = number

17 phone_numbers.push(phone_number)

18 end

19

20 def add_address(kind, street_1, street_2, city, state, postal_code)

21 address = Address.new

22 address.kind = kind

23 address.street_1 = street_1

24 address.street_2 = street_2

25 address.city = city

26 address.state = state

27 address.postal_code = postal_code

28 addresses.push(address)

29 end

30

31 def first_name

32 @first_name

33 end

34

master

0 0 0

Live Share

Ln 1, Col 1

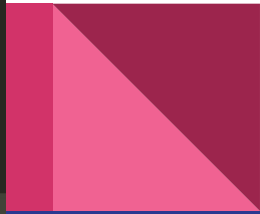
Spaces: 4

UTF-8

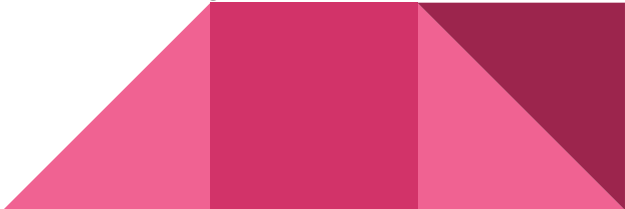
LF

Ruby

Go Live



Phone_number file:

- Make attribute accessor for the kind of numbers such as home or work or mobile.
 - Make string method for kind and number.
 - Add phone numbers to my contacts in the contact file.
 - My contact is attribute reader for phone numbers.
 - Initialize my contact and set the phone number in the empty array.
 - Create a method to add contacts to my contact list with two arguments kind and number.
 - Create method print phone numbers by using each method through the loop.
 - Make access to my phone number file by using require keyword and give a path to the contact file.
- 

EXPLORER

ADDRESSBOOK

docs

ppt

spec

src

image

spec

.rspec

address_book.rb

address.rb

contact.rb

contacts.yaml

error_handling.rb

Gemfile

Gemfile.lock

phone_number.rb

README.md

run_address_book.sh

phone_number.rb

src > phone_number.rb > PhoneNumber

```
1 class PhoneNumber
2   attr_accessor :kind, :number
3
4
5   def to_s
6     "#{kind}: #{number}"
7   end
8 end
```

OUTLINE

TIMELINE

master

0 0 0

Live Share

Ln 1, Col 1

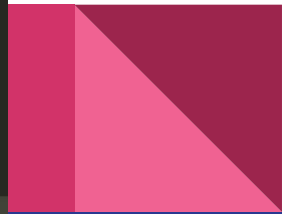
Spaces: 4

UTF-8

LF

Ruby

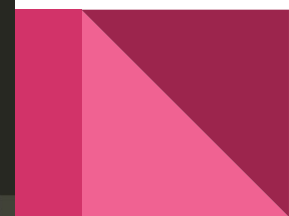
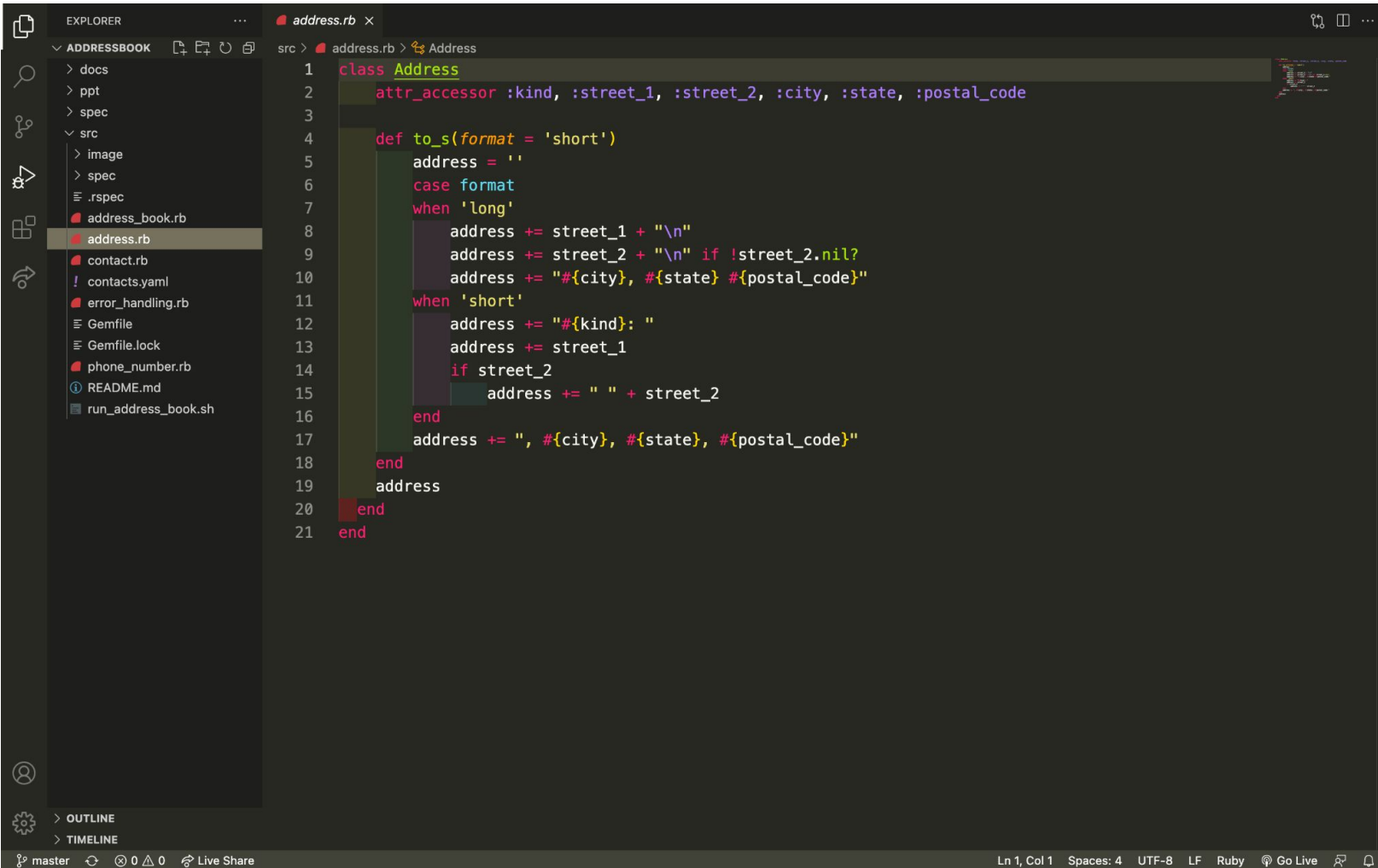
Go Live



Address file:

- Create address class by writing attribute accessor.
- Create a string to hold our address, add kind of address.
- Require the address class to contact the list file.
- Set the addresses like a phone number into the empty array.
- Write a method to address by adding an attribute to address.
- Initialize the new instance of address class.
- Make definition of print addresses.
- Adding addresses by referencing instancing variables.





Address book file:

- Made an attribute reader for contact.
- Write a method called find by name what pass in a name.
- I make an empty array by search result.
- Create method called print result, it takes a label for search and an array of results.
- Making a loop through contact and address by writing definition of find by address
- Use string method before using include method to see if one string is included in another.



EXPLORER

address_book.rb

ADDRESSBOOK

docs

ppt

spec

src

image

spec

.rspec

address_book.rb

address.rb

contact.rb

contacts.yaml

error_handling.rb

Gemfile

Gemfile.lock

phone_number.rb

README.md

run_address_book.sh

OUTLINE

TIMELINE

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

end

end

def save

File.open("contacts.yaml", "w") do |file|

file.write(contacts.to_yaml)

end

end

def run

loop do

puts "Address Book".colorize(:red)

puts "\xF0\x9F\x93\x95 \xF0\x9F\x93\x92 \xF0\x9F\x93\x98 \xF0\x9F\x93\x97 \xF0\x9F\x93\x93"

puts "a: Add Contact".colorize(:yellow)

puts "p: Print Address Book".colorize(:magenta)

puts "s: Search".colorize(:cyan)

puts "e: Exit".colorize(:green)

print "Enter your choice: "

input = gets.chomp.downcase

case input

when 'a'

add_contact

when 'p'

print_contact_list

when 's'

print "Search term: "

search = gets.chomp

find_by_name(search)

find_by_phone_number(search)

find_by_address(search)

when 'e'

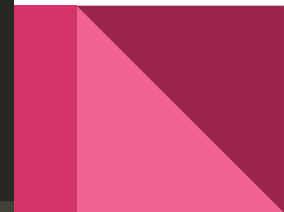
save()

break

end

end

Ln 1, Col 1 Spaces: 4 UTF-8 LF Ruby Go Live



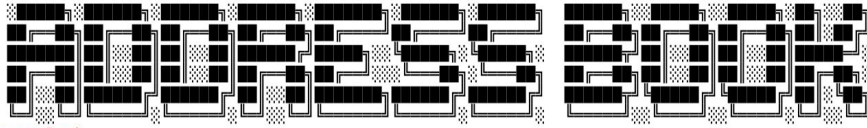
Input and output:

- Create a method call run which will display menu on a loop.
- Initialize a new contact and prompt the user to enter their information about their contact.
- Writing a loop to add a menu to our address book file.
- Making a response case method to help user to choose phone number or address.
- Add last option to save out address book to a file.
- Before exit another method call save method to save our contact list.
- Using Yaml from ruby standard library to save all of the ruby objects in to the text base format.



Using Bash Script to run the app: `./run_address_book.sh`

```
[Shabnams-MacBook-Pro:src shabnamlouie$ ./run_address_book.sh
```



Address Book



a: Add Contact

p: Print Address Book

s: Search

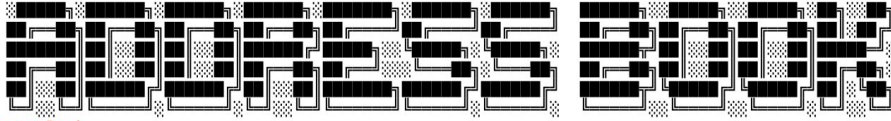
e: Exit

Enter your choice: e

```
Shabnams-MacBook-Pro:src shabnamlouie$ █
```

User interaction: Adding

```
[Shabnams-MacBook-Pro:src shabnamlouie$ ruby address_book.rb
```



Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: a

First name: Shabnam

Middle name:

Last name: Louie

Add phone number or address?

p: Add phone number

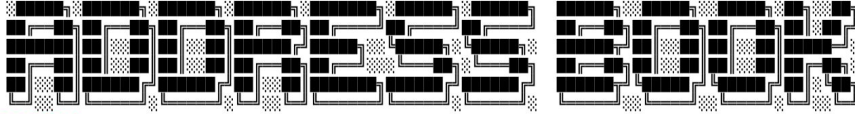
a: Add address

(Any other key to go back)



User interaction: Adding

```
[Shabnams-MacBook-Pro:src shabnamlouie$ ruby address_book.rb
```



Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: a

First name: Shabnam

Middle name:

Last name: Louie

Add phone number or address?

p: Add phone number

a: Add address

(Any other key to go back)

p

Phone number kind (Home, Work, etc): home

Number: 87654327

Add phone number or address?

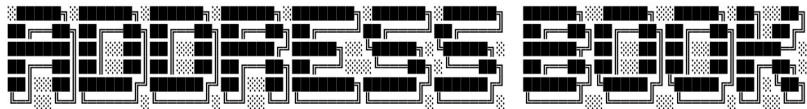
p: Add phone number

a: Add address

(Any other key to go back)

User interaction: Adding

```
[Shabnams-MacBook-Pro:src shabnamlouie$ ruby address_book.rb
```



Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: a

First name: Shabnam

Middle name:

Last name: Louie

Add phone number or address?

p: Add phone number

a: Add address

(Any other key to go back)

p

Phone number kind (Home, Work, etc): home

Number: 87654327

Add phone number or address?

p: Add phone number

a: Add address

(Any other key to go back)

a

Address Kind (Home, Work, etc): home

Address line 1: unit 18

Address line 2: 33 helix street

City: surry hills

State: nsw

Postal Code: 2000

Add phone number or address?

p: Add phone number

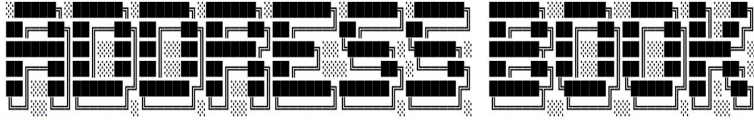
a: Add address


(Any other key to go back)

█

User interaction: Printing

```
|Shabnams-MacBook-Pro:src shabnamlouie$ ruby address_book.rb
```

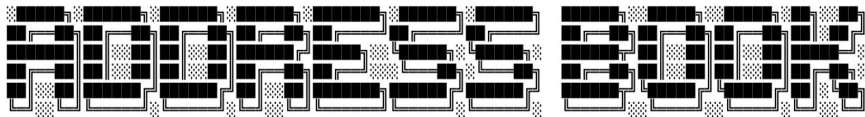


```
Address Book
Enter your choice: p
Your contact list saved
Louie, Shabnam .
Address Book

a: Add Contact
p: Print Address Book
s: Search
e: Exit
Enter your choice: █
```



User interaction: Searching

```
[Shabnams-MacBook-Pro:src shabnamlouie$ ruby address_book.rb
```



Address Book

Enter your choice: p

Your contact list saved

Louie, Shabnam .

Address Book



a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: s

Search term: a

Name search results (a)

Shabnam Louie

Phone Numbers

home: 9876543

Addresses

home: unit 13 70 holy street, surry hills, nsw, 2000

Phone search results (a)

Address search results (a)

Address Book



a: Add Contact

p: Print Address Book

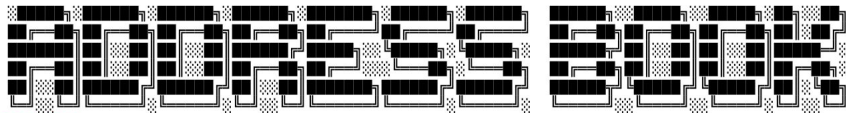
s: Search

e: Exit

Enter your choice: █

User interaction: Exit

```
[Shabnams-MacBook-Pro:src shabnamlouie$ ruby address_book.rb
```



Address Book

Enter your choice: p

Your contact list saved

Louie, Shabnam .

Address Book

📅 📇 📇 📇 📇

a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: s

Search term: a

Name search results (a)

Shabnam Louie

Phone Numbers

home: 9876543

Addresses

home: unit 13 70 holy street, surry hills, nsw, 2000

Phone search results (a)

Address search results (a)

Address Book

📅 📇 📇 📇 📇

a: Add Contact

p: Print Address Book

s: Search

e: Exit

Enter your choice: e

Shabnams-MacBook-Pro:src shabnamlouie\$ █

Challenges:

- Time management
- Too much to learn and understand what is going to happen
- Difficulty to know what gems to use
- Did not know where the codes go
- Never gone into programming before



Favourite part

- coding
- Using colorizer gem
- Saving the detail by yaml



Hope you enjoyed

Thanks for having me :)

