



OOP & Recursion 2

Lab 11



Recap – OOP and Recursion

▶ Lab09

- ▶ We made a Book, Member, Record, and MyLibrary which stored and managed books, and books loans in a library

▶ Lab10

- ▶ We made recursive functions which:
 - ▶ Went through a series of numbers to print different values
 - ▶ Went through lists to count, find, modify, or check some values or property of the list



Coderunner

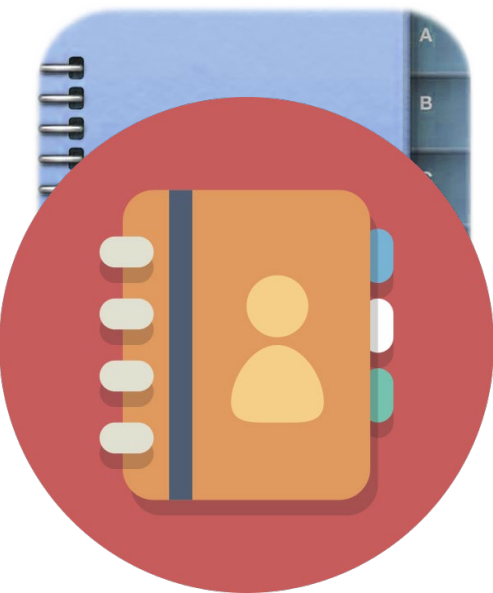
▶ Questions 1 – 4:

- ▶ We will make a `Contact` and a `PhoneBook` class to store information about different people like we might in our contacts list (or in a Phonebook)
- ▶ `PhoneBook` objects will store information about people in `Contact` objects and will use `Contact` object methods to get and change information

▶ Questions 5 – 11:

- ▶ Different recursion questions 🍰

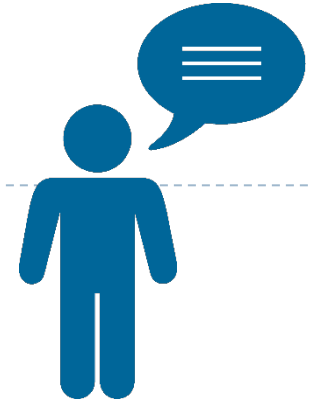
PhoneBook – a list of Contacts



Inactive Record



Contact: represents ONE contact

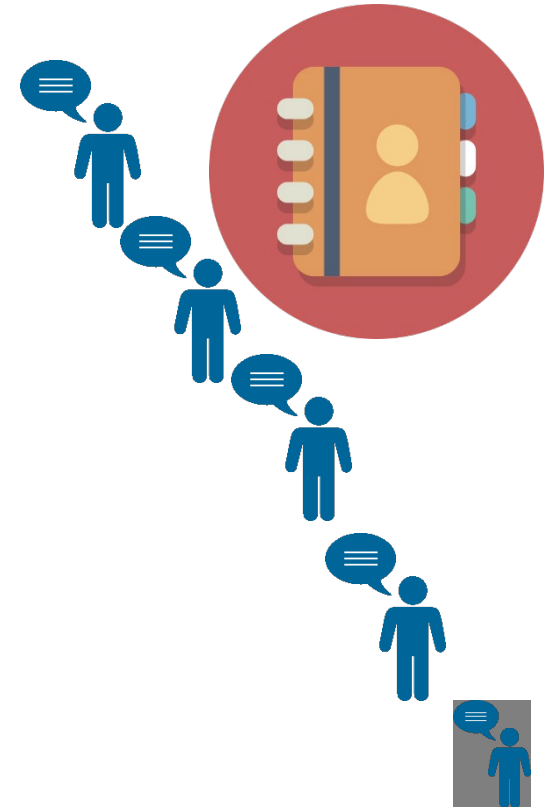


- ▶ `__name`
- ▶ `__phone_number`
- ▶ `__email`
- ▶ `__status`: True represents an active record

- ▶ `get_name(self)`
- ▶ `get_phone_number(self)`
- ▶ `get_email(self)`
- ▶ `is_active(self)`
- ▶ `set_phone_number(self, phone_number)`
- ▶ `set_active(self, value)`
- ▶ `__str__(self)` Kelly (9876543210), kelly@bmail.com
- ▶ ... Nicky is an inactive record.

PhoneBook: represents a list of contacts

- ▶ `self.__phone_book`: the contacts list
- ▶ `load_records(self, filename)`
- ▶ `show_all_records(self)`
- ▶ `show_active_records(self)`
- ▶ `find_record(self, search_name)`
 - ▶ find the record and return the object
- ▶ `update_phone(self, search_name, phone_number)`
 - ▶ find the record and update the phone number
- ▶ `set_inactive(self, search_name)`
 - ▶ find the record and change the status
- ▶ `set_active(self, search_name)`
 - ▶ find the record and change the status





get_sum_XXX (Q5, Q6)

► Work Flow: “abc”
result = 294

return 97 + 197

“abc”
ord(‘a’) + ...

return 98 + 99

“bc”
ord(‘b’) + ...

return 99 + 0

“c”
ord(‘c’) + ...

“”
return 0



Return a number

► Work Flow: [6, 4, 5, 9]

return 5

[6, 4, 5, 9]
compare 6 and get...

return 5

[4, 5, 9]
compare 4 and get...

return 5

[5, 9]
compare 5 and get...

return 9

[9]
if 9 is an odd number,
return it
else return 9999

Return a list ...

► Work Flow: [6, 4, 5, 9]

return [5, 9] [6, 4, 5, 9]
if 6 is an odd number, return [6] + get...
else return get...

return [5, 9] [4, 5, 9]
if 4 is an odd number, return [4] + get...
else return get...

return [5, 9] [5, 9]
if 5 is an odd number, return [5] + get...
else return get...

return [9] [9]
if 9 is an odd number, return [9] + get...
else return get...



Coderunner Tips

- ▶ Q3 – you will need to make Contact **objects** from the information in the file and store these in the `__phone_book` attribute
- ▶ Q4 – In this question you will need to use the methods defined in the Contact class – it might be helpful to look over what methods you wrote for the Contact class and think about how they can be used to update each contact.
- ▶ Q5 – the `ord(a_character)` function gives the ASCII value of a character
- ▶ Q11 – The Euclidean algorithm might help here