Słowa kluczowe
dodaj (A)
wyświetl urodziny (2)
wyszukaj (4)
edytuj (5)
usuń
zarządzaj_notatkami (6), (7), (8)

Kontakt 1 : { imię : . . .
⋮
notatki : „nie lubię go"
„2 pracy"

{ „tel" : True / False
„email" : True / False

{ „tel" : string
„email" : string }

Albo [{ „name" ~~birthday~~ }]

[{ „name" : „birthday"
. . . . }

klasa / moduł Birthday
(2) print [„name", . . .]?

klasa / moduł
Date_verification
(3)

(A) metoda add_contact (self)
interaction z użytkiem
imię = input („podaj imię")
zbieram to w nowy słownik
self.contacts.append (nowy_słownik)

klasa Contacts
(1) lista słowników
(4) metoda search
(5) metody edit/delete
na selfie

(B) get_note
(C) set_note

lista słowników
{ imię : notatki : lista

klasa / moduł Notes
(6)
(7)
(8)

moduł Main
import . . . . .
#Command_handlers
def function (book : Contacts)
⋮
KEYWORDS = {
"keyword" : command_handler
}

def main ():
- tworzy obiekt klasy Contacts
z pusta lista albo wczytanie z pliku

zgłoszenie command_answer :
keyword = input ("czego ode mnie chcesz")
try:
uruchomienie funkcji zgodnej z keyword
except:
keyword = command_assistant (keyword)
if „skończyć"
break
else:
KEYWORDS [keyword] (contacts)

klasa / moduł Command_assistant
podpowiedzi

zły keyword

dobry keyword

if __name__ = "__main__":
main