



**Πανεπιστήμιο Μακεδονίας**  
**Τμήμα Εφαρμοσμένης Πληροφορικής**



# **BookPal**

Chatbot με χρήση Κανονικών Εκφράσεων, γραμμένο σε Python

**Βραζάλης Κωνσταντίνος, ics22115**

Μάθημα: **Θεωρία Υπολογισμού**

Καθηγητής: **Γιάννης Ρεφανίδης**

Θεσσαλονίκη, Απρίλιος 2024

## Πίνακας Περιεχομένων

1. Εισαγωγή.....	3
2. Σκοπός εφαρμογής.....	3
3. Δομή προγράμματος.....	3
3.1 Εισαγωγή.....	3
3.2 Αρχείο book.py.....	4
3.3 Αρχείο regex.py.....	4
3.4 Αρχείο bookpal.py.....	4
3.5 Επισημάνσεις για τα εκτελέσιμα αρχεία.....	5
4. Επεξήγηση Κανονικών Εκφράσεων.....	6
5. Επίδειξη Λειτουργίας Εφαρμογής .....	7
6. Συμπεράσματα.....	10
7. Βιβλιογραφία .....	10

## 1. Εισαγωγή

Το BookPal είναι ένα **chatbot** γραμμένο σε **Python**, το οποίο κάνει χρήση **κανονικών εκφράσεων**, προκειμένου να αναγνωρίζει μοτίβα στις προτάσεις του χρήστη και να αποκρίνεται με ανάλογο τρόπο. Θεματολογία του συγκεκριμένου chatbot είναι η παροχή πληροφοριών σχετικά με **βιβλία**. Ο χρήστης μπορεί να ρωτήσει το BookPal πληροφορίες σχετικά με την ημερομηνία έκδοσης, τη γλώσσα γραφής, την περίληψη και τον σύνδεσμο ηλεκτρονικής αγοράς ενός βιβλίου. Προκειμένου να υπάρχει ευρύτερη λειτουργικότητα, χρησιμοποιείται το Application Programming Interface (API) βιβλίων της Google, ώστε να παρέχονται αντίστοιχες πληροφορίες για οποιοδήποτε βιβλίο θελήσει ο χρήστης. Έτσι, πρωταρχική του ερώτηση είναι αν υπάρχει βιβλίο με τον εκάστοτε τίτλο που έχει στο μυαλό του. Εάν το BookPal αποκριθεί θετικά, ο χρήστης μπορεί να διατυπώσει τις παραπάνω ερωτήσεις. Ο πηγαίος κώδικας της εφαρμογής, καθώς και τα αντίστοιχα εκτελέσιμα αρχεία για Linux, Windows και macOS, βρίσκονται στο [GitHub](#) του δημιουργού.

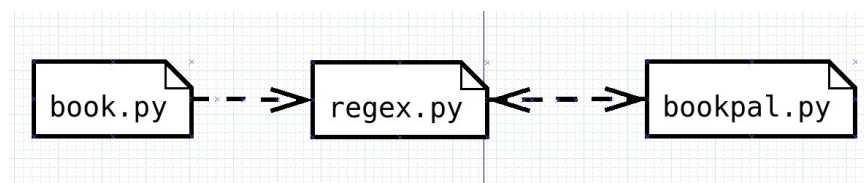
## 2. Σκοπός εφαρμογής

Σκοπός της παρούσας εργασίας είναι η εξοικείωση με την συγγραφή και αναγνώριση κανονικών εκφράσεων σε προγραμματιστικό επίπεδο. Για το σκοπό αυτό χρησιμοποιείται το **module re**<sup>[1]</sup> της Python, το οποίο παρέχει τις κατάλληλες λειτουργίες για την επίτευξη αυτού το σκοπού. Σε δεύτερο επίπεδο επιχειρείται η δημιουργία μίας πρότυπης γραφικής διασύνδεσης για την προσομοίωση μιας εφαρμογής ανταλλαγής μηνυμάτων με τη βιβλιοθήκη γραφικών **tkinter**<sup>[2]</sup>, ενώ τέλος γίνεται μια γνωριμία με τη χρήση APIs για την άντληση πληροφοριών από το διαδίκτυο, με τη χρήση του **Google Books API**<sup>[3]</sup>.

## 3. Δομή προγράμματος

### 3.1 Εισαγωγή

Προκειμένου να είναι πιο ευανάγνωστος ο κώδικας, το πρόγραμμα χωρίζεται σε **τρία αρχεία**, το καθένα εκ των οποίων αναλαμβάνει την υλοποίηση του **API**, την **αναγνώριση των κανονικών εκφράσεων** και την **υλοποίηση της γραφικής διασύνδεσης**, αντίστοιχα. Τα αρχεία αυτά είναι συνδεδεμένα (μέσω *import*), όπως φαίνεται στο παρακάτω σχήμα<sup>[4]</sup>:



Εικόνα 1: Δομή των αρχείων του προγράμματος

Λεπτομερής επεξήγηση των λειτουργιών του κάθε αρχείου δίνεται στις αντίστοιχες υποενότητες. Αξίζει να σημειωθεί ότι κάθε αρχείο χρησιμοποιεί τα δεδομένα που λαμβάνει από το προηγούμενο ως αφηρημένα εργαλεία και δεν ασχολείται με τις λεπτομέρειες της υλοποίησής τους. Συνεπώς, ο τελικός χρήστης αρκεί να εκτελέσει το *bookpal.py* (το αρχείο της γραφικής διασύνδεσης) και το πρόγραμμα θα είναι πλήρως λειτουργικό.

### 3.2 Αρχείο book.py

Το συγκεκριμένο αρχείο είναι υπεύθυνο για την αποθήκευση στοιχείων για ένα βιβλίο. Για να το πετύχει αυτό δημιουργείται μια **κλάση Book**, που περιέχει μία μοναδική μέθοδο, η οποία **επιστρέφει** ένα μονοδιάστατο **πίνακα συμβολοσειρών**, όπου κάθε στοιχείο του πίνακα αποτελεί το είδος της πληροφορίας που υποδηλώνει το όνομά του. Έτσι, στον πίνακα `book_info[]` που επιστρέφεται, περιλαμβάνονται: πληροφορίες για τίτλο και συγγραφέα, λεπτομέρειες για την έκδοση, πληροφορίες για τη γλώσσα, την περίληψη και τον σύνδεσμο ηλεκτρονικής αγοράς. Τα στοιχεία αυτά προστίθενται μετέπειτα στις απαντήσεις του BookPal στον χρήστη, ώστε κάθε απάντηση να είναι εξατομικευμένη για ένα συγκεκριμένο βιβλίο.

Προκειμένου να λάβουν τα παραπάνω πεδία τιμές, χρησιμοποιείται η βιβλιοθήκη `requests`<sup>[5]</sup> της Python για την πραγματοποίηση αιτημάτων **HTTP** στη διεύθυνση <https://www.googleapis.com/books/v1/volumes>, με παράμετρο τον τίτλο του βιβλίου που εισήγαγε ο χρήστης. Για λόγους απλότητας γίνεται ταύτιση των μεταβλητών με τα στοιχεία της πρώτης μόνο αντιστοίχισης τίτλου, καθώς η μέθοδος `requests.get()` επιστρέφει όλες τις δυνατές αντιστοιχίες που βρίσκει το API.

### 3.3 Αρχείο regex.py

Το αρχείο αυτό είναι υπεύθυνο για τη **διαμόρφωση** των απαντήσεων στις ερωτήσεις του χρήστη. Σκοπός του είναι η αναγνώριση των κανονικών εκφράσεων στη συμβολοσειρά εισόδου. Για τον λόγο αυτό κάνει `import re`, για την χρήση των λειτουργιών της βιβλιοθήκης. Ακόμη, εισάγει το αρχείο `book.py`, για τον **εμπλουτισμό** των απαντήσεων, όπως αναφέρθηκε στην παραπάνω υποενότητα.

Οι κανονικές **εκφράσεις**, μαζί με τις αντίστοιχες **απαντήσεις** τους, βρίσκονται στον πίνακα `patterns_responses[]`. Στην αριστερή στήλη βρίσκονται οι εκφράσεις, όπως αυτές έχουν οριστεί με τον τρόπο που περιγράφεται στην Ενότητα 4. Στην δεξιά στήλη βρίσκονται τα αρχικά τμήματα των προτάσεων που θα απαντήσει το BookPal.

Στη μέθοδο `chatbot()`, δεδομένης της συμβολοσειράς εισόδου, γίνεται διαδοχική σάρωση του πίνακα και καλείται η μέθοδος `re.match()` για να γίνει **αντιστοίχιση** της πρότασης του χρήστη σε κάποιο **μοτίβο**, αν αυτό ανιχνευθεί. Στη συνέχεια γίνεται εμπλουτισμός των ήδη υπάρχουσών απαντήσεων, ανάλογα της θέσης τους στον πίνακα και τη θέση του αντίστοιχου στοιχείου στον πίνακα `book_info[]`. Εάν δεν υπάρξει καμία αντιστοίχιση μοτίβου, το BookPal απαντάει με γενικόλογο τρόπο, ότι δεν κατανοεί την ερώτηση. Η μέθοδος, τέλος, **επιστρέφει** μια συμβολοσειρά με την **απάντηση** του BookPal, για να χρησιμοποιηθεί από το επόμενο αρχείο.

### 3.4 Αρχείο bookpal.py

Το αρχείο αυτό είναι υπεύθυνο για τη δημιουργία του **περιβάλλοντος αλληλεπίδρασης** με τον χρήστη. Το όνομα του είναι γενικόλογο για τη διευκόλυνση της εκτέλεσης, καθώς αυτό είναι το αρχείο που χρειάζεται να εκτελεστεί για λειτουργήσει ολόκληρο το πρόγραμμα.

Εισάγεται η βιβλιοθήκη `tkinter` για τη δημιουργία των γραφικών και το αρχείο `regex` για τη λήψη των απαντήσεων του BookPal. Η διαδικασία δημιουργίας των στοιχείων του παραθύρου δεν θα αναλυθεί σε αυτό το έγγραφο, καθώς δεν εμπίπτει στο πλαίσιο της εργασίας (ο κώδικας περιέχει αναλυτικά σχόλια). Σε γενικές γραμμές, δημιουργούνται δύο **frames**, ένα για την εισαγωγή του μηνύματος του χρήστη στο αντίστοιχο πεδίο εισόδου και ένα για την απεικόνιση των μηνυμάτων του χρήστη και του BookPal.

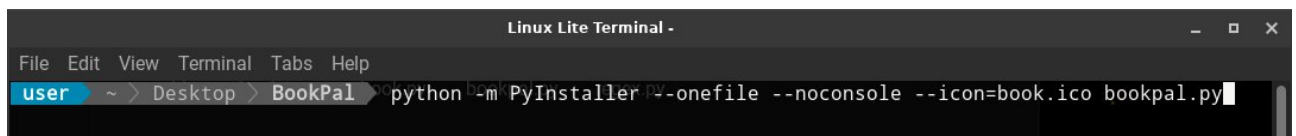
Ουσιαστική λειτουργικότητα επιτελεί η συνάρτηση `send_message()`, η οποία εμφανίζει το **μήνυμα του χρήστη** και στη συνέχεια την **απόκριση του BookPal** για τη δεδομένη είσοδο, μέσω της μεθόδου `regex.chatbot()`, η οποία δέχεται ως παράμετρο το εισαχθέν μήνυμα.

Ουσιαστικά, το συγκεκριμένο αρχείο αναλαμβάνει να μεταδώσει το μήνυμα του χρήστη στο `regex.py`, ώστε να ανιχνευθεί η κανονική έκφραση, και στη συνέχεια εμφανίζει την απάντηση που επιστρέφει η μέθοδος `chatbot()`. Η διαδικασία αυτή **επαναλαμβάνεται** μέχρι να τερματίσει ο χρήστης το πρόγραμμα **χειροκίνητα**, καθώς, για λόγους απλότητας, δεν υπάρχει κάποια συγκεκριμένη πρόταση εισόδου που να σταματά την εκτέλεση.

### 3.5 Επισημάνσεις για τα εκτελέσιμα αρχεία.

Για να εκτελεστεί το πρόγραμμα, αρκεί ο χρήστης να τρέξει την **εντολή** `python bookpal.py` στην τοποθεσία `/path_to_folder/BookPal/`, σε ένα **τερματικό** Linux ή macOS και στο CMD/PowerShell των Windows ή να τρέξει γραφικά το αρχείο από το **IDE** του. Στη συνέχεια θα εμφανιστεί το παράθυρο της εφαρμογής. Καθώς όμως υπάρχει το ενδεχόμενο κάποιος χρήστης να μην έχει εγκατεστημένα κάποια από τα modules, θα του εμφανιστεί το αντίστοιχο **μήνυμα σφάλματος** και θα πρέπει να τα εγκαταστήσει μέσω του **pip**<sup>[5]</sup>.

Προκειμένου να μην προκύψει το παραπάνω πρόβλημα, αλλά και ο χρήστης να έχει τη δυνατότητα να εκτελέσει κατευθείαν το πρόγραμμα, η Python δίνει την δυνατότητα να **συμπεριστούν όλα τα αρχεία** και να **συμπεριληφθούν όλες οι βιβλιοθήκες** σε ένα εκτελέσιμο αρχείο, μέσω του module **PyInstaller**<sup>[6]</sup>. Στον φάκελο BookPal περιλαμβάνονται εκτελέσιμα αρχεία για **Linux**, **Windows** και **macOS**, τα οποία δημιουργήθηκαν με την εκτέλεση της εντολής:



Εικόνα 2: Εντολή συμπίεσης σε εκτελέσιμο αρχείο

Στην παραπάνω εικόνα:

- m**: υποδηλώνει την χρήση ενός module της Python
- PyInstaller**: το όνομα του module
- onefile**: υποδηλώνει ότι το εκτελέσιμο θα είναι μοναδικό αρχείο
- noconsole**: υποδηλώνει ότι δεν θα ανοίγει τερματικό κατά την εκτέλεση
- icon**: δηλώνει το εικονίδιο<sup>[7]</sup> του εκτελέσιμου (το .ico αρχείο περιλαμβάνεται στον φάκελο BookPal)
- bookpal.py**: δηλώνει το αρχείο που θα συμπιεστεί

Η εκτέλεση της παραπάνω εντολής δημιουργεί στον φάκελο `BookPal/dist/` το **εκτελέσιμο** αρχείο, ανάλογα με τον τύπο του λειτουργικού συστήματος. Το αρχείο αυτό μπορεί να εκτελεστεί σε οποιονδήποτε συμβατό υπολογιστή, **χωρίς** να είναι απαραίτητα **εγκατεστημένη η Python**. Σημειώνεται ότι σε **Windows 10** και άνω, καθώς ζητείται να συμπεριστεί και κώδικας που χρησιμοποιεί API, δεν θα επιτραπεί η εκτέλεση της εντολής όσο είναι ενεργοποιημένο το **Windows Defender** ή γενικά κάποιο **Antivirus**. Αντίστοιχα, αν επιχειρηθεί να εκτελεστεί το `bookpal.exe` σε κάποιον υπολογιστή με ενεργοποιημένο ένα από τα παραπάνω, θα υπάρξει **προειδοποιητικό μήνυμα ασφαλείας**. Στην πραγματικότητα, **δεν συντρέχει κανένας απολύτως κίνδυνος** από την εκτέλεση του .exe αρχείου, καθώς προέρχεται από **ανοιχτό, ασφαλή κώδικα**.

## 4. Επεξήγηση Κανονικών Εκφράσεων

Στην παρακάτω ενότητα αναλύονται όλες οι κανονικές εκφράσεις που αναγνωρίζει το BookPal, όπως αυτές συναντώνται στο αρχείο *regex.py*:

a) `hello|hi|hey`

Το χαιρετιστήριο μήνυμα του χρήστη. Αναγνωρίζεται οποιαδήποτε από τις παραπάνω λέξεις, ακολουθούμενη από οποιοδήποτε συνδυασμό λέξεων (ή την κενή). Έτσι, αναγνωρίζονται φράσεις όπως “hi” ή “hello there” ή “hi friend”. Το BookPal αποκρίνεται συστήνοντας τον εαυτό του.

b) `(.*)(thank you|thanks|(good)?bye)`

Το αποχαιρετιστήριο μήνυμα του χρήστη. Αναγνωρίζεται οποιοσδήποτε συνδυασμός λέξεων (ή η κενή), ακολουθούμενη από τις τρεις εναλλακτικές επιλογές της δεύτερης παρένθεσης (με το “good” να είναι προαιρετικό) και στη συνέχεια οποιονδήποτε συνδυασμό λέξεων. Έτσι, αναγνωρίζονται φράσεις όπως “okay thanks” ή “great goodbye” ή “thank you bye”. Το BookPal αποχαιρετά ευχόμενο καλή ανάγνωση.

c) `(.*)(know about|know|tell me about)( the| a)?( book named| book called)? (.*)`

Το μήνυμα του χρήστη για εύρεση ενός βιβλίου. Αναγνωρίζεται οποιοσδήποτε συνδυασμός λέξεων, ακολουθούμενος από μία από τις εναλλακτικές της δεύτερης παρένθεσης, στη συνέχεια το “the” ή το “a” (αλλά προαιρετικά), στη συνέχεια προαιρετικά μία από τις επιλογές της επόμενης παρένθεσης και, τέλος, μια οποιαδήποτε συμβολοσειρά, η οποία θα αποτελέσει και το κλειδί αναζήτησης βιβλίου (γι’ αυτό και τοποθετείται το “(.\*)”, καθώς εξάγονται τα περιεχόμενα της τελευταίας παρένθεσης). Έτσι, αναγνωρίζονται φράσεις όπως “do you know a book named lord of the rings” ή “can you tell me about the lord of the rings” ή “I would like to know about lord of the rings”, και πολλοί άλλοι συνδυασμοί. Το BookPal αποκρίνεται πως θα αναζητήσει την συμβολοσειρά που εξήγαγε από την πρόταση και στη συνέχεια εμφανίζει τα αποτελέσματα της αναζήτησης μέσω του API.

d) `(.*)when(.*)published|(.*)publish date`

Ερώτηση σχετικά με την ημερομηνία έκδοσης. Αναγνωρίζεται οποιοσδήποτε συνδυασμός λέξεων, ακολουθούμενος από το “when” ακολουθούμενος από οποιοδήποτε συνδυασμό λέξεων, ακολουθούμενος από τις δύο εναλλακτικές σχετικά με την έκδοση και στη συνέχεια οποιαδήποτε συμβολοσειρά. Έτσι αναγνωρίζονται φράσεις όπως “do you know when it was published?” ή “can you tell me the publish date?”. Το BookPal αποκρίνεται αναφέροντας την ημερομηνία έκδοσης και τον εκδοτικό οίκο.

e) `(.*)(what|which)?(.*)language`

Ερώτηση σχετικά με την γλώσσα συγγραφής. Ακολουθώντας παρόμοια λογική με τις προηγούμενες ερωτήσεις, αναγνωρίζονται φράσεις όπως “do you know what language it is written in” ή “do you know the language?”. Το BookPal αποκρίνεται αναφέροντας την γλώσσα.

f) `(.*)(give|provide|tell)(.*)(description|info|details|summary)`

Ερώτηση σχετικά με την περίληψη. Ακολουθώντας παρόμοια λογική με τις προηγούμενες ερωτήσεις, αναγνωρίζονται φράσεις όπως “could you provide more details” ή “give me a summary” ή “can you tell me some info” και πολλά άλλα. Το BookPal αποκρίνεται παραθέτοντας την περίληψη του βιβλίου.

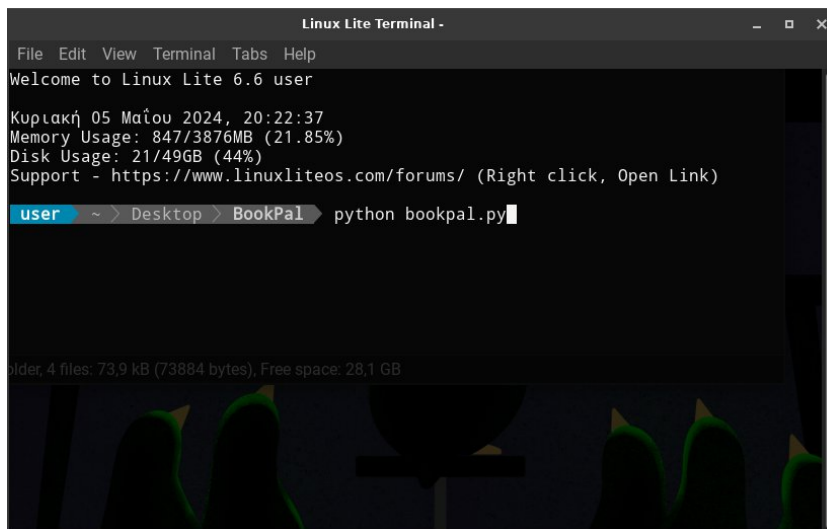
g) `(.*)(buy|purchase)(link|it)`

Ερώτηση σχετικά με το σύνδεσμο ηλεκτρονικής αγοράς. Ακολουθώντας παρόμοια λογική με τις προηγούμενες ερωτήσεις, αναγνωρίζονται φράσεις όπως “give me a buy link” ή “can you tell me where I can purchase it?”. Το BookPal αποκρίνεται δίνοντας τον σύνδεσμο, αν αυτός υπάρχει. Αν όχι, αναφέρει ότι είναι άγνωστος.

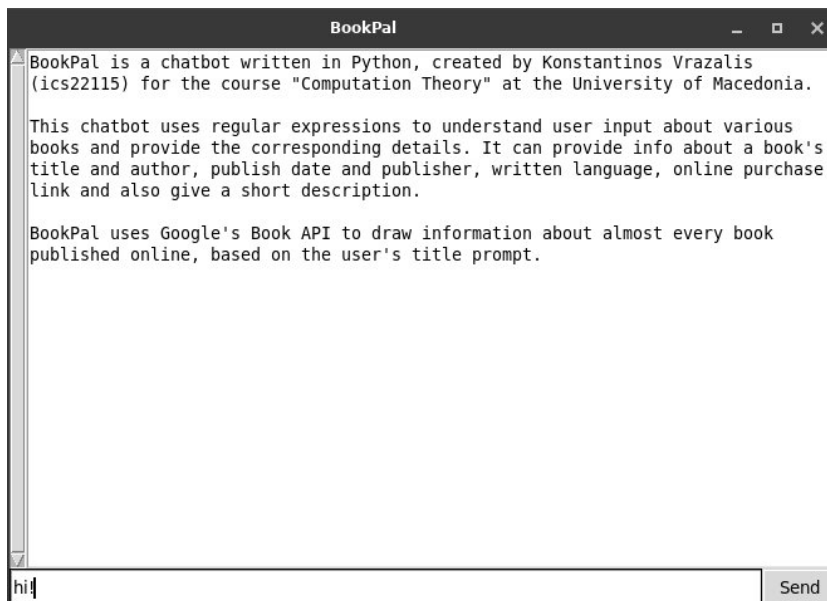
Όλες οι παραπάνω κανονικές εκφράσεις αγνοούν την ύπαρξη κεφαλαίων γραμμάτων, καθώς χρησιμοποιείται η συνάρτηση `lower()` στην είσοδο του χρήστη. Επίσης έχουν σχεδιαστεί με τρόπο ώστε να αναγνωρίζουν όσο το δυνατό περισσότερες προτάσεις του ίδιου είδους. Έτσι, από μία κανονική έκφραση που περιέχει πολλές εναλλακτικές σε πολλές παρενθέσεις, μπορούν να προκύψουν όλοι οι συνδυασμοί των στοιχείων που εμπλέκονται.

## 5. Επίδειξη Λειτουργίας Εφαρμογής

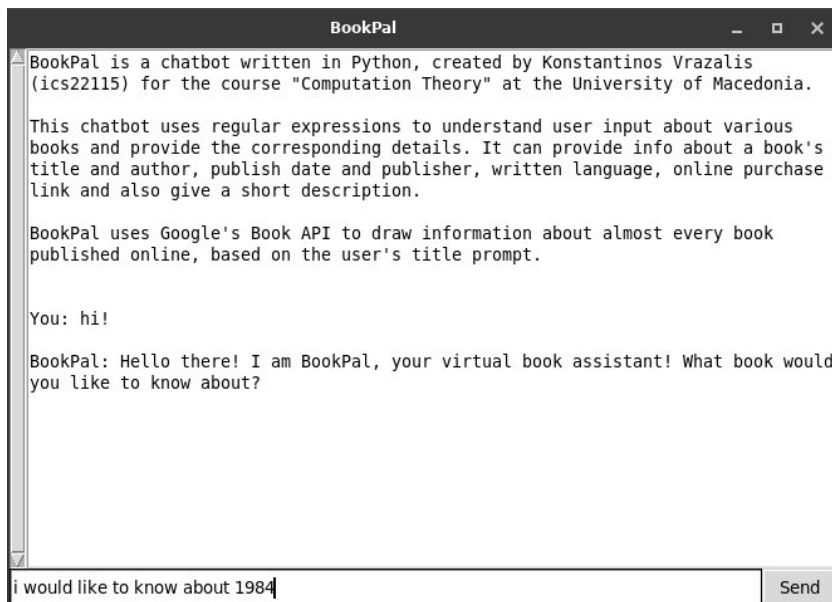
Στη παρούσα ενότητα απεικονίζονται στιγμιότυπα από την εκτέλεση της εφαρμογής, όπου πραγματοποιούνται όλοι οι πιθανοί τύποι ερωτήσεων που μπορεί να διατυπώσει ο χρήστης, συν την περίπτωση που η είσοδος δεν γίνεται κατανοητή. Η πρώτη εικόνα αποτυπώνει τον επιλεγμένο τρόπο εκτέλεσης της εφαρμογής (όπως αναφέρεται και στην υποενότητα 3.5, αυτός δεν είναι ο μοναδικός).



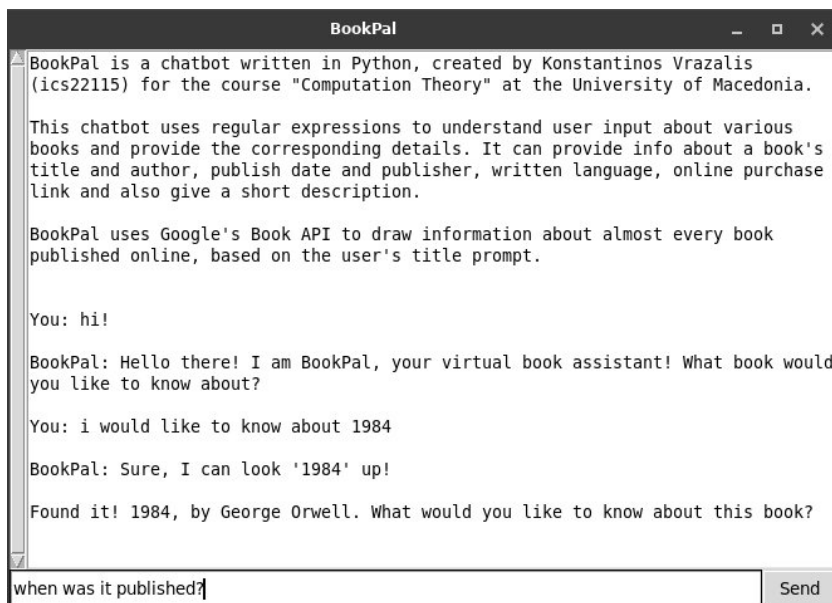
1: Εκτέλεση εφαρμογής.



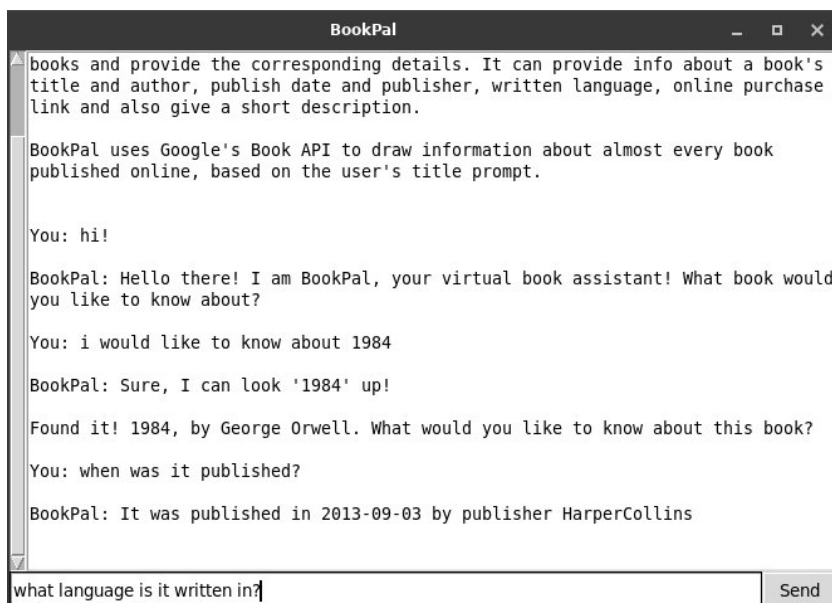
2: Χαιρετιστήριο μήνυμα.



3: Αναζήτηση βιβλίου.

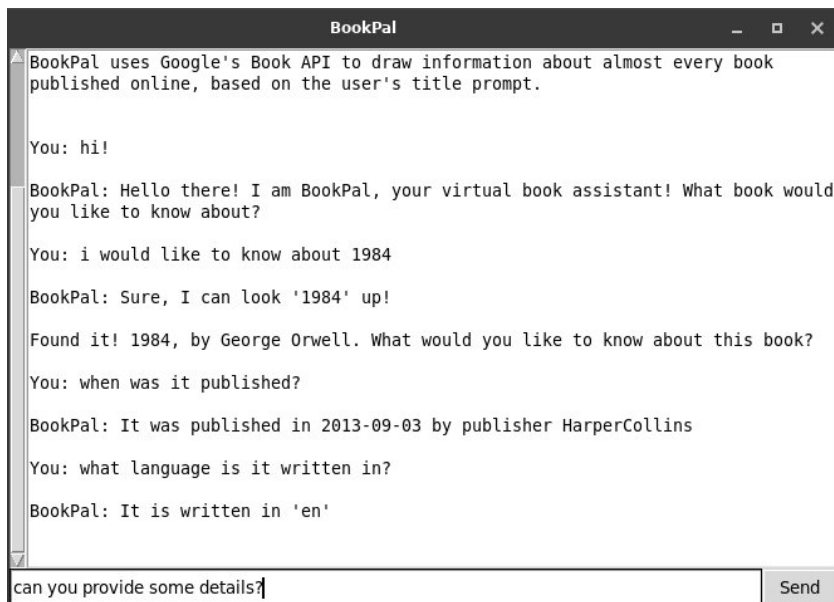


4: Ερώτηση για ημερομηνία έκδοσης.

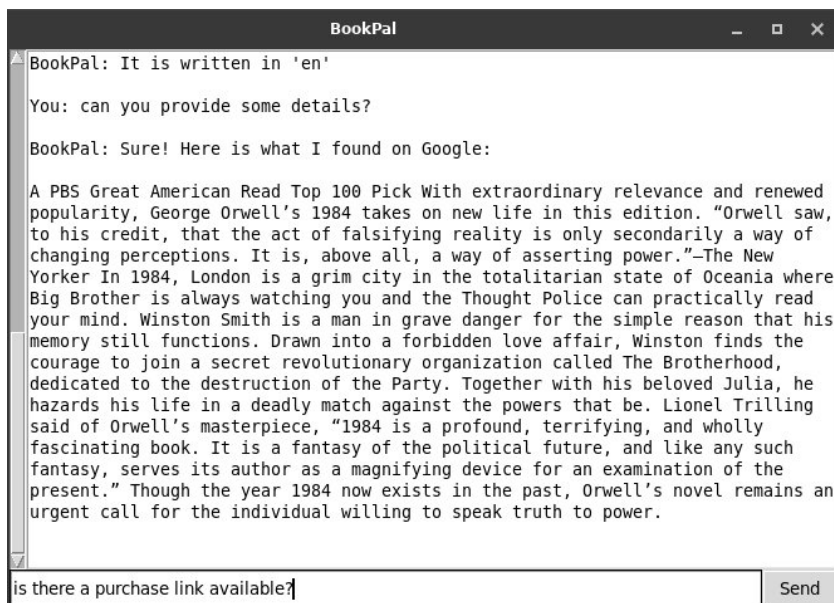


5: Ερώτηση για γλώσσα.

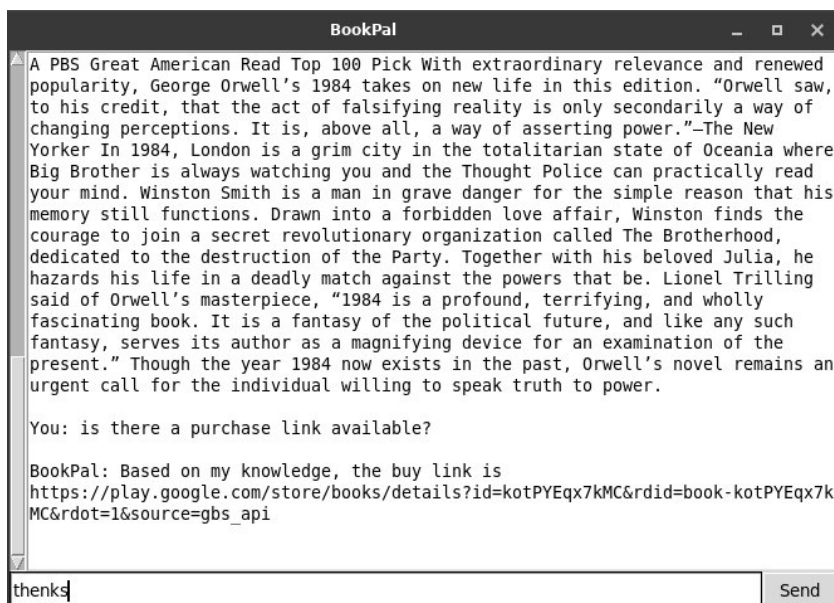




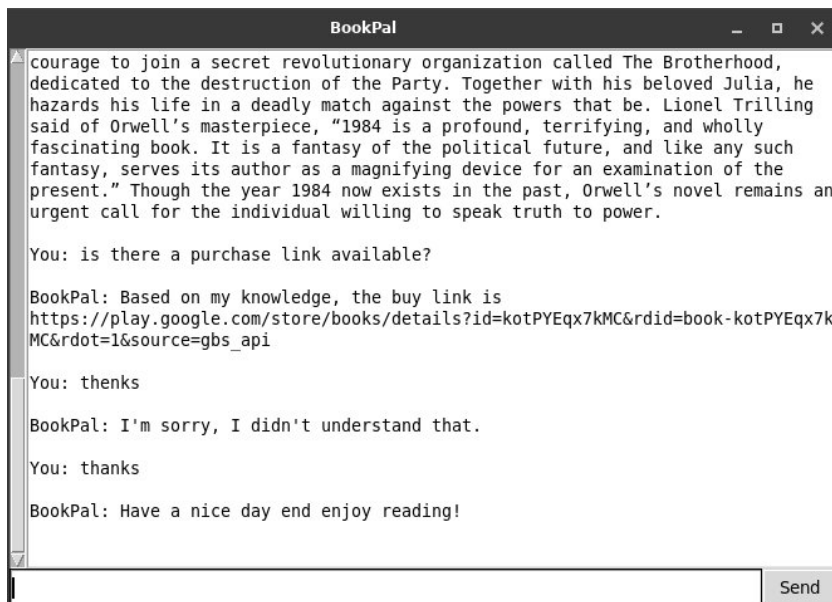
6: Ερώτηση για περίληψη.



7: Ερώτηση για σύνδεσμο αγοράς.



8: Διατύπωση λάθος μηνύματος.



9: Αποχαιρετιστήριο μήνυμα.

## 6. Συμπεράσματα

Η παρούσα εργασία αποτέλεσε μια εισαγωγή στην χρήση κανονικών εκφράσεων με τη γλώσσα προγραμματισμού Python. Σε δευτερεύον επίπεδο δόθηκε και η δυνατότητα εξοικείωσης με τη γραφική διασύνδεση προγραμμάτων Python, αλλά και με τη χρήση APIs για την άντληση πληροφορίας από το διαδίκτυο. Μελλοντική επέκταση της εφαρμογής θα μπορούσε να περιλαμβάνει βελτιστοποίηση των υπαρχουσών ή προσθήκη νέων κανονικών εκφράσεων, αλλά και βελτίωση της γραφικής διασύνδεσης, ώστε να ομοιάζει περισσότερο με ένα περιβάλλον ανταλλαγής μηνυμάτων. Τέλος, εξήχθη το συμπέρασμα πως η χρήση APIs μπορεί να διευρύνει σημαντικά τη λειτουργικότητα μίας εφαρμογής, ώστε αυτή να αποτελέσει ένα πραγματικά χρήσιμο εργαλείο.

## 7. Βιβλιογραφία

1. Τεκμηρίωση του βιβλιοθήκης re: <https://docs.python.org/3/library/re.html>
2. Τεκμηρίωση της βιβλιοθήκης tkinter: <https://docs.python.org/3/library/tkinter.html>
3. Google Books API: <https://developers.google.com/books>
4. Για τη σχεδίαση της Εικόνας 1 χρησιμοποιήθηκε το λογισμικό Dia: <http://dia-installer.de/doc/index.html>
5. Τεκμηρίωση βιβλιοθήκης requests: <https://pypi.org/project/requests>
6. Τεκμηρίωση pip: <https://pypi.org/project/pip>
7. Τεκμηρίωση PyInstaller: <https://pyinstaller.org/en/stable>
8. Σύνδεσμος εικονιδίου εφαρμογής: <https://icon-icons.com/download/25711/ICO/512/>