



Τέταρτο παραδοτέο εξαμηνιαίας εργασίας στον Αντικειμενοστραφή Προγραμματισμό II  
(Deadline 18 Ιανουαρίου)

## Υπηρεσία Συστάσεων Πόλεων με χρήση Open Data

Ερώτημα 1 (lecture 10)

Να παραλληλοποιήσετε κάποιες λειτουργίες του προγράμματός σας με **Thread Pools** [1]. Ενδεικτικά την διαδικασία Retrieve data από την wikipedia και την διαδικασία Retrieve data από το OpenWeatherMap.

Ερώτημα 2 (lecture 11)

Οι συστάσεις που κάνουμε προς τους υποψήφιους Travellers μέχρι τώρα είναι generic. Δημιουργήστε την μέθοδο personalizedRecommend() και χρησιμοποιήστε **Streams** και **Lambda Expressions** [2]. Ένας υποψήφιος Traveller θα δηλώνει explicitly τον βαθμό [0,1] που τον ενδιαφέρουν τα 7 terms και μέσω ενός dot product θα του γίνεται recommend η πόλη που του ταιριάζει περισσότερο.

Ενδεικτική προσέγγιση: Φτιάξτε ένα Stream που θα χρησιμοποιεί Lambda Expressions όπου θα γίνεται ένα pipeline από operations όπως οι filter, map, limit, sorted, ή reduction (max) [3]:

1. Θα παίρνει ως είσοδο το collection των City και θα το μετατρέπει σε Stream.
2. Θα συγκρίνει τα κριτήρια (7-dim vector representation) που επέλεξε ο Traveller με κάθε City (τα 7 από τα 10 features) του Stream με ένα dot product.
3. Θα επιστρέφει την πόλη που επισκέφτηκε ο Traveller που έχει αποτέλεσμα το μεγαλύτερο value στην πράξη του dot product.

Ερώτημα 3 (lecture 10)

Δημιουργήστε ένα σύστημα καταγραφής αρχείων (log files) [4] όπου θα καταγραφονται ενέργειες στην εκτέλεση του προγράμματος, όπως η ανάκτηση δεδομένων από τα OpenData, η αποθήκευση στο JSON file και οι επιλογές που έκανε ο user στο GUI.

[1]<https://www.tutorialspoint.com/thread-pools-in-java>

[2] <https://dzone.com/articles/java-8-concepts-fp-lambda-expressions-and-streams>

[3]<https://docs.oracle.com/javase/8/docs/api/java/util/stream/Stream.html#max-java.util.Comparator->

[4]<https://examples.javacodegeeks.com/core-java/util/logging/java-util-logging-example/>