

1. ΕΙΣΑΓΩΓΗ ΣΤΟ GRASS GIS

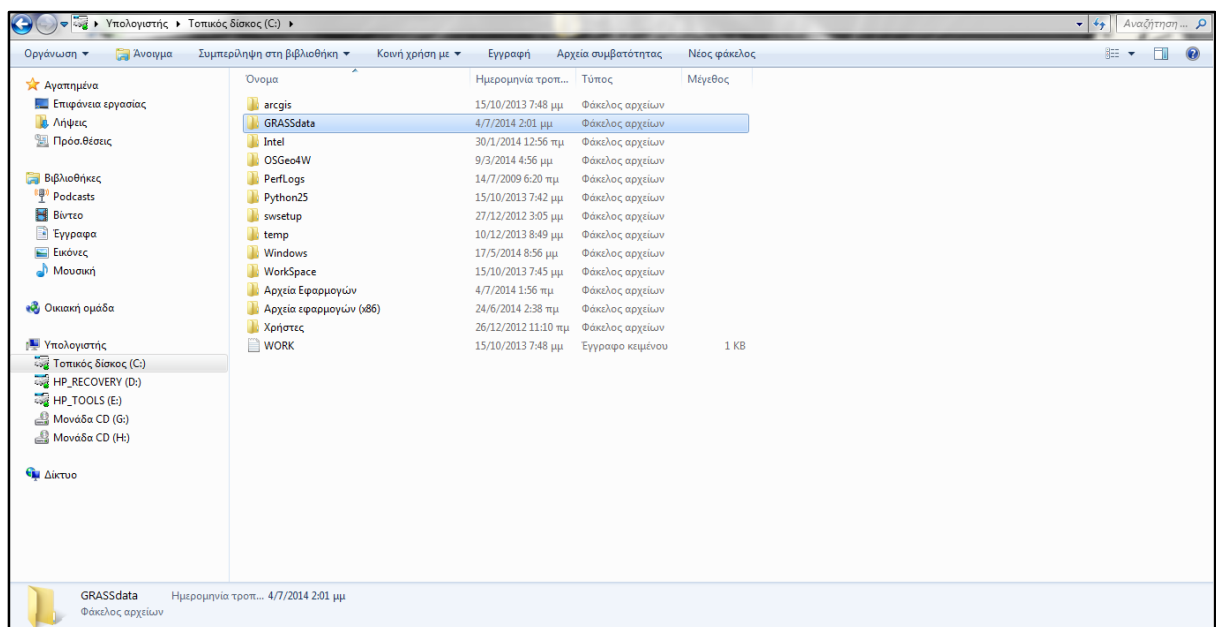
1.1. ΤΙ ΕΙΝΑΙ ΤΟ GRASS GIS ;

Το GRASS (Geographical Resources Analysis Support System) αποτελεί ένα raster/vector φορέα / κομμάτι του GIS που συνδυάζει την ολοκληρωμένη επεξεργασία εικόνας με την απεικόνιση δεδομένων υποσυστημάτων. Περιλαμβάνει πάνω από 350 μονάδες για τη διαχείριση, ανάλυση, επεξεργασία και οπτικοποίηση των δεδομένων με γεωγραφική αναφορά. Το κλειδί για την ανάπτυξη του προγράμματος GRASS αποτέλεσε η υιοθέτηση της GNU GPL (General Public License) οδηγώντας έτσι το GRASS το 1999 στη φιλοσοφία των Open Source προγραμμάτων. Έτσι το GRASS ως ένα ανοιχτό λογισμικό επιτρέπει σε όλους τους χρήστες του να συμμετέχουν σε κριτικές, διορθώσεις και βελτιώσεις στη δομή και στους αλγόριθμους του προγράμματος φέρνοντας χαμηλότερο κόστος για τη διαχείρισή του.

2. ΞΕΚΙΝΩΝΤΑΣ ΜΕ ΤΟ GRASS GIS

2.1. GRASSDATA, LOCATION, MAPSET

Ξεκινώντας μία εργασία στο GRASS, όλα τα δεδομένα αποθηκεύονται σε έναν κατάλογο (GISDBASE) ή όπως αλλιώς αποκαλείται *GRASSDATA*. Αυτός ο φάκελος δημιουργείται προτού ξεκινήσει η επεξεργασία των δεδομένων στο πρόγραμμα GRASS, στον προσωπικό μας χώρο δουλειάς.



Μέσα σε αυτό το φάκελο (GRASSdata) τα δεδομένα οργανώνονται και τοποθετούνται σε υποκαταλόγους που ονομάζονται *LOCATIONS* (τοποθεσία). Κάθε τοποθεσία (Location) ορίζεται απ' ένα σύστημα συντεταγμένων, ένα προβολικό σύστημα και από γεωγραφικά

όρια. Ξεκινώντας λοιπόν το πρόγραμμα, πρέπει αρχικά να ορίσουμε την περιοχή απ' την οποία το πρόγραμμα θα παίρνει τα δεδομένα. Πρόκειται για το *GIS Directory* ή αλλιώς *path* το οποίο ορίζει στο πρόγραμμα τον ακριβές χώρο δουλειάς μας.

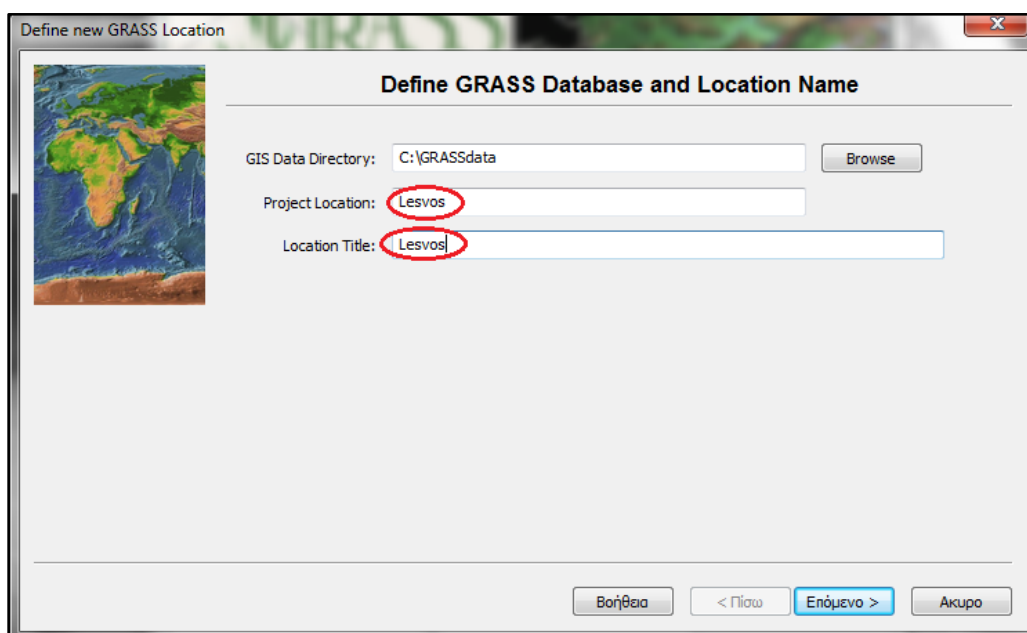


Στη συγκεκριμένη περίπτωση ως *GIS Directory* ορίζουμε τον προσωπικό μας χώρο, εκεί όπου έχουμε δημιουργήσει το φάκελο GRASSdata (C: \GRASSdata).

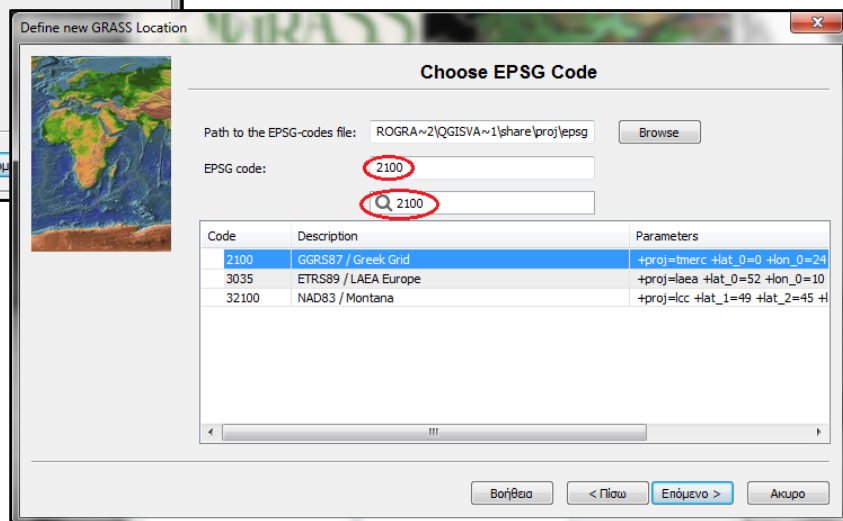
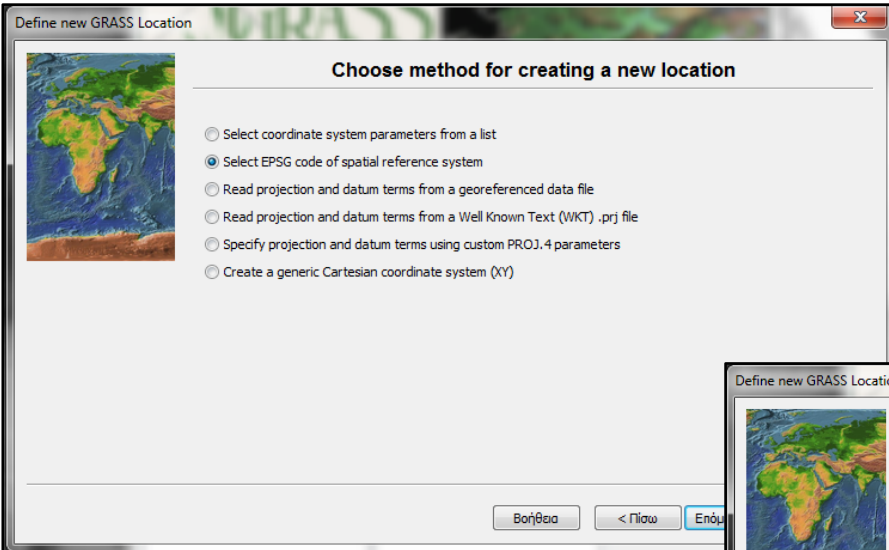
Στη συνέχεια, πρέπει να δημιουργήσουμε την τοποθεσία μας (LOCATION), εκεί δηλαδή όπου και θα ορίσουμε σύστημα συντεταγμένων και προβολικό σύστημα.



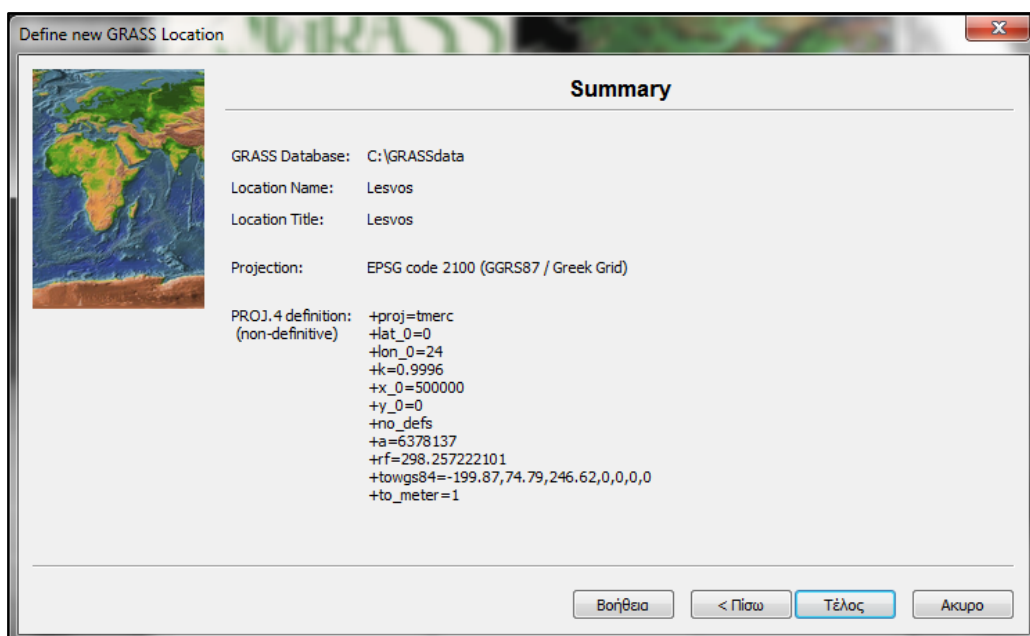
Στο LOCATION μπορούμε να δώσουμε ότι ονομασία θέλουμε (*Project Location, Location Title*). Συνήθως προτιμάται το όνομα της εργασίας την οποία πρόκειται να εκτελέσουμε (πχ. Lesvos) ή αν πρόκειται να μοιραστούμε το αποτέλεσμα με αρκετούς χρήστες συνίσταται η χρήση των αρχικών του ονόματός μας.



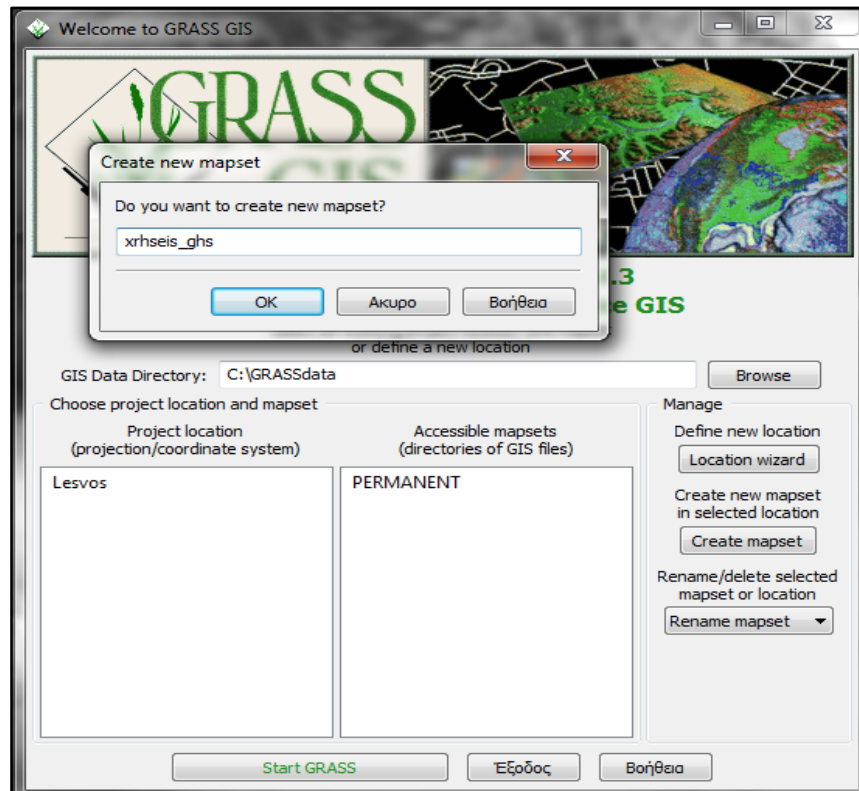
Επίσης, εδώ δεν πρέπει να ξεχνάμε και τον ορισμό ενός προβολικού συστήματος (*EPSG code*). Συνήθως επιλέγεται η χρήση του ελληνικού προβολικού συστήματος με EPSG 2100 ή το WGS 84 UTM zone 35N, μέσω της αναζήτησης.



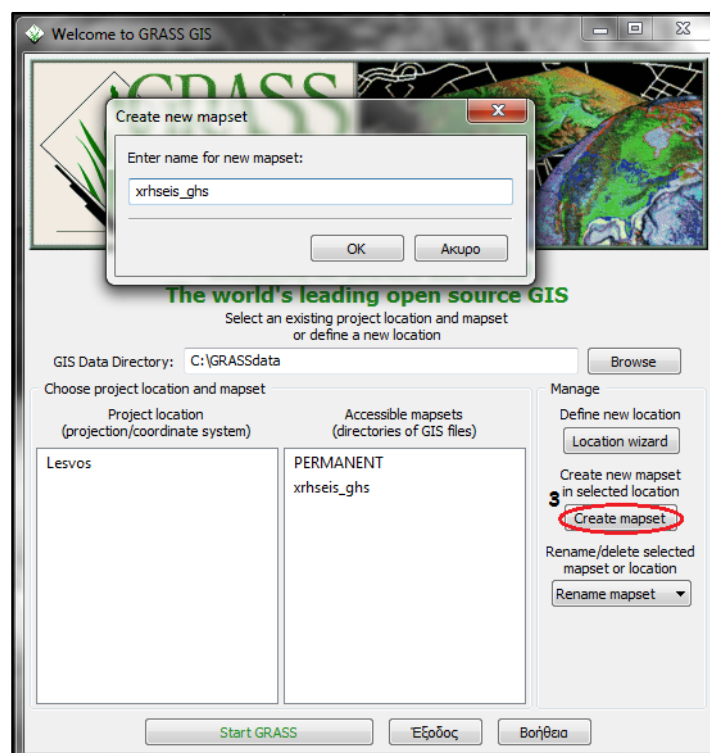
Με βάση τις παραπάνω ενέργειες έχουμε πλέον ολοκληρώσει τη δημιουργία μιας τοποθεσίας (Location) για το πρόγραμμά μας, η οποία φέρει ονομασία, συντεταγμένες και προβολικό σύστημα.



Κάθε *Location* μπορεί να έχει και τα δικά του *mapsets*, υποφακέλους δηλαδή μέσα στους οποίους μπορούμε να αποθηκεύσουμε στοιχεία της δουλειάς μας χωρίζοντάς τα σε κατηγορίες. Μετά τον ορισμό της τοποθεσίας (*Location*), το πρόγραμμα περνάει απευθείας στη δημιουργία ενός *mapset*, στο οποίο μπορούμε να δώσουμε ένα όνομα.



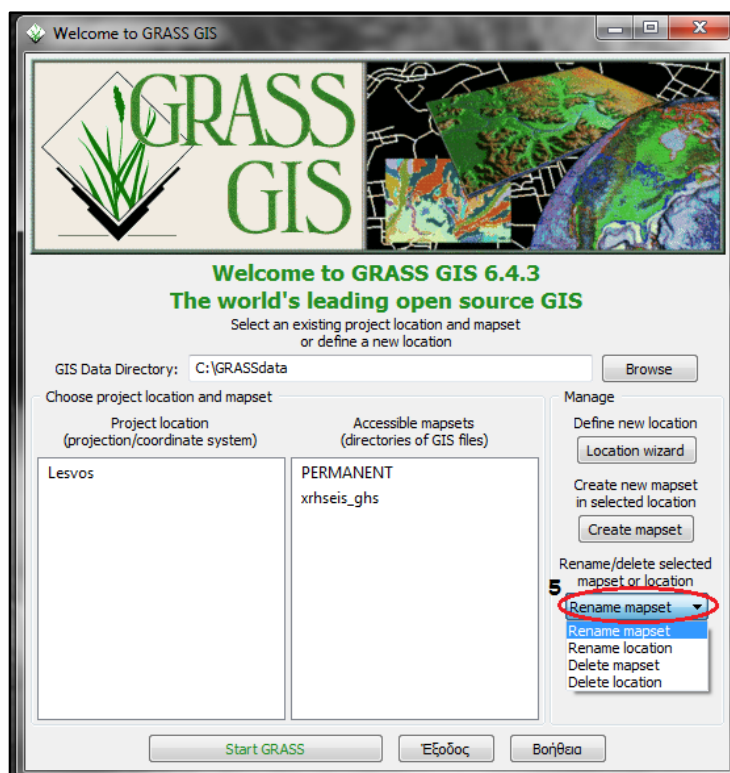
Κάθε χρήστης εκτός απ' τα δικά του *mapsets*, μπορεί να δει και των υπολοίπων χρηστών χωρίς όμως να έχει τη δυνατότητα να τα τροποποιήσει. *Mapsets* μπορούμε να δημιουργήσουμε και εμείς όποια στιγμή θέλουμε. (*Create Mapset*)



Κάθε φορά που δημιουργείται μια τοποθεσία (Location), το GRASS δημιουργεί αυτόματα ένα ειδικό mapset (PERMANENT) στο οποίο αποθηκεύονται βασικά δεδομένα για το έργο που δουλεύουμε. Μόνο ο κάτοχος αυτού του mapset (PERMANENT) μπορεί να προσθέσει, να τροποποιήσει ή να καταργήσει τα δεδομένα του, τα οποία μπορούν να αντιγραφούν και να αναλυθούν και από άλλους χρήστες αντίστοιχα.



Επίσης, μπορούμε να μετονομάσουμε ένα mapset ή ένα location ακόμη και να τα διαγράψουμε.



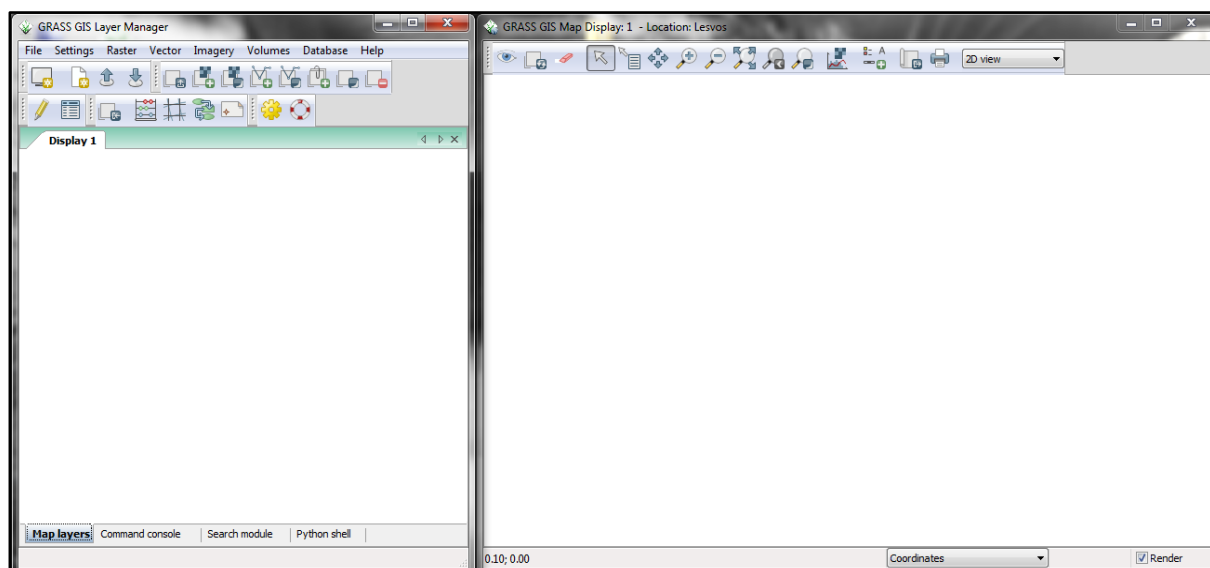
Έτσι, αφού έχουμε κάνει όλες τις παραπάνω διεργασίες και έχουμε επιλέξει το mapset στο οποίο θα δουλέψουμε,



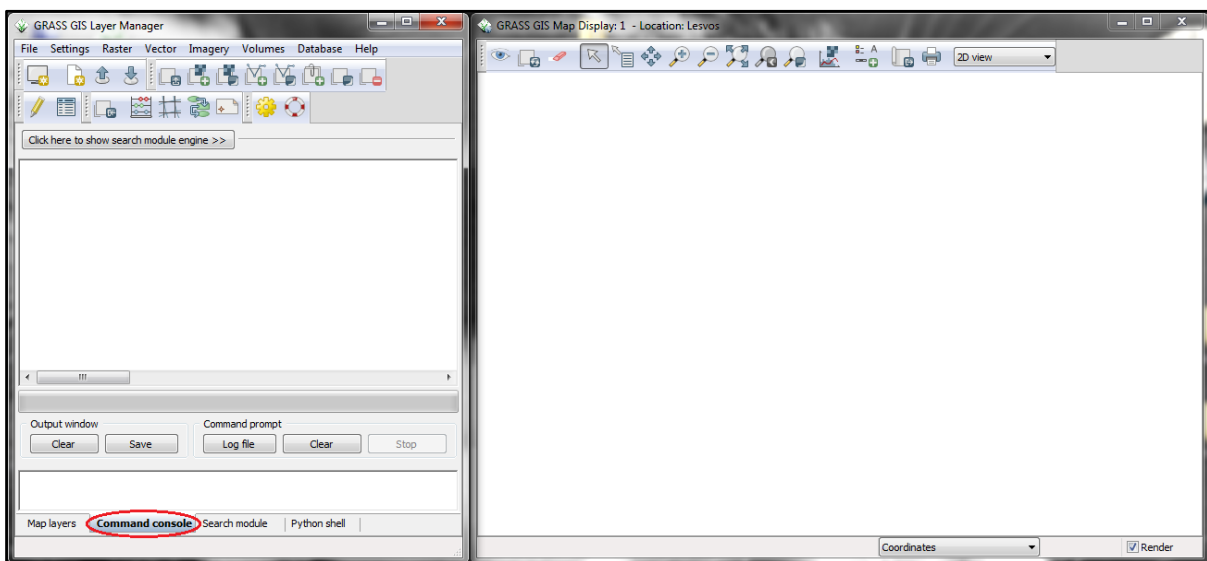
μπορούμε να ξεκινήσουμε την περιήγησή μας στο GRASS.

2.2. ΤΙ ΠΡΕΠΕΙ ΝΑ ΓΝΩΡΙΖΩ ;

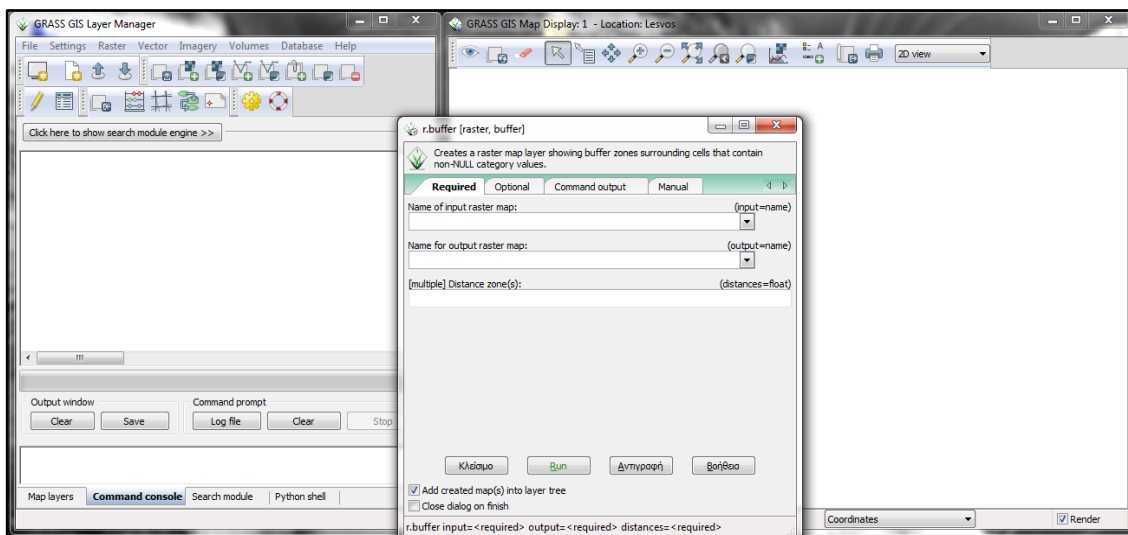
Το GRASS αποτελεί ένα παραθυρικό πρόγραμμα σε Python γλώσσα.



Είναι ένα Open Source πρόγραμμα στο οποίο οποιαδήποτε ενέργεια γίνεται αποθηκεύεται αυτόματα. Κάθε εντολή ή ενότητα του προγράμματος GRASS, οργανώνεται με βάση το όνομα ανάλογα την τάξη της λειτουργίας τους. Το πρώτο τους γράμμα αναφέρεται στη λειτουργία και ακολουθείται από μία τελεία και από μία ή δύο άλλες λέξεις ξανά χωριζόμενες από τελείες, περιγράφοντας τη συγκεκριμένη εντολή. Για παράδειγμα η εντολή `v.in.org` είναι μια εντολή για εισαγωγή αρχείων διανυσματικής μορφής (vector) σε διάφορες μορφές, η εντολή `r.buffer` υπολογίζει μία ζώνη εγγύτητας (Buffer Zone) γύρω από γραμμές ή και περιοχές raster, η εντολή `d.rast` μας εμφανίζει χάρτες σε raster μορφή κτλ. Οι συγκεκριμένες εντολές βρίσκονται σε κάθε λειτουργία του προγράμματος GRASS. Για περισσότερη ευκολία, η κάθε εντολή μπορεί να πληκτρολογηθεί στο κομμάτι *Command Console* του προγράμματος.



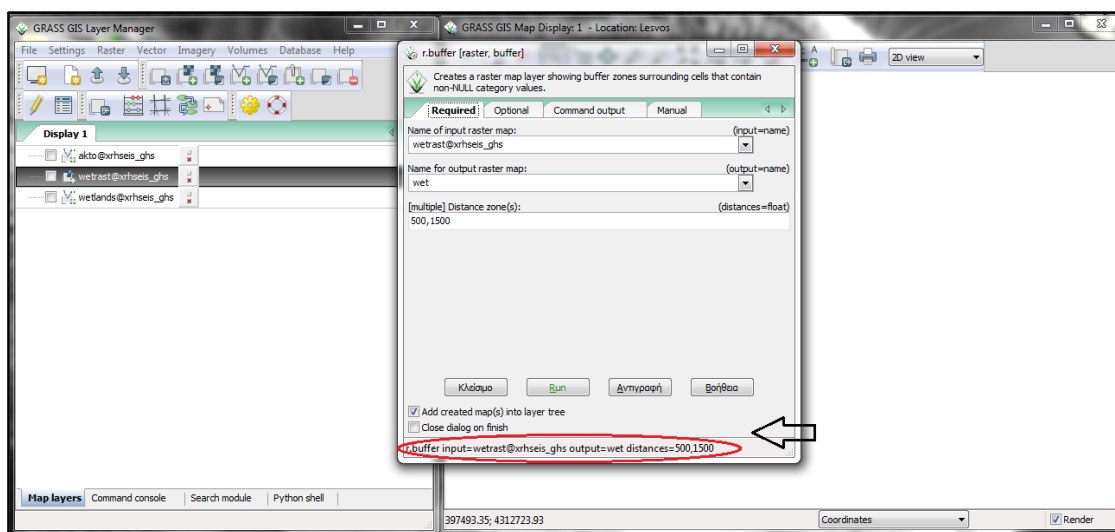
Έτσι γράφοντας την επιθυμητή εντολή στο *Command Console* (πχ. `r.buffer`) και πατώντας ENTER εμφανίζονται τα στοιχεία αυτής που πρέπει να συμπληρωθούν έτσι ώστε να εκτελεστεί.



Παρακάτω παρατίθενται κάποιες συντομεύσεις οι οποίες θα κάνουν την ανάγνωση και το χειρισμό των εντολών του προγράμματος πιο εύκολη/ο αντίστοιχα.

ΠΡΟΘΕΜΑ	ΚΑΤΗΓΟΡΙΑ ΛΕΙΤΟΥΡΓΙΑΣ	ΕΙΔΟΣ ΕΝΤΟΛΗΣ
d.*	display	Εμφάνιση γραφικών δεδομένων
db.*	database	Διαχείριση βάσης δεδομένων
g.*	general	Γενικές λειτουργίες
i.*	imagery	Επεξεργασία εικόνας
m.*	misc	Διάφορες εντολές
ps.*	postscript	Δημιουργία χάρτη σε μορφή Postscript
r.*	raster	Επεξεργασία δεδομένων raster (2D)
r3.*	3D raster	Επεξεργασία δεδομένων raster (3D)
v.*	vector	Επεξεργασία δεδομένων vector (2D,3D)

Επίσης, δεν πρέπει να ξεχνάμε ότι το GRASS είναι ένα πρόγραμμα το οποίο λειτουργεί στη βάση των «εντολών». Κάθε εντολή που εκτελούμε καταγράφεται (πχ. `r.buffer input=wetlands@xrhseis_ghs output=wet distance=500, 1500`, όπου `wetlands` το όνομα του αρχείου που επεξεργαζόμαστε και `xrhseis_ghs` το όνομα του mapset στο οποίο δουλεύουμε).



Έτσι, κατά τη διάρκεια εκτέλεσης μιας εργασίας η οποία απαιτεί τη χρήση αρκετών εντολών στο GRASS, σε περίπτωση διορθώσεων μπορούμε να κρατήσουμε την εντολή που εμφανίζεται στο *Command Console* σε ένα notepad, να τη διορθώσουμε και να ξανατρέξουμε την παρούσα ενέργεια.

