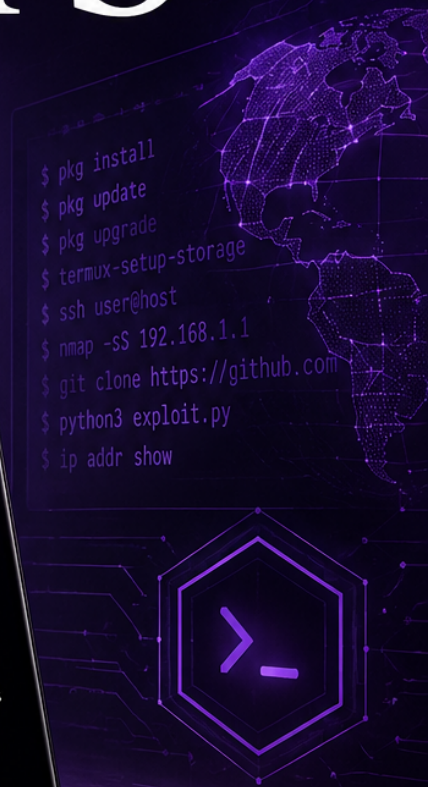




# MASTER TERMUX IN 7 DAYS



# Πίνακας περιεχομένων

Ο πίνακας δείχνει πού ξεκινά κάθε μάθημα. Το εξώφυλλο δεν μετράει ως σελίδα. Αν διαβάζεις σε κινητό, χρησιμοποίησε την αναζήτηση του PDF και γράψε τον τίτλο του μαθήματος.

A/A	Ενότητα	Σελίδες
1	Εισαγωγή - πώς σε μαθαίνει αυτό το βιβλίο	4
2	Μέρος 1 - Μαθήματα, εξηγήσεις και καθοδηγούμενα βήματα	5
3	Πιο βαθιά κατανόηση - τι είναι πραγματικά το Termux	6
4	Πώς διαβάζεις μια εντολή πριν την τρέξεις	7
5	Πιο αναλυτικά για Android APK και permissions	8
6	Πιο χρήσιμες συνήθειες για καθημερινό Termux	9
7	Πριν ξεκινήσεις - κινητό, internet, ασφάλεια και σημειώσεις	10
8	Κατέβασμα και εγκατάσταση APK - ακριβή βήματα στο κινητό	11-12
9	Πρώτο άνοιγμα Termux - τι σημαίνει η οθόνη	13
10	Storage access - Downloads, shared storage και paths	14
11	Package management - update, upgrade, install, search	15
12	Πλοήγηση - folders, files, κενά και ασφαλές σβήσιμο	16
13	Nano editor - άνοιγμα, γράψιμο, save, exit	17
14	GitHub από browser και από Termux	18
15	Τρέξιμο Python scripts με ασφάλεια	19
16	Project 1 - DedSec Project: τι κάνει και πώς το μελετάς	20
17	Project 2 - Offline Survival Project: offline βάση γνώσης	21
18	Project 3 - Corrupted Files Project: offline archive υποθέσεων	22
19	Επαληθευμένα links για projects και README οδηγίες	23
20	DedSec Project - README install guide με ακριβή βήματα	24-25
21	Offline Survival Project - README install/run guide με ακριβή βήματα	26-27
22	Corrupted Files Project - README download/use guide με ακριβή βήματα	28-29
23	Καθημερινό workflow - η ίδια ρουτίνα για κάθε project	30
24	Terminal troubleshooting - τι σημαίνουν τα errors	31-33
25	APK και Android troubleshooting - προβλήματα εγκατάστασης	34
26	Git και GitHub troubleshooting - clone, pull, access	35
27	Python troubleshooting - modules, paths και script names	36
28	Command library - τι γράφεις και πότε	37-39
29	Περισσότερα χρήσιμα - backups, logs, exports και cleanup	40-41

A/A	Ενότητα	Σελίδες
30	Πλάνο 7 ημερών με ασκήσεις και ελέγχους	42
31	Τελικό checklist - τι πρέπει να μπορείς να κάνεις μόνος σου	43
32	Android settings appendix - ακριβώς πού να κοιτάξεις	44-45
33	APK install walkthrough - κάθε οθόνη που μπορεί να δει ο χρήστης	46-47
34	Termux keyboard και touch controls - μικρά πράγματα που χάνουν οι αρχάριοι	48
35	ZIP files, extract και μεταφορά projects από Downloads	49
36	Πώς διαβάζεις README σαν οδηγίες και όχι σαν διακόσμηση	50
37	Project worksheets - συγκεκριμένες εργασίες για τα τρία practice projects	51
38	Terminal decision trees - διάλεξε τον πρώτο ασφαλή έλεγχο	52
39	Extra practice labs - για να δυναμώσει ο χρήστης	53
40	Μέρος 2 - Ασκήσεις και πρακτικά εργαστήρια	54
41	Άσκηση 1 - prompt και output	55
42	Άσκηση 2 - φάκελοι, αρχεία και ασφαλής διαγραφή	56
43	Άσκηση 3 - storage και Downloads	57
44	Άσκηση 4 - packages	58
45	Άσκηση 5 - nano	59
46	Άσκηση 6 - README σαν οδηγίες	60
47	Άσκηση 7 - DedSec Project worksheet	61
48	Άσκηση 8 - Offline Survival worksheet	62
49	Άσκηση 9 - Corrupted Files worksheet	63
50	Άσκηση 10 - μικρό Python script	64
51	Άσκηση 11 - local server	65
52	Άσκηση 12 - troubleshooting drills	66
53	Τελική εργασία - καθαρό Termux workspace	67
54	Πίνακας αυτοαξιολόγησης	68
55	Μέρος 3 - Android σε PC ή USB για Termux και Android apps	69
56	Ποιες επιλογές υπάρχουν και ποια να διαλέξεις	70-71
57	Λεξιλόγιο που πρέπει να ξέρεις πριν πειράξεις USB ή δίσκο	72
58	Πριν κατεβάσεις OS - έλεγχος hardware και backup	73-74
59	Bliss OS - τι να ξέρεις πριν το χρησιμοποιήσεις	75
60	Android-x86 - απλή επιλογή για δοκιμή Android σε PC	76

A/A	Ευότητα	Σελίδες
61	FydeOS - επιλογή τύπου laptop με Android apps	77
62	Waydroid - Android apps μέσα σε Linux χωρίς νέο Android OS	78-79
63	ChromeOS Flex και άλλα ονόματα που πρέπει να προσέξεις	80
64	Πώς φτιάχνεις bootable USB σε Windows με Rufus	81
65	Πώς φτιάχνεις bootable USB σε Linux με dd χωρίς να σβήσεις λάθος δίσκο	82
66	Πώς κάνεις boot από USB χωρίς να χαθείς στο BIOS	83
67	Live boot test checklist - πριν από εγκατάσταση	84
68	Εγκατάσταση σε δίσκο ή δεύτερο USB/SSD - τι σημαίνει κάθε επιλογή	85
69	Πώς βάζεις Termux μέσα στο Android PC περιβάλλον	86-87
70	Γιατί κάποια Android apps δεν δουλεύουν σε PC Android	88
71	Troubleshooting για Android OS σε PC/USB	89-90
72	Ασκήσεις Μέρους 3 - Android PC/USB lab	91-92

# Εισαγωγή - πώς σε μαθαίνει αυτό το βιβλίο

Αυτό το βιβλίο είναι γραμμένο για κάποιον που μπορεί να μην έχει χρησιμοποιήσει ποτέ terminal. Δεν θεωρεί δεδομένο ότι ξέρεις πού να πατήσεις, τι είναι APK, τι σημαίνει path φακέλου ή γιατί χρειάζεται ένα update command.

Κάθε μάθημα λειτουργεί σαν πρακτικό φύλλο εργασίας: πρώτα καταλαβαίνεις την ιδέα, μετά κάνεις μικρά βήματα, μετά ελέγχεις αν πέτυχε. Ο στόχος δεν είναι να αποστηθίσεις εντολές. Ο στόχος είναι να ξέρεις τι κάνεις πριν το τρέξεις.

## Τρόπος διαβάσματος

Διάβασε ένα μικρό block, κάν' το αμέσως στο κινητό και γράψε το αποτέλεσμα στις σημειώσεις σου. Αν η οθόνη σου φαίνεται διαφορετική, σταμάτα και σύγκρινε τις ακριβείς λέξεις της οθόνης σου με τη γραμμή "τι πρέπει να δεις".

- Μην κάνεις paste εντολή που δεν καταλαβαίνεις.
- Μην εγκαθιστάς APK από τυχαίες πηγές.
- Κράτα screenshots ή σημειώσεις από errors. Τα errors είναι στοιχεία, όχι αποτυχία.
- Χρησιμοποίησε τα official links του βιβλίου και όχι εντολές από τυχαία posts.

# Μέρος 1 - Μαθήματα, εξηγήσεις και καθοδηγούμενα βήματα

Το πρώτο μέρος εξηγεί το περιβάλλον και δείχνει ακριβώς τι κάνεις στο κινητό. Μην το διαβάσεις σαν άρθρο. Διάβασέ το με το Termux ανοιχτό και με ένα notes app δίπλα σου.

## Στόχος Μέρους 1

Να καταλάβεις settings, APK εγκατάσταση, paths, commands, packages, GitHub repos, Python scripts, local links και terminal errors πριν περάσεις στις ασκήσεις.

# Πιο βαθιά κατανόηση - τι είναι πραγματικά το Termux

Το Termux είναι Linux-like περιβάλλον μέσα στο Android. Δεν είναι root, δεν αντικαθιστά το Android και δεν βλέπει αυτόματα όλα τα αρχεία του κινητού. Σου δίνει έναν ξεχωριστό χώρο εργασίας όπου μπορείς να εγκαθιστάς packages, να κατεβάζεις repositories, να γράφεις scripts και να τρέχεις local εργαλεία.

- Το home του Termux είναι συνήθως `/data/data/com.termux/files/home`.
- Τα Android Downloads είναι συνήθως `/storage/emulated/0/Download`.
- Το σύμβολο `~` σημαίνει Termux home, όχι ολόκληρο το κινητό.
- Το prompt δείχνει ότι το terminal περιμένει εντολή.
- Το output είναι η απάντηση της εντολής και πρέπει να το διαβάζεις πριν συνεχίσεις.

## Μοντέλο σκέψης

Πού βρίσκεσαι: Πριν από κάθε command.

Τι κάνεις: Πρώτα: πού είμαι, τι αρχείο θέλω, ποια εντολή τρέχω, τι θα αλλάξει, τι αποτέλεσμα περιμένω;

Τι πρέπει να δεις: Μπορείς να απαντήσεις σε αυτά πριν πατήσεις Enter.

Γιατί το κάνεις: Αυτή η σκέψη μειώνει τα τυφλά λάθη.

Αν αποτύχει: Αν δεν μπορείς να απαντήσεις, τρέξε πρώτα `pwd`, `ls` και διάβασε README.

## Όρια του Termux

Πού βρίσκεσαι: Πριν περιμένεις να δουλέψει ένα project.

Τι κάνεις: Θυμήσου ότι χρειάζεται permissions, packages, σωστό folder και σωστό command.

Τι πρέπει να δεις: Δεν αντιμετωπίζεις κάθε error σαν καταστροφή.

Γιατί το κάνεις: Τα περισσότερα errors είναι λάθος path, missing dependency ή permission.

Αν αποτύχει: Αν κολλήσει, κράτα exact error και κάνε τον πρώτο μικρό έλεγχο.

# Πώς διαβάζεις μια εντολή πριν την τρέξεις

Κάθε command έχει μέρη. Μην το βλέπεις σαν μαγική φράση. Χώρισέ το σε πρόγραμμα, πράξη, επιλογές, αρχεία, paths και URLs.

```
git clone https://github.com/dedsec1121fk/DedSec
```

## Γραμμή προς γραμμή εξήγηση:

- git είναι το πρόγραμμα.
- clone είναι η πράξη: κατέβασε αντίγραφο repo.
- Το URL είναι η πηγή.
- Το αποτέλεσμα είναι ένας νέος φάκελος DedSec.

```
cd DedSec && bash Setup.sh
```

## Γραμμή προς γραμμή εξήγηση:

- cd DedSec αλλάζει folder.
- && σημαίνει συνέχισε μόνο αν το προηγούμενο πέτυχε.
- bash Setup.sh τρέχει το setup file.
- Αν το cd αποτύχει, το setup δεν τρέχει.

## Κανόνες πριν το Enter

Αν βλέπεις rm, άγνωστο URL, curl pipe bash, sudo ή path που δεν καταλαβαίνεις, σταμάτα και εξήγησε πρώτα την εντολή γραμμή προς γραμμή.

# Πιο αναλυτικά για Android APK και permissions

Η εγκατάσταση APK έχει στάδια: official source, download warning, άδεια unknown apps, installer, Play Protect, install result. Ο χρήστης πρέπει να ξέρει τι βλέπει σε κάθε οθόνη.

## Download warning

Πού βρίσκεσαι: Browser popup.

Τι κάνεις: Αν είσαι σε official link, πάτα Download/OK. Αν είσαι σε άγνωστο APK site, ακύρωσε.

Τι πρέπει να δεις: Το αρχείο κατεβαίνει και φαίνεται σε notification ή download list.

Γιατί το κάνεις: Το warning υπάρχει επειδή APK μπορεί να εγκαταστήσει app.

Αν αποτύχει: Αν δεν είσαι σίγουρος για την πηγή, μην συνεχίσεις.

## Allow from this source

Πού βρίσκεσαι: Android Settings μετά το prompt.

Τι κάνεις: Άνοιξε Allow from this source μόνο για τον browser ή My Files που άνοιξε το APK. Μετά πάτα Back.

Τι πρέπει να δεις: Επιστρέφεις στο installer με κουμπί Install.

Γιατί το κάνεις: Το Android δίνει άδεια εγκατάστασης ανά εφαρμογή.

Αν αποτύχει: Μετά την εγκατάσταση μπορείς να το κλείσεις ξανά.

## App not installed

Πού βρίσκεσαι: Android installer.

Τι κάνεις: Μην πατάς συνέχεια install. Σημείωσε το μήνυμα και σκέψου source conflict.

Τι πρέπει να δεις: Καταλαβαίνεις αν υπάρχει παλιό Termux από άλλη πηγή.

Γιατί το κάνεις: Τα Termux add-ons πρέπει να είναι από ίδια πηγή.

Αν αποτύχει: Απεγκατέστησε παλιά Termux/add-ons και βάλε όλα από F-Droid ή όλα από official GitHub release.

## Samsung example path

Settings -> Apps -> Special access -> Install unknown apps -> επιλέγεις browser ή My Files -> Allow from this source.

# Πιο χρήσιμες συνήθειες για καθημερινό Termux

Αυτές οι συνήθειες κάνουν τον χρήστη πιο σταθερό: καθαροί φάκελοι, logs, μικρές εντολές, README first, backup πριν πείραγμα.

```
mkdir -p ~/Termux-Projects ~/Termux-Course/notes ~/Termux-Course/backups
printf "Termux course log\n" > ~/Termux-Course/notes/log.txt
ls ~/Termux-Course
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει οργανωμένους φακέλους.
- Δημιουργεί αρχικό log.
- Ελέγχει τη δομή.
- Μην δουλεύεις απευθείας μέσα στα Downloads αν δεν χρειάζεται.
- Πριν τρέξεις script, διάβασε README.
- Πριν διαγράψεις, τρέξε ls και δες τι υπάρχει.
- Αν αποτύχει command, μην τρέχεις άλλα δέκα τυχαία. Διάβασε το πρώτο error.
- Κράτα screenshots ή αντιγραφή των τελευταίων γραμμών του error.

# Πριν ξεκινήσεις - κινητό, internet, ασφάλεια και σημειώσεις

Αυτή η προετοιμασία είναι σημαντική γιατί πολλά αρχάρια προβλήματα δεν είναι προβλήματα του Termux. Είναι ρυθμίσεις κινητού, storage permission, κακό internet, λάθος APK source ή εντολή που τρέχεις από λάθος φάκελο.

## Τι ετοιμάζεις πριν ανοίξεις Termux

Φόρτισε το κινητό πάνω από 40%, βάλε σταθερό Wi-Fi, άνοιξε μια εφαρμογή σημειώσεων και φτιάξε note με όνομα "Termux course errors". Κάθε φορά που κάτι αποτυγχάνει, γράψε εκεί το ακριβές error.

- Άνοιξε Android Settings και έλεγξε τον ελεύθερο χώρο. Κράτα τουλάχιστον 2 GB ελεύθερα για το μάθημα.
- Χρησιμοποίησε έναν σύγχρονο browser όπως Chrome, Samsung Internet, Firefox ή Brave. Ο browser έχει σημασία γιατί το Android δίνει άδεια εγκατάστασης APK ανά εφαρμογή.
- Κράτα το Google Play Protect ενεργό. Σκανάρει apps και μπορεί να εμφανίσει προειδοποίηση για APK εκτός Play Store.
- Μην ανακατεύεις Termux από Google Play με add-ons από F-Droid ή GitHub. Αυτό συχνά προκαλεί signature conflicts.

## Βήμα 1 - έλεγχος χώρου

Πού βρίσκεσαι: Android home screen -> Settings.

Τι κάνεις: Άνοιξε Settings, ψάξε Storage και δες πόσος χώρος είναι διαθέσιμος.

Τι πρέπει να δεις: Οθόνη storage που δείχνει used/free space.

Γιατί το κάνεις: Downloads, repositories, Python modules και exports χρειάζονται χώρο. Λίγος χώρος προκαλεί περίεργα errors.

Αν αποτύχει: Σβήσε άχρηστα αρχεία ή μετέφερε φωτογραφίες/βίντεο πριν την εγκατάσταση.

## Βήμα 2 - ετοίμασε σημειώσεις

Πού βρίσκεσαι: Οποιαδήποτε notes app στο κινητό.

Τι κάνεις: Φτιάξε note με όνομα Termux Course Log. Γράψε ημερομηνία και μοντέλο κινητού.

Τι πρέπει να δεις: Κενό note έτοιμο για commands και errors.

Γιατί το κάνεις: Χρειάζεσαι σημείο καταγραφής για να ξέρεις τι άλλαξε. Αυτό κάνει το troubleshooting εφικτό.

Αν αποτύχει: Αν δεν χρησιμοποιείς notes, κράτα screenshots των errors.

# Κατέβασμα και εγκατάσταση APK - ακριβή βήματα στο κινητό

APK είναι το αρχείο εγκατάστασης μιας Android εφαρμογής. Όταν εγκαθιστάς APK εκτός Play Store, αυτό λέγεται sideloading. Είναι φυσιολογικό για open-source εργαλεία, αλλά πρέπει να γίνεται προσεκτικά.

## Official links

<https://termux.dev/en/>

<https://f-droid.org/en/packages/com.termux/>

<https://github.com/termux/termux-app/releases>

<https://support.google.com/googleplay/answer/2812853>

<https://www.samsung.com/ae/support/mobile-devices/how-to-enable-permission-to-install-apps-from-unknown-source-on-my-samsung-phone/>

## Προτεινόμενη σειρά πηγών

Χρησιμοποίησε F-Droid ή το official Termux GitHub releases. Μην χρησιμοποιείς παλιά Play Store builds για αυτό το μάθημα. Αν βάλεις Termux, Termux:API και Termux:Styling, βάλε τα από την ίδια οικογένεια πηγής όταν γίνεται.

## Βήμα 1 - άνοιξε official source

Πού βρίσκεσαι: Android browser.

Τι κάνεις: Πήγαινε στο <https://f-droid.org/en/packages/com.termux/> ή στο <https://github.com/termux/termux-app/releases>.

Τι πρέπει να δεις: Σελίδα που γράφει καθαρά Termux και ανήκει σε F-Droid ή github.com/termux.

Γιατί το κάνεις: Ελέγχεις την πηγή πριν εγκαταστήσεις κώδικα στο κινητό.

Αν αποτύχει: Αν είναι ad mirror ή τυχαίο APK site, κλείσ' το.

## Βήμα 2 - κατέβασε το APK

Πού βρίσκεσαι: F-Droid Termux page ή GitHub Releases page.

Τι κάνεις: Στο F-Droid, κατέβα σε version και πάτα Download APK. Στο GitHub Releases, άνοιξε Assets και πάτα το APK file.

Τι πρέπει να δεις: Ο browser δείχνει warning τύπου "This type of file can harm your device" ή download progress.

Γιατί το κάνεις: Το Android προειδοποιεί επειδή APK μπορεί να εγκαταστήσει app. Είναι αναμενόμενο ακόμη και για σωστό APK.

Αν αποτύχει: Αν βλέπεις πολλά APK και δεν ξέρεις ποιο, ξεκίνα από F-Droid Download APK.

### Βήμα 3 - άνοιξε το κατεβασμένο APK

Πού βρίσκεσαι: Download bar του browser ή Android Downloads/My Files.

Τι κάνεις: Πάτα το .apk αρχείο.

Τι πρέπει να δεις: Το Android ανοίγει install screen ή ζητά permission για unknown apps.

Γιατί το κάνεις: Το κινητό ρωτά αν ο συγκεκριμένος browser/file manager επιτρέπεται να εγκαθιστά apps.

Αν αποτύχει: Αν δεν ανοίγει, πήγαινε My Files -> Downloads και πάτα το APK από εκεί.

### Βήμα 4 - άδεια unknown apps μόνο για αυτή την πηγή

Πού βρίσκεσαι: Android settings screen που άνοιξε από το prompt.

Τι κάνεις: Άνοιξε Allow from this source για τον browser ή file manager που χρησιμοποίησες. Μετά πάτα Back.

Τι πρέπει να δεις: Επιστρέφεις στην οθόνη εγκατάστασης με κουμπί Install.

Γιατί το κάνεις: Το Android δίνει άδεια ανά εφαρμογή, όχι γενικά σε όλο το κινητό.

Αν αποτύχει: Μετά την εγκατάσταση μπορείς να γυρίσεις εδώ και να το κλείσεις.

### Βήμα 5 - εγκατάσταση και έλεγχος

Πού βρίσκεσαι: APK install screen.

Τι κάνεις: Πάτα Install. Περίμενε. Πάτα Open μόνο όταν γράψει App installed.

Τι πρέπει να δεις: Το Termux ανοίγει και δείχνει μαύρη terminal οθόνη με prompt.

Γιατί το κάνεις: Η εφαρμογή εγκαταστάθηκε. Το prompt δείχνει ότι μπορεί να ξεκινήσει.

Αν αποτύχει: Αν γράψει App not installed, κάνε uninstall παλιό Termux/add-ons από άλλες πηγές και βάλε όλα από μία πηγή.

### Κανόνες ασφαλείας

To Allow from this source δεν είναι μόνιμη συνήθεια. Είναι προσωρινή άδεια για έμπιστη πηγή. Μετά την εγκατάσταση πήγαινε Settings -> Apps -> Special app access -> Install unknown apps και κλείσε την άδεια αν δεν τη χρειάζεσαι.

# Πρώτο άνοιγμα Termux - τι σημαίνει η οθόνη

Η πρώτη οθόνη του Termux μπορεί να φαίνεται άδεια ή περίεργη. Στην πραγματικότητα περιμένει εντολές. Το σημαντικό σημείο είναι το prompt: εκεί πληκτρολογείς.

## Βήμα 1 - βρες το prompt

Πού βρίσκεσαι: Μέσα στο Termux μετά το άνοιγμα.  
Τι κάνεις: Κοίτα την τελευταία γραμμή. Μπορεί να τελειώνει με \$ ή να δείχνει path όπως ~ \$.  
Τι πρέπει να δεις: Ένας cursor που αναβοσβήνει μετά το prompt.  
Γιατί το κάνεις: Ο cursor σημαίνει ότι το Termux είναι έτοιμο.  
Αν αποτύχει: Αν ακόμη κινείται κείμενο εγκατάστασης, περίμενε να σταματήσει.

## Βήμα 2 - δοκίμασε απλή είσοδο

Πού βρίσκεσαι: Termux prompt.  
Τι κάνεις: Γράψε echo hello και πάτα Enter.  
Τι πρέπει να δεις: Η επόμενη γραμμή γράφει hello και μετά εμφανίζεται νέο prompt.  
Γιατί το κάνεις: Επιβεβαιώνεις ότι keyboard και Enter δουλεύουν.  
Αν αποτύχει: Αν το keyboard κρύβει το prompt, γύρνα το κινητό ή χρησιμοποίησε terminal-friendly keyboard.

## Βήμα 3 - κατάλαβε το output

Πού βρίσκεσαι: Termux prompt.  
Τι κάνεις: Τρέξε pwd.  
Τι πρέπει να δεις: Path όπως /data/data/com.termux/files/home.  
Γιατί το κάνεις: pwd σημαίνει print working directory και δείχνει πού βρίσκεσαι.  
Αν αποτύχει: Αν περιμένεις Downloads, δεν είσαι εκεί ακόμα. Πρέπει να κάνεις cd στο σωστό path.

```
echo hello
```

### Γραμμή προς γραμμή εξήγηση:

- echo εκτυπώνει κείμενο στην οθόνη.
- hello είναι το κείμενο που ζήτησες να εμφανιστεί.
- Η εντολή είναι ασφαλής γιατί δεν αλλάζει αρχεία.

```
pwd
```

### Γραμμή προς γραμμή εξήγηση:

- pwd σημαίνει print working directory.
- Το αποτέλεσμα είναι ο τρέχων φάκελος.
- Όταν μια εντολή δεν βρίσκει αρχείο, το pwd είναι από τους πρώτους ελέγχους.

# Storage access - Downloads, shared storage και paths

Οι Android εφαρμογές δεν βλέπουν αυτόματα όλα τα αρχεία του κινητού. Το Termux έχει δικό του private home folder. Για να δουλέψεις με Downloads πρέπει να ζητήσεις storage permission.

## Official links

<https://wiki.termux.com/wiki/Termux-setup-storage>

```
termux-setup-storage
```

## Γραμμή προς γραμμή εξήγηση:

- Ζητά από το Android άδεια πρόσβασης σε shared storage.
- Πρέπει να εμφανιστεί permission popup.
- Μετά την αποδοχή, το Termux δημιουργεί shortcuts μέσα στο ~/storage.

### Βήμα 1 - ζήτα storage permission

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Γράψε termux-setup-storage και πάτα Enter.

Τι πρέπει να δεις: Το Android ζητά να επιτρέψεις σε Termux πρόσβαση σε files/media/storage.

Γιατί το κάνεις: Χωρίς αυτό, scripts δεν σώζουν εύκολα σε Downloads ούτε διαβάζουν αρχεία από εκεί.

Αν αποτύχει: Αν δεν εμφανιστεί popup, πήγαινε Settings -> Apps -> Termux -> Permissions και δώσε άδεια.

### Βήμα 2 - έλεγξε shortcuts

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Τρέξε ls ~/storage.

Τι πρέπει να δεις: Θα δεις folders όπως downloads, shared, dcim, pictures.

Γιατί το κάνεις: Τα shortcuts είναι πιο εύκολα από τα μακριά Android paths.

Αν αποτύχει: Αν δεν υπάρχει ~/storage, ξανατρέξε termux-setup-storage και κλείσε/άνοιξε Termux.

### Βήμα 3 - κατάλαβε το πραγματικό Downloads path

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Τρέξε ls /storage/emulated/0/Download.

Τι πρέπει να δεις: Θα δεις αρχεία από τον φάκελο Downloads του κινητού.

Γιατί το κάνεις: Κάποια scripts χρησιμοποιούν το πραγματικό Android path. Πρέπει να το αναγνωρίζεις.

Αν αποτύχει: Αν εμφανιστεί permission denied, δεν έχει εγκριθεί το storage permission.

# Package management - update, upgrade, install, search

Το Termux χρησιμοποιεί packages. Package είναι έτοιμο εργαλείο που το Termux ξέρει πώς να κατεβάσει και να εγκαταστήσει. Η εντολή pkg είναι ο φιλικός τρόπος διαχείρισης packages στο Termux.

## Official links

[https://wiki.termux.com/wiki/Package\\_Management](https://wiki.termux.com/wiki/Package_Management)

<https://github.com/termux/termux-packages/wiki/package-management>

```
pkg update
```

### Γραμμή προς γραμμή εξήγηση:

- Κατεβάζει τις πιο νέες λίστες πακέτων.
- Δεν εγκαθιστά τα πάντα από μόνο του.
- Λέει στο Termux ποιες εκδόσεις υπάρχουν.

```
pkg upgrade
```

### Γραμμή προς γραμμή εξήγηση:

- Εγκαθιστά διαθέσιμα upgrades για packages που ήδη έχεις.
- Μπορεί να ζητήσει επιβεβαίωση με Y ή N.
- Διάβασε το prompt πριν πατήσεις Enter.

```
pkg install git nano python
```

### Γραμμή προς γραμμή εξήγηση:

- Εγκαθιστά Git, Nano και Python.
- Το git κατεβάζει repositories.
- Το nano επεξεργάζεται αρχεία.
- Το python τρέχει Python scripts.

# Πλοήγηση - folders, files, κενά και ασφαλές σβήσιμο

Τα περισσότερα αρχάρια terminal λάθη είναι λάθη τοποθεσίας. Το terminal ενεργεί στον τρέχοντα φάκελο εκτός αν του δώσεις πλήρες path.

```
pwd
ls
cd folder-name
cd ..
```

## Γραμμή προς γραμμή εξήγηση:

- pwd δείχνει πού είσαι.
- ls δείχνει τι υπάρχει μέσα στον τρέχοντα φάκελο.
- cd folder-name μπαίνει σε φάκελο.
- cd .. γυρίζει έναν φάκελο πίσω.

```
mkdir -p ~/Termux-Projects
cd ~/Termux-Projects
pwd
```

## Γραμμή προς γραμμή εξήγηση:

- mkdir δημιουργεί φάκελο.
- -p σημαίνει μην παραπονιέσαι αν ο φάκελος ήδη υπάρχει.
- cd μπαίνει στον φάκελο εξάσκησης.
- pwd επιβεβαιώνει πού βρέθηκες.

## Κενά σε ονόματα αρχείων

Αν αρχείο ή φάκελος έχει κενά, βάλ' το σε εισαγωγικά. Παράδειγμα: `python "Corrupted Files.py"`. Χωρίς εισαγωγικά, το shell νομίζει ότι κάθε λέξη είναι ξεχωριστό argument.

## Μην τρέχεις destructive commands στα τυφλά

Μην μαθαίνεις με `rm -rf`. Αν δεν καταλαβαίνεις το target path, μην τρέξεις την εντολή. Το σβήσιμο στο Termux δεν είναι παιχνίδι.

# Nano editor - άνοιγμα, γράψιμο, save, exit

Το nano είναι απλός terminal editor. Δεν είναι εντυπωσιακό, αλλά είναι προβλέψιμο. Κάτω στην οθόνη δείχνει shortcuts. Το σύμβολο ^ σημαίνει Ctrl.

```
nano notes.txt
```

## Γραμμή προς γραμμή εξήγηση:

- Ανοίγει το notes.txt.
- Αν δεν υπάρχει, το δημιουργεί όταν κάνεις save.
- Γράφεις κανονικό κείμενο στην περιοχή editor.

### Βήμα 1 - δημιουργήσε note

Πού βρίσκεσαι: Μέσα στον course φάκελο.

Τι κάνεις: Τρέξε nano notes.txt. Γράψε μία πρόταση για το τι έμαθες.

Τι πρέπει να δεις: Ανοίγει το nano και μπορείς να γράψεις.

Γιατί το κάνεις: Η επεξεργασία αρχείων χρειάζεται για scripts, notes και configs.

Αν αποτύχει: Αν λείπει το nano, τρέξε pkg install nano.

### Βήμα 2 - save

Πού βρίσκεσαι: Μέσα στο nano.

Τι κάνεις: Πάτα Ctrl+O και μετά Enter.

Τι πρέπει να δεις: Το nano γράφει το αρχείο και μένει ανοιχτό.

Γιατί το κάνεις: Ctrl+O σημαίνει Write Out, δηλαδή save.

Αν αποτύχει: Αν δεν έχεις Ctrl, ενεργοποίησε extra keys στο Termux ή χρησιμοποίησε άλλη keyboard app.

### Βήμα 3 - exit και έλεγχος

Πού βρίσκεσαι: Μέσα στο nano.

Τι κάνεις: Πάτα Ctrl+X. Μετά τρέξε cat notes.txt.

Τι πρέπει να δεις: Η πρότασή σου εμφανίζεται στο terminal.

Γιατί το κάνεις: Αποδεικνύεις ότι το αρχείο σώθηκε πριν συνεχίσεις.

Αν αποτύχει: Αν το cat λέει missing file, σώθηκε αλλού ή δεν πάτησες Enter στο save.

# GitHub από browser και από Termux

Το GitHub είναι χώρος όπου αποθηκεύονται projects. Μπορείς να το χρησιμοποιήσεις με δύο τρόπους: Download ZIP από browser ή git clone μέσα από Termux.

## Official links

<https://docs.github.com/en/repositories/creating-and-managing-repositories/cloning-a-repository>

<https://docs.github.com/en/authentication/connecting-to-github-with-ssh>

### Browser method - Download ZIP

Πού βρίσκεσαι: Android browser σε GitHub repository page.

Τι κάνεις: Πάτα Code, μετά Download ZIP. Περίμενε να εμφανιστεί το αρχείο στα Downloads.

Τι πρέπει να δεις: Ένα .zip αρχείο στα Downloads.

Γιατί το κάνεις: Είναι ο πιο εύκολος τρόπος όταν θες μόνο τα αρχεία.

Αν αποτύχει: Αν δεν φαίνεται το Code button, γύρνα το κινητό ή βάλε desktop site.

### Termux method - clone

Πού βρίσκεσαι: Termux μέσα στο ~/Termux-Projects.

Τι κάνεις: Τρέξε git clone και το repository URL.

Τι πρέπει να δεις: Εμφανίζεται νέος φάκελος με το όνομα του repo.

Γιατί το κάνεις: Το clone κρατά Git history και επιτρέπει update με git pull.

Αν αποτύχει: Αν λείπει το git, τρέξε pkg install git.

```
mkdir -p ~/Termux-Projects
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/DedSec.git
ls
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει καθαρό φάκελο projects.
- Μπαίνει σε αυτόν.
- Κατεβάζει το DedSec repository.
- Δείχνει τον φάκελο για επιβεβαίωση.

# Τρέξιμο Python scripts με ασφάλεια

Ένα Python script είναι αρχείο κειμένου που το Python εκτελεί γραμμή προς γραμμή. Πριν τρέξεις script από project, διάβασε README, δες τα αρχεία και βρες το main file.

## Official links

<https://docs.python.org/3/>

<https://docs.python.org/3/library/venv.html>

<https://pip.pypa.io/en/stable/>

```
python --version
python script.py
python "File With Spaces.py"
```

## Γραμμή προς γραμμή εξήγηση:

- `python --version` επιβεβαιώνει ότι υπάρχει Python.
- `python script.py` τρέχει script στον τρέχοντα φάκελο.
- Τα εισαγωγικά χρειάζονται όταν το αρχείο έχει κενά.

## Κανόνες requirements

Αν project έχει requirements.txt, διάβασέ το πριν εγκαταστήσεις random modules. Στο Termux κάποια Python packages χρειάζονται πρώτα system packages.

# Project 1 - DedSec Project: τι κάνει και πώς το μελετάς

Το DedSec Project είναι εκπαιδευτικό toolkit για Android + Termux. Περιλαμβάνει scripts, utilities, local web interfaces, games, settings/configuration tools και δίγλωσση τεκμηρίωση English/Greek.

## Official links

<https://github.com/dedsec1121fk/DedSec>

<https://ded-sec.space/>

### Τι μαθαίνει ο χρήστης από αυτό

Δομή repository, οργάνωση scripts, setup commands, local web UIs, διάβασμα README, ασφαλής επιθεώρηση project, updates με git pull και πώς οργανώνεται ένα μεγάλο Termux toolkit.

```
mkdir -p ~/Termux-Projects
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/DedSec.git
cd DedSec
ls
```

### Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει φάκελο learning projects.
- Κατεβάζει το repository.
- Μπαίνει στον φάκελο DedSec.
- Δείχνει τα αρχεία πριν τρέξεις οτιδήποτε.

### Βήμα 1 - διάβασε README

Πού βρίσκεσαι: Μέσα στο ~/Termux-Projects/DedSec.

Τι κάνεις: Άνοιξε README.md στο GitHub ή τρέξε less README.md αν υπάρχει.

Τι πρέπει να δεις: Βλέπεις installation sections, categories και disclaimers.

Γιατί το κάνεις: Το README είναι ο χάρτης του project. Το να τρέξεις random files πρώτα είναι λάθος.

Αν αποτύχει: Αν το less λείπει, χρησιμοποίησε cat README.md για λίγο κείμενο ή άνοιξε GitHub στον browser.

### Βήμα 2 - βρες categories

Πού βρίσκεσαι: Repo folder ή GitHub page.

Τι κάνεις: Ψάξε folders όπως Scripts, Games, Developer Base ή τις κατηγορίες που γράφει το README.

Τι πρέπει να δεις: Καταλαβαίνεις ότι είναι συλλογή και όχι μία εντολή.

Γιατί το κάνεις: Μεγάλα repositories τα πλοηγείσαι, δεν τα μαντεύεις.

Αν αποτύχει: Αν τα ονόματα διαφέρουν, εμπιστεύσου το τωρινό repo listing.

## Project 2 - Offline Survival Project: offline βάση γνώσης

Το Offline Survival Project είναι δίγλωσσο offline-first project γνώσης για Android + Termux. Συνδυάζει JSON knowledge database, terminal app, local browser interface, TXT export tools και update logs.

### Official links

<https://github.com/dedsec1121fk/Offline-Survival-Project>

#### Τι μαθαίνει ο χρήστης από αυτό

Πώς δουλεύουν data folders, πώς Python διαβάζει JSON databases, πώς μια offline εφαρμογή ψάχνει γνώση, πώς σώζονται exports και γιατί το storage permission έχει σημασία.

```
mkdir -p ~/Termux-Projects
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/Offline-Survival-Project.git
cd Offline-Survival-Project
ls
```

#### Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει/χρησιμοποιεί project area.
- Κάνει clone το project.
- Μπαίνει στον φάκελο.
- Δείχνει database, script και support files.

```
python "Offline Survival.py"
```

#### Γραμμή προς γραμμή εξήγηση:

- Τρέχει το main script.
- Τα εισαγωγικά προστατεύουν το κενό στο όνομα αρχείου.
- Το αναμενόμενο αποτέλεσμα είναι menu/interface που περιγράφει το project.

# Project 3 - Corrupted Files Project: offline archive υποθέσεων

To Corrupted Files Project είναι δίγλωσσο offline incident archive για Ελλάδα και ΗΠΑ. Οργανώνει public-interest incidents, institutional-failure cases, disasters, corruption-related files και historical dossiers σε folders, JSON records, indexes και media references.

## Official links

<https://github.com/dedsec1121fk/Corrupted-Files-Project>

### Τι μαθαίνει ο χρήστης από αυτό

Πώς οργανώνονται structured archives, πώς λειτουργεί country/year/incident structure, πώς βοηθούν τα indexes και πώς ένα Python reader μπορεί να πλοηγεί dataset.

```
mkdir -p ~/Termux-Projects
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/Corrupted-Files-Project.git
cd Corrupted-Files-Project
ls
```

### Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει/χρησιμοποιεί φάκελο projects.
- Κατεβάζει το archive.
- Μπαίνει στο project.
- Δείχνει country folders, indexes και main script.

```
python "Corrupted Files.py"
```

### Γραμμή προς γραμμή εξήγηση:

- Τρέχει το main reader script.
- Τα εισαγωγικά είναι απαραίτητα γιατί το όνομα έχει κενό.
- Το αναμενόμενο αποτέλεσμα είναι terminal menu/reader.

# Επαληθευμένα links για projects και README οδηγίες

Αυτή η ενότητα μαζεύει σε ένα σημείο τα links των τριών projects και τα verified README links με τις οδηγίες εγκατάστασης/χρήσης. Πρώτα ανοίγεις το repository, μετά το README, μετά το install/use section, και μόνο μετά τρέχεις εντολές στο Termux.

## Official links

DedSec Project repository: <https://github.com/dedsec1121fk/DedSec>

DedSec Project README: <https://github.com/dedsec1121fk/DedSec/blob/main/README.md>

DedSec README file - scroll to "How To Install And Setup The DedSec Project":

<https://github.com/dedsec1121fk/DedSec/blob/main/README.md>

Offline Survival repository: <https://github.com/dedsec1121fk/Offline-Survival-Project>

Offline Survival README: <https://github.com/dedsec1121fk/Offline-Survival-Project/blob/main/README.md>

Offline Survival README file - scroll to "How To Install And Run Offline Survival":

<https://github.com/dedsec1121fk/Offline-Survival-Project/blob/main/README.md>

Corrupted Files repository: <https://github.com/dedsec1121fk/Corrupted-Files-Project>

Corrupted Files README: <https://github.com/dedsec1121fk/Corrupted-Files-Project/blob/main/README.md>

Corrupted Files README file - scroll to "How To Download":

<https://github.com/dedsec1121fk/Corrupted-Files-Project/blob/main/README.md>

## Σημαντικό

Αν το README στο GitHub έχει αλλάξει, εμπιστεύσου το current README. Το βιβλίο δείχνει τη σωστή μέθοδο: official repo -> README -> install/use guide -> Termux command.

# DedSec Project - README install guide με ακριβή βήματα

Το README του DedSec Project περιγράφει first-time full install. Εδώ το βάζουμε σε απλά, πρακτικά βήματα για κινητό και Termux.

## Official links

[README file - installation section inside this README:](#)  
<https://github.com/dedsec1121fk/DedSec/blob/main/README.md>

### Βήμα 1 - F-Droid και Termux

Πού βρίσκεσαι: Android browser και F-Droid.

Τι κάνεις: Κατέβασε F-Droid, άνοιξέ το, ψάξε Termux και εγκατέστησέ το. Αν βάλεις Termux:API ή Termux:Styling, βάλε τα από την ίδια πηγή.

Τι πρέπει να δεις: Το Termux υπάρχει στο app drawer και ανοίγει χωρίς App not installed.

Γιατί το κάνεις: Το README προτείνει F-Droid για καλύτερη συμβατότητα. Η ίδια πηγή μειώνει signature conflicts.

Αν αποτύχει: Αν βγει App not installed, απεγκατέστησε παλιά Termux/add-ons από άλλη πηγή και ξαναβάλε όλα από μία πηγή.

### Βήμα 2 - άνοιξε Termux πριν κάνεις paste

Πού βρίσκεσαι: Termux app.

Τι κάνεις: Άνοιξε Termux και περίμενε να σταματήσει το αρχικό κείμενο.

Τι πρέπει να δεις: Βλέπεις prompt με cursor, συνήθως κάτι σαν ~ \$.

Γιατί το κάνεις: Οι εντολές πρέπει να τρέξουν μέσα στο Termux, όχι στον browser.

Αν αποτύχει: Αν δεν εμφανίζεται πληκτρολόγιο, πάτα στο μαύρο terminal area.

### Βήμα 3 - prepare packages

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Τρέξε την εντολή που δίνει το README για update/upgrade/git/nano/storage.

Τι πρέπει να δεις: Βλέπεις download/update messages και storage permission popup.

Γιατί το κάνεις: Έτσι ετοιμάζεται το Termux να κατεβάσει repo, να ανοίγει αρχεία και να χρησιμοποιεί storage.

Αν αποτύχει: Αν storage popup δεν εμφανιστεί, πήγαινε Settings -> Apps -> Termux -> Permissions και έλεγξε Files/Storage.

### Βήμα 4 - clone και setup

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Κάνε clone το repo, μπες στον φάκελο DedSec και τρέξε bash Setup.sh.

Τι πρέπει να δεις: Το setup/menu ξεκινά ή εμφανίζει οδηγίες.

Γιατί το κάνεις: Το Setup.sh είναι το documented entry point του README.

Αν αποτύχει: Αν λέει No such file, τρέξε pwd και ls. Μάλλον είσαι σε λάθος φάκελο.

```
pkg update -y  
pkg upgrade -y  
pkg install git nano -y  
termux-setup-storage
```

### **Γραμμή προς γραμμή εξήγηση:**

- Ανανεώνει Termux packages.
- Εγκαθιστά git και nano.
- Ζητά storage access.

```
git clone https://github.com/dedsec1121fk/DedSec  
cd DedSec  
bash Setup.sh
```

### **Γραμμή προς γραμμή εξήγηση:**

- Κατεβάζει το DedSec Project.
- Μπαίνει στον project folder.
- Ξεκινά το setup που αναφέρει το README.

# Offline Survival Project - README install/run guide με ακριβή βήματα

Το README λέει ότι το project είναι offline-first knowledge base με JSON database, terminal app, browser interface, TXT exports και update logs. Για να δουλέψει, πρέπει να υπάρχουν το main Python file και ο database folder.

## Official links

[README file - install/run section inside this README:](#)

<https://github.com/dedsec1121fk/Offline-Survival-Project/blob/main/README.md>

### Βήμα 1 - κατέβασμα project

Πού βρίσκεσαι: GitHub page ή Termux.

Τι κάνεις: Χρησιμοποίησε Code -> Download ZIP και extract ή git clone από Termux.

Τι πρέπει να δεις: Έχεις folder Offline-Survival-Project ή Offline-Survival-Project-main.

Γιατί το κάνεις: Χωρίς local files δεν υπάρχει offline database για να διαβάσει το app.

Αν αποτύχει: Αν το ZIP δεν ανοίγει, ξανακατέβασέ το με σταθερό internet.

### Βήμα 2 - storage για exports

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Τρέξε termux-setup-storage αν θέλεις exports/Downloads access.

Τι πρέπει να δεις: Το Android ζητά άδεια και δημιουργείται ~/storage.

Γιατί το κάνεις: Το README αναφέρει storage access για exports και files στα Downloads.

Αν αποτύχει: Αν Permission denied, έλεγξε Android permissions για Termux.

### Βήμα 3 - μπες στον σωστό φάκελο

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Τρέξε cd με το πραγματικό folder name και μετά ls.

Τι πρέπει να δεις: Βλέπεις Offline Survival.py και database folder.

Γιατί το κάνεις: Πριν τρέξεις Python πρέπει να είσαι στον φάκελο που έχει το script.

Αν αποτύχει: Αν ls δεν δείχνει το script, είσαι σε λάθος path.

### Βήμα 4 - τρέξε το main script

Πού βρίσκεσαι: Μέσα στον project folder.

Τι κάνεις: Τρέξε python "Offline Survival.py" ή python3 "Offline Survival.py" αν χρειάζεται.

Τι πρέπει να δεις: Εμφανίζεται menu/app output.

Γιατί το κάνεις: Τα εισαγωγικά χρειάζονται επειδή το filename έχει κενό.

Αν αποτύχει: Αν ModuleNotFoundError, γράψε ακριβώς το module και έλεγξε README/requirements.

## Βήμα 5 - browser UI

Πού βρίσκεσαι: Μέσα από το terminal menu.

Τι κάνεις: Αν διαλέξεις browser option, άνοιξε `http://127.0.0.1:8765/` στον browser.

Τι πρέπει να δεις: Ανοίγει local interface στο ίδιο κινητό.

Γιατί το κάνεις: 127.0.0.1 σημαίνει local phone, όχι δημόσιο site.

Αν αποτύχει: Αν δεν ανοίγει, κράτα το Termux process ανοιχτό και έλεγξε αν το port είναι 8765 ή άλλο.

```
termux-setup-storage  
cd ~/Offline-Survival-Project-main  
python "Offline Survival.py"
```

### Γραμμή προς γραμμή εξήγηση:

- Ζητά storage access.
- Μπαίνει στον extracted project folder.
- Τρέχει το main script με quotes.

```
cd ~/Offline-Survival-Project-main  
python3 "Offline Survival.py"
```

### Γραμμή προς γραμμή εξήγηση:

- Εναλλακτική μόνο αν χρειάζεται python3.
- Κάνει το ίδιο launch με διαφορετικό Python command.

# Corrupted Files Project - README download/use guide με ακριβή βήματα

Το README περιγράφει το project ως bilingual offline incident archive για Ελλάδα και ΗΠΑ. Μπορείς να το χρησιμοποιήσεις με manual browsing ή με το main Python script.

## Official links

[README file - download section inside this README:](#)

<https://github.com/dedsec1121fk/Corrupted-Files-Project/blob/main/README.md>

[README: https://github.com/dedsec1121fk/Corrupted-Files-Project/blob/main/README.md](https://github.com/dedsec1121fk/Corrupted-Files-Project/blob/main/README.md)

### Βήμα 1 - download από browser

Πού βρίσκεσαι: GitHub repository page.

Τι κάνεις: Πάτα Code -> Download ZIP και κάνε extract όπου θέλεις.

Τι πρέπει να δεις: Βλέπεις folder Corrupted-Files-Project-main.

Γιατί το κάνεις: Είναι ο πιο απλός τρόπος αν δεν θες Git commands.

Αν αποτύχει: Αν το ZIP είναι μισό, ξανακατέβασέ το.

### Βήμα 2 - ή clone με Termux

Πού βρίσκεσαι: Termux μέσα στο ~/Termux-Projects.

Τι κάνεις: Τρέξε git clone με το official URL και μπες στον φάκελο.

Τι πρέπει να δεις: Βλέπεις Greece, USA, database/tools ή root index files.

Γιατί το κάνεις: Το clone κάνει πιο εύκολο update αργότερα.

Αν αποτύχει: Αν git λείπει, τρέξε pkg install git.

### Βήμα 3 - manual browsing

Πού βρίσκεσαι: File manager ή Termux.

Τι κάνεις: Άνοιξε Greece ή USA, διάλεξε year folder και μετά incident folder.

Τι πρέπει να δεις: Βλέπεις paired EN/EL files, JSON records, metadata και media references.

Γιατί το κάνεις: Το manual browsing σε μαθαίνει τη δομή πριν χρησιμοποιήσεις reader.

Αν αποτύχει: Αν λείπουν folders, extraction/clone δεν ολοκληρώθηκε.

### Βήμα 4 - run reader

Πού βρίσκεσαι: Termux μέσα στον project folder.

Τι κάνεις: Τρέξε python "Corrupted Files.py".

Τι πρέπει να δεις: Εμφανίζεται guided terminal reader/menu.

Γιατί το κάνεις: Το script βοηθά όταν το archive είναι μεγάλο.

Αν αποτύχει: Αν λείπει cannot open file, έλεγξε pwd, ls και quotes.

## Βήμα 5 - validation tools αν υπάρχουν

Πού βρίσκεσαι: Termux μέσα στον project folder.

Τι κάνεις: Αν υπάρχουν τα files που αναφέρει το README, τρέξε validate ή rebuild indexes.

Τι πρέπει να δεις: Βλέπεις validation/rebuild output.

Γιατί το κάνεις: Χρήσιμο για έλεγχο structure και indexes.

Αν αποτύχει: Αν το αρχείο δεν υπάρχει, μην το εφεύρεις. Έλεγξε current README και ls.

```
git clone https://github.com/dedsec1121fk/Corrupted-Files-Project.git
cd Corrupted-Files-Project
python "Corrupted Files.py"
```

### Γραμμή προς γραμμή εξήγηση:

- Κατεβάζει το project.
- Μπαίνει στον φάκελο.
- Τρέχει το main reader με quotes.

```
python "Corrupted Files Tools/validate_rebuilt_structure.py"
python "Corrupted Files Tools/rebuild_indexes.py"
```

### Γραμμή προς γραμμή εξήγηση:

- Τρέξε τα μόνο αν υπάρχουν αυτά τα files.
- Το πρώτο ελέγχει structure.
- Το δεύτερο ξαναφτιάχνει indexes.

# Καθημερινό workflow - η ίδια ρουτίνα για κάθε project

Ο ασφαλέστερος τρόπος να μάθεις Termux είναι να επαναλαμβάνεις την ίδια ρουτίνα ελέγχου για κάθε νέο φάκελο. Αυτή η ρουτίνα αποτρέπει τα περισσότερα αρχάρια λάθη.

```
pwd
ls
ls -la
cat README.md
python --version
```

## Γραμμή προς γραμμή εξήγηση:

- Επιβεβαίωση τοποθεσίας.
- Λίστα ορατών αρχείων.
- Λίστα hidden files και details.
- Διάβασμα README αν είναι αρκετά μικρό.
- Έλεγχος ότι υπάρχει Python πριν τρέξεις scripts.

## Project inspection checklist

1. Πώς λέγεται το project; 2. Ποιο είναι το main file; 3. Θέλει storage permission; 4. Θέλει internet μετά το setup; 5. Πού σώζει output; 6. Ποια εντολή το ξεκινά; 7. Πώς μοιάζει ένα επιτυχημένο start;

# Terminal troubleshooting - τι σημαίνουν τα errors

Χρησιμοποίησε αυτή την ενότητα σαν λεξικό. Βρες την ακριβή φράση, διάβασε τι σημαίνει και κάνε το πρώτο ασφαλές fix.

## No such file or directory

Σημασία: Δεν είσαι στον σωστό φάκελο ή το όνομα αρχείου γράφτηκε διαφορετικά.  
Πρώτο fix: Τρέξε `pwd` και μετά `ls`. Αν το αρχείο έχει κενά, βάλε εισαγωγικά: `python "Offline Survival.py"`.

## Permission denied

Σημασία: Το αρχείο υπάρχει αλλά δεν μπορεί να εκτελεστεί, ή λείπει άδεια storage.  
Πρώτο fix: Τρέξε `bash script.sh` αντί για `./script.sh`, ή `termux-setup-storage` και δώσε άδεια.

## command not found

Σημασία: Δεν υπάρχει εγκατεστημένη η εντολή ή γράφτηκε λάθος.  
Πρώτο fix: Τρέξε `pkg search` όνομα και μετά `pkg install` πακέτο. Για Python modules χρησιμοποίησε `pip install module`.

## Unable to locate package

Σημασία: Λάθος όνομα πακέτου, παλιά repositories ή χαλασμένος mirror.  
Πρώτο fix: Τρέξε `pkg update`. Μετά `pkg search keyword`. Μην μαντεύεις ονόματα.

## repository is under maintenance or down

Σημασία: Ο mirror μπορεί να έχει προσωρινό πρόβλημα.  
Πρώτο fix: Τρέξε `termux-change-repo`, διάλεξε Main repository, άλλαξε mirror και μετά `pkg update`.

## Hash Sum mismatch

Σημασία: Οι package lists δεν ταιριάζουν ή κόπηκε το internet.  
Πρώτο fix: Τρέξε `pkg clean`, `pkg update` και ξαναδοκίμασε με σταθερό Wi-Fi.

## pkg update κολλάει

Σημασία: Πρόβλημα δικτύου, mirror, VPN ή DNS.  
Πρώτο fix: Κλείσε προσωρινά VPN, άλλαξε Wi-Fi/mobile data ή άλλαξε repo mirror.

## ModuleNotFoundError

Σημασία: Το Python τρέχει, αλλά λείπει βιβλιοθήκη που ζητά το script.  
Πρώτο fix: Διάβασε ακριβώς το όνομα module. Βάλε `pip install module-name` ή δες το README requirements.

## SyntaxError

Σημασία: Το Python αρχείο είναι σπασμένο, μισό, λάθος αντιγραμμένο ή λάθος πειραγμένο.  
Πρώτο fix: Μην μαντεύεις. Ξανακατέβασε το αρχείο, δες το line number και ψάξε για quotes/brackets.

## ImportError από pip package

Σημασία: Υπάρχει package αλλά λάθος version ή δεν ταιριάζει με Termux.  
Πρώτο fix: Κάνε upgrade μόνο αν το λέει το README. Δοκίμασε `pip install --upgrade package`. Στο Termux προτίμησε `pkg` όταν υπάρχει.

## git: not found

Σημασία: Δεν είναι εγκατεστημένο το Git.  
Πρώτο fix: Τρέξε `pkg install git`.

## fatal: repository not found

Σημασία: Λάθος URL, private repo ή δεν έχεις access.  
Πρώτο fix: Άνοιξε πρώτα το repo στο browser. Αν είναι private, κάνε login ή έλεγξε permissions.

## Authentication failed

Σημασία: Το GitHub δεν χρησιμοποιεί password για Git operations.  
Πρώτο fix: Χρησιμοποίησε HTTPS με token ή SSH keys. Δες τα GitHub SSH docs.

## Connection timed out

Σημασία: Πρόβλημα internet, DNS, firewall, VPN ή προσωρινό θέμα GitHub.  
Πρώτο fix: Άνοιξε το link στον browser. Ξαναδοκίμασε ή άλλαξε δίκτυο.

## Address already in use

Σημασία: Ένας local server τρέχει ήδη σε αυτό το port.  
Πρώτο fix: Σταμάτα τον παλιό με `Ctrl+C` ή διάλεξε άλλο port.

## Δεν ανοίγει αρχείο από Downloads

Σημασία: Το Termux δεν έχει storage access ή το path είναι λάθος.

Πρώτο fix: Τρέξε `termux-setup-storage`, δώσε άδεια και χρησιμοποίησε `/storage/emulated/0/Download`.

## App not installed

Σημασία: Σύγκρουση APK, παλιό Play Store build, λάθος architecture ή signature mismatch.

Πρώτο fix: Κάνε `uninstall` παλιό Termux/add-ons από άλλη πηγή. Βάλε όλα τα Termux apps από την ίδια πηγή.

## Play Protect warning

Σημασία: Το Android θέλει να σκανάρει το APK ή προειδοποιεί για sideloading.

Πρώτο fix: Χρησιμοποίησε μόνο official sources. Κάνε scan αν ζητηθεί. Μην κλείνεις Play Protect για random αρχεία.

## nano δεν σώζει

Σημασία: Ο φάκελος είναι read-only, λείπει permission ή άνοιξες λάθος path.

Πρώτο fix: Σώσε πρώτα μέσα στο home folder. Μετά κάνε `cp` όπου χρειάζεται.

## rm έσβησε λάθος αρχείο

Σημασία: Η εντολή έκανε ακριβώς αυτό που γράφτηκε.

Πρώτο fix: Σταμάτα. Μη γράψεις άλλα αρχεία εκεί. Κάνε `restore` από backup αν υπάρχει. Στην εκμάθηση χρησιμοποίησε `rm -i`.

# APK και Android troubleshooting - προβλήματα εγκατάστασης

Τα APK προβλήματα συνήθως έρχονται από source mixing, μπλοκαρισμένο unknown-app permission, παλιά Play Store builds, Play Protect warnings ή μισό download.

## Πρόβλημα - το Install button είναι μπλοκαρισμένο

Πού βρίσκεσαι: Android APK install screen.

Τι κάνεις: Περίμενε λίγα δευτερόλεπτα, κάνε scroll στο dialog αν χρειάζεται ή κλείσε overlay apps.

Τι πρέπει να δεις: Το Install γίνεται πατήσιμο.

Γιατί το κάνεις: Το Android προστατεύει από accidental taps και overlay attacks.

Αν αποτύχει: Κλείσε προσωρινά screen filters/overlay apps αν χρειάζεται.

## Πρόβλημα - App not installed

Πού βρίσκεσαι: Android installer.

Τι κάνεις: Κάνε uninstall παλιό Termux και όλα τα Termux add-ons από άλλες πηγές, μετά βάλε τα από μία πηγή.

Τι πρέπει να δεις: Η εγκατάσταση ολοκληρώνεται.

Γιατί το κάνεις: Τα Android signatures πρέπει να ταιριάζουν. Το source mixing φέρνει signature mismatch.

Αν αποτύχει: Κράτα backup πρώτα αν χρησιμοποιούσες ήδη την παλιά εφαρμογή.

## Πρόβλημα - δεν ανοίγει το APK

Πού βρίσκεσαι: Downloads/My Files.

Τι κάνεις: Ξανακατέβασε από official source, βεβαιώσου ότι τελειώνει σε .apk και πάτα το ξανά.

Τι πρέπει να δεις: Το Android ανοίγει package installer.

Γιατί το κάνεις: Μισό download δεν είναι valid APK.

Αν αποτύχει: Δοκίμασε άλλο browser ή F-Droid page.

# Git και GitHub troubleshooting - clone, pull, access

Τα Git errors είναι συνήθως κυριολεκτικά. Το μήνυμα λέει αν το πρόβλημα είναι URL, network, authentication ή local changes.

```
git status
git remote -v
git pull
```

## Γραμμή προς γραμμή εξήγηση:

- git status δείχνει local changes και branch state.
- git remote -v δείχνει πού συνδέεται το repo.
- git pull κατεβάζει updates από remote.
- Αν εμφανιστεί repository not found, άνοιξε πρώτα το URL στον browser.
- Αν authentication fails, θυμήσου ότι το GitHub δεν χρησιμοποιεί account password για Git operations.
- Αν local changes μπλοκάρουν pull, σώσε τις αλλαγές πριν κάνεις reset.
- Αν repo είναι private, το clone θέλει logged-in τρόπο με access.

# Python troubleshooting - modules, paths και script names

Τα Python προβλήματα συνήθως έρχονται από λάθος αρχείο, λάθος φάκελο, missing module ή package που δεν χτίζεται εύκολα στο Android.

```
python --version
which python
pip --version
python -m pip --version
```

## Γραμμή προς γραμμή εξήγηση:

- Επιβεβαιώνει Python version.
- Δείχνει το Python executable path.
- Ελέγχει pip command.
- Καλεί pip μέσω Python, που συχνά είναι πιο ξεκάθαρο.

## Πρόβλημα - ModuleNotFoundError

Πού βρίσκεσαι: Μετά από Python script.

Τι κάνεις: Διάβασε ακριβώς το missing module name. Ψάξε στο README για requirements.

Τι πρέπει να δεις: Ξέρεις ποιο dependency λείπει.

Γιατί το κάνεις: Random packages μπορούν να χειροτερέψουν την κατάσταση.

Αν αποτύχει: Χρησιμοποίησε pip install exact-name μόνο αν είναι ξεκάθαρο.

## Πρόβλημα - αρχείο με κενά

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Βάλε quotes γύρω από filename: python "Corrupted Files.py".

Τι πρέπει να δεις: Το Python ξεκινά το αρχείο αντί να λέει cannot open file.

Γιατί το κάνεις: Το shell κόβει λέξεις στα κενά αν δεν βάλεις εισαγωγικά.

Αν αποτύχει: Μπορείς να γράψεις τα πρώτα γράμματα και να πατήσεις Tab completion.

# Command library - τι γράφεις και πότε

Η command library εξηγεί όχι μόνο την εντολή αλλά και πότε τη χρησιμοποιείς.

## **pwd**

Το χρησιμοποιείς όταν έχεις χαθεί. Δείχνει τον τρέχοντα φάκελο.

## **ls**

Το χρησιμοποιείς πριν ανοίξεις/τρέξεις αρχεία. Δείχνει τι υπάρχει εδώ.

## **ls -la**

Το χρησιμοποιείς όταν σε νοιάζουν hidden files ή permissions.

## **cd folder**

Μπαίνει σε φάκελο. Αντικαθιστάς το folder με το ακριβές όνομα.

## **cd ..**

Γυρίζει ένα επίπεδο πίσω.

## **mkdir -p folder**

Δημιουργεί φακέλους με ασφάλεια, μαζί με parent folders.

## **cp source destination**

Κάνει αντίγραφο πριν πειράξεις κάτι.

## **mv old new**

Μετονομάζει ή μετακινεί αρχεία.

## **rm -i file**

Σβήνει με επιβεβαίωση όσο μαθαίνεις.

### **cat file**

Εμφανίζει μικρά text files.

### **nano file**

Επεξεργάζεται αρχείο κειμένου.

### **pkg search word**

Το χρησιμοποιείς πριν install όταν δεν ξέρεις ακριβές package name.

### **pkg install package**

Εγκαθιστά Termux packages.

### **python file.py**

Τρέχει Python file στον τρέχοντα φάκελο.

### **git clone URL**

Κατεβάζει repository σε φάκελο.

### **git pull**

Μέσα σε cloned repo, το ενημερώνει.

### **clear**

Καθαρίζει την οθόνη μόνο. Δεν σβήνει αρχεία.

### **history**

Δείχνει πρόσφατες εντολές.

### **du -sh folder**

Δείχνει το μέγεθος φακέλου.

```
find . -name "*.py"
```

Βρίσκει Python files κάτω από τον τρέχοντα φάκελο.

# Περισσότερα χρήσιμα - backups, logs, exports και cleanup

## Error journal

Φτιάξε `~/Termux-Course/errors.txt`. Κάθε φορά που κάτι αποτυγχάνει, γράψε ημερομηνία, εντολή, ακριβές error και τι το διόρθωσε. Αυτό γίνεται προσωπικό manual.

## Καθαροί φάκελοι project

Κράτα τα ZIP στα Downloads, αλλά τα working clones στο `~/Termux-Projects`. Έτσι δεν μπερδεύεις archives με ενεργούς φακέλους.

## Backup πριν από πείραμα

Πριν πειράξεις script που δουλεύει, κάνε copy: `cp script.py script.py.bak`. Αν χαλάσει, επαναφέρεις backup.

## Tab completion

Γράψε τα πρώτα γράμματα αρχείου/φακέλου και πάτα Tab. Μειώνει spelling mistakes και βοηθά σε μεγάλα ονόματα.

## Quotes για κενά

Αν όνομα αρχείου έχει κενά, βάλε εισαγωγικά. Ισχύει για Python files, folders και paths.

## Έλεγχος output location

Όταν script λέει ότι έσωσε αρχείο, τρέξε αμέσως `ls` στον αναμενόμενο φάκελο ή έλεγξε Downloads. Μην υποθέτεις ότι σώθηκε.

## Θερμοκρασία και μπαταρία

Μεγάλες εγκαταστάσεις μπορούν να ζεστάνουν low/mid-range κινητό. Κράτα χαμηλότερη φωτεινότητα και μην κάνεις heavy builds σε κακή φόρτιση.

## Σταθερό internet πρώτα

Αν pkg ή git βγάζουν errors σε αδύναμο Wi-Fi, μην κατηγορείς το project. Φτιάξε πρώτα το δίκτυο.

## Μία πηγή ανά app family

Βάλε Termux και add-ons από την ίδια πηγή για να αποφύγεις signature mismatch.

## Μάθε από successful output

Όταν μια εντολή δουλεύει, διάβασε και το output. Έτσι αναγνωρίζεις αργότερα τι είναι φυσιολογικό και τι όχι.

## Πλάνο 7 ημερών με ασκήσεις και ελέγχους

Οι 7 ημέρες δεν είναι για βιασύνη. Ένας καλός αρχάριος μπορεί να βάζει δύο συγκεντρωμένες ώρες την ημέρα. Ένας τελείως αρχάριος μπορεί να χρειαστεί παραπάνω και είναι φυσιολογικό.

- Ημέρα 1: Σωστή εγκατάσταση Termux, storage permission, prompt, ασφαλείς εντολές.
- Ημέρα 2: Folders, files, quotes, nano, copy, move, safe delete.
- Ημέρα 3: pkg update, pkg upgrade, package search, install core tools.
- Ημέρα 4: GitHub browser download, ZIP extraction, git clone, git pull, README reading.
- Ημέρα 5: Python scripts, missing modules, project folders και quoted filenames.
- Ημέρα 6: DedSec Project, Offline Survival Project και Corrupted Files Project.
- Ημέρα 7: Καθαρό workspace, σημειώσεις, troubleshooting και τελική πρακτική.

### Daily proof of work

Στο τέλος κάθε ημέρας γράψε τρεις γραμμές στο notes.txt: τι έκανες, τι δούλεψε, τι απέτυχε. Έτσι φτιάχνεις προσωπική troubleshooting database.

# Τελικό checklist - τι πρέπει να μπορείς να κάνεις μόνος σου

Στο τέλος, ο αναγνώστης δεν πρέπει απλώς να αντιγράψει commands. Πρέπει να καταλαβαίνει εγκατάσταση Termux, APK settings, folders, packages, GitHub, Python scripts και βασικό troubleshooting.

- Μπορώ να κατεβάσω Termux από official source και να εξηγήσω γιατί η πηγή έχει σημασία.
- Μπορώ να ανοίξω Allow from this source μόνο για την εφαρμογή που χρησιμοποίησα.
- Μπορώ να το κλείσω ξανά μετά την εγκατάσταση.
- Μπορώ να τρέξω termux-setup-storage και να βρω Downloads από Termux.
- Μπορώ να εξηγήσω διαφορά home folder και Android Downloads.
- Μπορώ να χρησιμοποιήσω pwd, ls, cd, mkdir, cp, mv, rm -i, cat, nano.
- Μπορώ να κάνω pkg update, pkg upgrade, pkg install και pkg search.
- Μπορώ να κάνω clone public GitHub repository και update με git pull.
- Μπορώ να τρέξω Python script και να χειριστώ filenames με κενά.
- Μπορώ να διαβάσω error και να κάνω σωστό πρώτο έλεγχο αντί να μαντεύω.

## Κανόνας ολοκλήρωσης

Το course ολοκληρώνεται όταν μπορείς να στήσεις νέο Termux, να κάνεις clone τα τρία practice projects, να τρέξεις τα documented entry points και να γράψεις troubleshooting note για οποιοδήποτε error χωρίς πανικό.

# Android settings appendix - ακριβώς πού να κοιτάξεις

Τα Android menus διαφέρουν ανά Samsung, Χίαomi, Pixel, OnePlus και παλιότερες εκδόσεις. Μη μάθεις μόνο ένα path. Μάθε ποιες λέξεις να ψάχνεις μέσα στα Settings.

## Λέξεις που ψάχνεις στα Android Settings

Ψάξε μία μία αυτές τις λέξεις: Install unknown apps, Unknown apps, Special app access, Permissions, Storage permission, Battery optimization, App info, Downloads.

## Βρες Install unknown apps

Πού βρίσκεσαι: Android Settings search bar.

Τι κάνεις: Γράψε Install unknown apps. Άνοιξε το αποτέλεσμα. Διάλεξε browser ή My Files.

Τι πρέπει να δεις: Λίστα εφαρμογών με switches ή σελίδα Allow from this source.

Γιατί το κάνεις: Εδώ το Android ελέγχει ποια εφαρμογή επιτρέπεται να εγκαθιστά APK.

Αν αποτύχει: Αν δεν βρίσκει τίποτα, πήγαινε Settings -> Apps -> Special app access -> Install unknown apps.

## Βρες Termux permissions

Πού βρίσκεσαι: Settings -> Apps -> Termux -> Permissions.

Τι κάνεις: Άνοιξε Termux permissions και δεξ Files/Storage/Photos and videos ανάλογα με Android version.

Τι πρέπει να δεις: Η οθόνη δείχνει Allowed ή Not allowed.

Γιατί το κάνεις: Τα scripts δεν σώζουν εύκολα σε Downloads αν το Android δεν δώσει άδεια.

Αν αποτύχει: Αν δεν εμφανίζεται storage, τρέξε πρώτα termux-setup-storage μέσα στο Termux.

## Βρες battery restrictions

Πού βρίσκεσαι: Settings -> Apps -> Termux -> Battery.

Τι κάνεις: Βάλε Termux σε unrestricted/not optimized μόνο όσο κάνεις μεγάλα installs.

Τι πρέπει να δεις: Η σελίδα battery δείχνει optimization mode.

Γιατί το κάνεις: Το Android μπορεί να σταματήσει μακρά terminal δουλειά στο background.

Αν αποτύχει: Μετά γύρνα σε normal optimization αν θες καλύτερη μπαταρία.

## Βρες Downloads

Πού βρίσκεσαι: Άνοιξε My Files ή Files app.

Τι κάνεις: Άνοιξε Internal storage -> Download.

Τι πρέπει να δεις: Βλέπεις APKs, ZIPs, PDFs και exported files.

Γιατί το κάνεις: Εκεί συνήθως σώζει ο browser.

Αν αποτύχει: Αν δεν βρίσκεις αρχείο, κάνε sort Downloads by newest first.

## Σημαντικό safety note

Μην ανοίγεις unknown-app install για όλες τις εφαρμογές. Άνοιξέ το μόνο για τον browser/file manager που χρησιμοποίησες, εγκατέστησε το trusted APK και μετά κλείσ' το.

# APK install walkthrough - κάθε οθόνη που μπορεί να δει ο χρήστης

Αυτό το μάθημα εξηγεί την εγκατάσταση σαν ακολουθία οθονών. Ο αρχάριος πρέπει να μπορεί να συγκρίνει την οθόνη του με αυτή τη λίστα.

## Οθόνη A - browser download warning

Πού βρίσκεσαι: Browser αφού πατήσεις Download APK.  
Τι κάνεις: Διάβασε το warning και συνέχισε μόνο αν το URL είναι official.  
Τι πρέπει να δεις: Το download ξεκινά και φαίνεται progress ή notification.  
Γιατί το κάνεις: Το warning είναι γενικό επειδή APKs εγκαθιστούν apps.  
Αν αποτύχει: Αν το URL δεν είναι official, ακύρωσε.

## Οθόνη B - download complete

Πού βρίσκεσαι: Browser download bar ή notification shade.  
Τι κάνεις: Πάτα Open ή άνοιξε Downloads και πάτα το APK χειροκίνητα.  
Τι πρέπει να δεις: Ανοίγει Android package installer.  
Γιατί το κάνεις: Το άνοιγμα του APK ξεκινά τη διαδικασία εγκατάστασης.  
Αν αποτύχει: Αν Open δεν κάνει τίποτα, πήγαινε My Files -> Downloads.

## Οθόνη C - χρειάζεται permission

Πού βρίσκεσαι: Android λέει ότι αυτή η εφαρμογή δεν μπορεί να κάνει install unknown apps.  
Τι κάνεις: Πάτα Settings, άνοιξε Allow from this source και μετά Back.  
Τι πρέπει να δεις: Γυρίζεις στον installer.  
Γιατί το κάνεις: Το Android δεν μπλοκάρει ειδικά το Termux, προστατεύει APK installs.  
Αν αποτύχει: Αν άνοιξες λάθος app, γύρνα και άνοιξε τον browser/file manager που χρησιμοποιείς.

## Οθόνη D - install screen

Πού βρίσκεσαι: Android package installer.  
Τι κάνεις: Πάτα Install. Μην πατάς random popups.  
Τι πρέπει να δεις: Εμφανίζεται progress, μετά App installed ή App not installed.  
Γιατί το κάνεις: Αυτό είναι το πραγματικό βήμα εγκατάστασης.  
Αν αποτύχει: Αν αποτύχει, διάβασε το exact error. Signature conflicts είναι συχνά με source mixing.

## Οθόνη E - first open

Πού βρίσκεσαι: App installed screen.  
Τι κάνεις: Πάτα Open.  
Τι πρέπει να δεις: Το Termux ανοίγει σε terminal prompt.  
Γιατί το κάνεις: Αυτό αποδεικνύει ότι η εφαρμογή ξεκινά.  
Αν αποτύχει: Αν κλείνει αμέσως, κάνε reboot και ξαναάνοιξε. Αν συνεχίζει, κάνε reinstall από μία official source.

## Τι γράφεις στις σημειώσεις

Γράψε ποια πηγή χρησιμοποίησες, το exact APK filename αν φαίνεται, αν το Android ζήτησε Allow from this source και αν το Play Protect το σκάνανε.

# Termux keyboard και touch controls - μικρά πράγματα που χάνουν οι αρχάριοι

Το terminal είναι πιο δύσκολο σε κινητό γιατί το keyboard κρύβει πλήκτρα που είναι συνηθισμένα σε υπολογιστή. Μάθε αυτά νωρίς γιατί επηρεάζουν nano, Ctrl+C, Tab completion και μεγάλα commands.

- Πάτα μία φορά το terminal πριν γράψεις αν δεν εμφανίζεται keyboard.
- Long-press μέσα στο Termux για paste, more και style ανάλογα την έκδοση.
- Ctrl+C σταματά πολλά running commands ή local servers. Δεν είναι copy στο terminal context.
- Ctrl+O στο nano κάνει save. Ctrl+X βγαίνει από nano.
- Tab completion συμπληρώνει file/folder names και μειώνει spelling mistakes.
- Αν το paste βάζει περίεργα line breaks, κάνε paste ένα command block κάθε φορά.

## Εξάσκηση Ctrl+C με ασφάλεια

Πού βρίσκεσαι: Termux prompt.

Τι κάνεις: Τρέξε `sleep 20` και μετά πάτα Ctrl+C.

Τι πρέπει να δεις: Το `sleep` σταματά και επιστρέφει prompt.

Γιατί το κάνεις: Μαθαίνεις να σταματάς process πριν το χρειαστείς σε πραγματικό πρόβλημα.

Αν αποτύχει: Αν δεν έχεις Ctrl, ενεργοποίησε Termux extra keys ή άλλο keyboard.

## Εξάσκηση Tab completion

Πού βρίσκεσαι: Μέσα σε φάκελο με αρχεία.

Τι κάνεις: Γράψε τα πρώτα γράμματα filename και πάτα Tab.

Τι πρέπει να δεις: Το Termux συμπληρώνει ή δείχνει επιλογές.

Γιατί το κάνεις: Αποφεύγεις λάθη σε μεγάλα ονόματα projects.

Αν αποτύχει: Αν δεν κάνει τίποτα, γράψε περισσότερα γράμματα ή έλεγξε με `ls`.

# ZIP files, extract και μεταφορά projects από Downloads

Πολλοί αρχάριοι κατεβάζουν ZIP από GitHub και μετά δεν ξέρουν πού πήγε ή πώς το χρησιμοποιούν. ZIP είναι archive. Πρέπει να γίνει extract πριν τρέξεις scripts μέσα του.

```
pkg install unzip
mkdir -p ~/Termux-Projects
unzip /storage/emulated/0/Download/project.zip -d ~/Termux-Projects
```

## Γραμμή προς γραμμή εξήγηση:

- Εγκαθιστά unzip αν λείπει.
- Φτιάχνει καθαρό φάκελο projects.
- Κάνει extract το ZIP από Android Downloads στο Termux projects folder.

### Βήμα 1 - επιβεβαίωσε ότι υπάρχει ZIP

Πού βρίσκεσαι: Termux prompt.  
Τι κάνεις: Τρέξε `ls /storage/emulated/0/Download | grep .zip`.  
Τι πρέπει να δεις: Βλέπεις το ZIP filename.  
Γιατί το κάνεις: Πρέπει να ξέρεις το exact filename πριν κάνεις extract.  
Αν αποτύχει: Αν δεν φαίνεται, έλεγξε My Files -> Downloads ή ξανακατέβασε.

### Βήμα 2 - extract σε καθαρό σημείο

Πού βρίσκεσαι: Termux prompt.  
Τι κάνεις: Τρέξε unzip με full ZIP path και `-d ~/Termux-Projects`.  
Τι πρέπει να δεις: Τα αρχεία γίνονται extract και εμφανίζεται folder.  
Γιατί το κάνεις: Είναι πιο καθαρό να δουλεύεις σε stable folder, όχι μέσα στο compressed Downloads.  
Αν αποτύχει: Αν γράψει command not found, τρέξε `pkg install unzip`.

### Βήμα 3 - μπες στον extracted folder

Πού βρίσκεσαι: Termux prompt.  
Τι κάνεις: Τρέξε `cd ~/Termux-Projects`, μετά `ls`, μετά `cd exact-folder-name`.  
Τι πρέπει να δεις: Είσαι μέσα στον extracted project folder.  
Γιατί το κάνεις: Τα περισσότερα scripts περιμένουν να ξεκινήσουν από project root folder.  
Αν αποτύχει: Αν το folder name είναι μεγάλο, χρησιμοποίησε Tab completion.

### ZIP vs folder

Μην προσπαθείς να τρέξεις Python files ενώ είναι ακόμη μέσα σε unextracted ZIP. Κάνε πρώτα extract και μετά τρέξε από extracted folder.

# Πώς διαβάζεις README σαν οδηγίες και όχι σαν διακόσμηση

Το README δεν είναι απλή περιγραφή. Για αρχάριο είναι instructions sheet, safety note, χάρτης και πρώτο σημείο troubleshooting.

## Σειρά διαβάσματος README

Διάβασε με αυτή τη σειρά: project purpose, requirements, installation, first run command, folder structure, output location, update instructions, known errors, license/disclaimer.

## Βήμα 1 - βρες run command

Πού βρίσκεσαι: GitHub README page ή local README.md.

Τι κάνεις: Χρησιμοποίησε Find in page και ψάξε install, run, usage, python, Termux.

Τι πρέπει να δεις: Βρίσκεις το exact command που περιμένει το project.

Γιατί το κάνεις: Το σωστό run command γλιτώνει χρόνο.

Αν αποτύχει: Αν δεν υπάρχει καθαρό run command, δες filenames αλλά μην τρέχεις random files στα τυφλά.

## Βήμα 2 - βρες requirements

Πού βρίσκεσαι: README page.

Τι κάνεις: Ψάξε requirements, dependencies, pkg install, pip install.

Τι πρέπει να δεις: Βλέπεις εργαλεία ή modules που χρειάζονται.

Γιατί το κάνεις: Missing dependencies προκαλούν τα περισσότερα Python errors.

Αν αποτύχει: Εγκατάστησε μόνο ό,τι γράφεται ή ό,τι δείχνει ξεκάθαρα το error.

## Βήμα 3 - βρες output path

Πού βρίσκεσαι: README page.

Τι κάνεις: Ψάξε save, export, output, Downloads, storage.

Τι πρέπει να δεις: Ξέρεις πού πάνε τα generated files.

Γιατί το κάνεις: Αν δεν ξέρεις output path, μπορεί να νομίζεις ότι απέτυχε ενώ δούλεψε.

Αν αποτύχει: Μετά το run, έλεγξε αυτόν τον φάκελο με ls.

# Project worksheets - συγκεκριμένες εργασίες για τα τρία practice projects

Αυτά τα worksheets κάνουν τα τρία projects μάθημα και όχι απλό download. Ο χρήστης πρέπει να ολοκληρώνει κάθε γραμμή και να γράφει αποτέλεσμα.

## DedSec worksheet

1. Κάνε clone το repo. 2. Γράψε το repo path. 3. Κάνε list top-level folders. 4. Βρες README install section. 5. Βρες safe documented entry point. 6. Γράψε τι κάνει το project με δικά σου λόγια. 7. Τρέξε μόνο documented command. 8. Αν βγει error, αντέγραψέ το ακριβώς.

## Offline Survival worksheet

1. Κάνε clone ή extract το repo. 2. Επιβεβαίωσε ότι υπάρχει main Python file. 3. Επιβεβαίωσε database folder. 4. Τρέξε main script. 5. Κάνε search/browse μία entry. 6. Βρες πού σώζονται exports. 7. Δοκίμασε offline μετά το setup. 8. Γράψε από ποια data files εξαρτάται το script.

## Corrupted Files worksheet

1. Κάνε clone ή extract το repo. 2. Άνοιξε Greece ή USA χειροκίνητα. 3. Διάλεξε ένα year. 4. Άνοιξε έναν incident folder. 5. Βρες text, JSON, metadata και index files. 6. Τρέξε reader script. 7. Ψάξε μια υπόθεση. 8. Γράψε διαφορά manual browsing και script reader.

```
cd ~/Termux-Projects
find . -maxdepth 2 -type f -name "README.md"
find . -maxdepth 3 -type f -name "*.py"
```

## Γραμμή προς γραμμή εξήγηση:

- Μπαίνει στο projects folder.
- Βρίσκει README files κοντά στην αρχή των projects.
- Βρίσκει Python files κοντά στην αρχή των projects.

# Terminal decision trees - διάλεξε τον πρώτο ασφαλή έλεγχο

Το decision tree σταματά το μάντεμα. Ξεκίνα από το σύμπτωμα, κάνε τον μικρότερο ασφαλή έλεγχο και πήγαινε στον επόμενο μόνο αν χρειάζεται.

## Όταν ένα αρχείο δεν τρέχει

1. Τρέξε pwd. 2. Τρέξε ls. 3. Έλεγξε spelling. 4. Αν έχει κενά, βάλε quotes. 5. Αν είναι Python, τρέξε python "file name.py". 6. Αν λείπει database, κάνε re-extract/re-clone. 7. Αν συνεχίζει, αντέγραψε exact error.

## Όταν αποτυγχάνει pkg

1. Έλεγξε internet στον browser. 2. Τρέξε pkg update. 3. Αν έχει mirror error, τρέξε termux-change-repo. 4. Τρέξε pkg search package. 5. Εγκατάστησε exact package. 6. Αν συνεχίζει, περίμενε και ξαναδοκίμασε.

## Όταν αποτυγχάνει git clone

1. Άνοιξε repo URL στον browser. 2. Έλεγξε spelling. 3. Επιβεβαίωσε public/private access. 4. Έλεγξε internet/VPN. 5. Βάλε git αν λείπει. 6. Για private repos χρειάζεται authenticated method. 7. Σώσε exact error.

## Όταν λείπει Python module

1. Διάβασε exact ModuleNotFoundError name. 2. Ψάξε README requirements. 3. Ψάξε requirements.txt. 4. Εγκατάστησε μόνο το απαραίτητο module. 5. Αν build fails, δες αν υπάρχει Termux package. 6. Αντέγραψε full error.

# Extra practice labs - για να δυναμώσει ο χρήστης

Τα labs είναι μικρά επίτηδες. Τα μικρά labs χτίζουν αυτοπεποίθηση χωρίς να χαλάνε αρχεία.

```
mkdir -p ~/Termux-Course/lab-files
cd ~/Termux-Course/lab-files
echo "first note" > notel.txt
cp notel.txt notel-backup.txt
ls
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει lab folder.
- Μπαίνει σε αυτό.
- Δημιουργεί text file.
- Το κάνει backup.
- Δείχνει ότι υπάρχουν και τα δύο.

```
nano checklist.txt
cat checklist.txt
```

## Γραμμή προς γραμμή εξήγηση:

- Ανοίγει checklist file.
- Μετά το save στο nano, το cat αποδεικνύει ότι υπάρχει.

```
mkdir folder-a folder-b
mv notel-backup.txt folder-a/
ls folder-a
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει δύο folders.
- Μετακινεί backup στο folder-a.
- Επιβεβαιώνει ότι το αρχείο είναι εκεί.

```
find . -type f
du -sh .
history | tail
```

## Γραμμή προς γραμμή εξήγηση:

- Βρίσκει files κάτω από τον τρέχοντα φάκελο.
- Δείχνει περίπου μέγεθος lab folder.
- Δείχνει πρόσφατες εντολές.

## Μέρος 2 - Ασκήσεις και πρακτικά εργαστήρια

Από εδώ ξεκινά το δεύτερο μέρος. Το πρώτο μέρος εξηγεί. Το δεύτερο σε βάζει να κάνεις. Κάθε άσκηση έχει σκοπό, εντολές, αποτέλεσμα και έλεγχο κατανόησης.

### Αυτοβαθμολόγηση

Πράσινο: το έκανες και μπορείς να το εξηγήσεις. Κίτρινο: το έκανες με βοήθεια. Κόκκινο: απέτυχε αλλά κράτησες exact error.

# Άσκηση 1 - prompt και output

Σκοπός: να ξεχωρίζεις τι γράφεις εσύ από τι απαντά το terminal.

```
echo "I control the input"  
pwd  
date  
whoami
```

## Γραμμή προς γραμμή εξήγηση:

- echo εκτυπώνει κείμενο.
- pwd δείχνει current folder.
- date δείχνει ώρα/ημερομηνία.
- whoami δείχνει τον χρήστη περιβάλλοντος.

## Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 2 - φάκελοι, αρχεία και ασφαλής διαγραφή

Σκοπός: να χειρίζεσαι folders χωρίς να διαγράφεις λάθος πράγματα.

```
mkdir -p ~/Termux-Course/exercises/files
cd ~/Termux-Course/exercises/files
printf "line one\nline two\n" > notes.txt
ls
cat notes.txt
cp notes.txt notes-backup.txt
rm -i notes-backup.txt
```

### Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει ασφαλή φάκελο.
- Δημιουργεί αρχείο.
- Το διαβάζει.
- Φτιάχνει backup.
- Το `rm -i` ζητά επιβεβαίωση.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 3 - storage και Downloads

Σκοπός: να δεις τη διαφορά Termux home και Android Downloads.

```
termux-setup-storage  
ls ~/storage  
ls /storage/emulated/0/Download  
mkdir -p /storage/emulated/0/Download/Termux-Course-Test  
echo "saved from Termux" > /storage/emulated/0/Download/Termux-Course-Test/test.txt
```

### Γραμμή προς γραμμή εξήγηση:

- Ζητά άδεια storage.
- Δείχνει shortcuts.
- Δείχνει Downloads.
- Γράφει test αρχείο στα Downloads.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 4 - packages

Σκοπός: να κάνεις update, search, install και verify.

```
pkg update
pkg search python
pkg install python -y
python --version
```

### Γραμμή προς γραμμή εξήγηση:

- Ανανεώνει package lists.
- Ψάχνει Python.
- Εγκαθιστά Python.
- Ελέγχει έκδοση.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 5 - nano

Σκοπός: να γράψεις και να σώσεις αρχείο στο terminal.

```
cd ~/Termux-Course/exercises
nano checklist.txt
cat checklist.txt
```

### Γραμμή προς γραμμή εξήγηση:

- Άνοιξε nano.
- Γράψε 3 γραμμές checklist.
- Ctrl+O σώζει, Enter επιβεβαιώνει, Ctrl+X βγαίνει.
- cat ελέγχει ότι σώθηκε.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 6 - README σαν οδηγίες

Σκοπός: να βρίσκεις install/run/requirements σε ένα repo.

```
pkg install git -y
git clone https://github.com/dedsec1121fk/Offline-Survival-Project.git
cd Offline-Survival-Project
ls
cat README.md | head
```

### Γραμμή προς γραμμή εξήγηση:

- Εγκαθιστά git.
- Κατεβάζει verified repo.
- Δείχνει files.
- Διαβάζει αρχή README.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

# Άσκηση 7 - DedSec Project worksheet

Σκοπός: να ελέγξεις μεγάλο repo πριν τρέξεις setup.

```
mkdir -p ~/Termux-Projects
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/DedSec
cd DedSec
ls
cat README.md | head -40
```

## Γραμμή προς γραμμή εξήγηση:

- Κατεβάζει DedSec repo.
- Δείχνει δομή.
- Διαβάζει README.
- Βρες το documented setup command πριν τρέξεις κάτι.

## Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 8 - Offline Survival worksheet

Σκοπός: να τρέξεις project που βασίζεται σε main script και database folder.

```
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/Offline-Survival-Project.git
cd Offline-Survival-Project
ls
python "Offline Survival.py"
```

### Γραμμή προς γραμμή εξήγηση:

- Κατεβάζει repo.
- Μπαίνει στον folder.
- Επιβεβαιώνει files.
- Τρέχει script με quotes λόγω space.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

# Άσκηση 9 - Corrupted Files worksheet

Σκοπός: manual browsing και Python reader.

```
cd ~/Termux-Projects
git clone https://github.com/dedsec1121fk/Corrupted-Files-Project.git
cd Corrupted-Files-Project
ls
find . -maxdepth 2 -type d | head
python "Corrupted Files.py"
```

## Γραμμή προς γραμμή εξήγηση:

- Κατεβάζει archive.
- Δείχνει folders.
- Τρέχει reader.
- Σύγκρινε manual browsing με reader.

## Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

# Άσκηση 10 - μικρό Python script

Σκοπός: create, run, edit, fix.

```
mkdir -p ~/Termux-Course/exercises/python
cd ~/Termux-Course/exercises/python
printf 'name = input("Name: ")\nprint("Hello " + name)\n' > hello.py
python hello.py
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει Python file.
- Ζητά input.
- Εκτυπώνει απάντηση.
- Μετά άλλαξέ το με nano και ξανατρέξτο.

## Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

# Άσκηση 11 - local server

Σκοπός: να δεις τι σημαίνει local link.

```
mkdir -p ~/Termux-Course/exercises/web
cd ~/Termux-Course/exercises/web
printf "<h1>Hello from Termux</h1>" > index.html
python -m http.server 8080
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει HTML.
- Ξεκινά server.
- Άνοιξε <http://127.0.0.1:8080> στο browser.
- Ctrl+C σταματά τον server.

## Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

## Άσκηση 12 - troubleshooting drills

Σκοπός: να εξασκηθείς σε errors χωρίς φόβο.

```
cd ~/Termux-Course/exercises
python missing-file.py
cd folder-that-does-not-exist
cat not-here.txt
```

### Γραμμή προς γραμμή εξήγηση:

- Προκαλεί ελεγχόμενα errors.
- Διάβασε τι λείπει.
- Μετά τρέξε pwd και ls για διάγνωση.

### Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

# Τελική εργασία - καθαρό Termux workspace

Σκοπός: να ενώσεις όλα τα βασικά σε μία δομή που ξαναχρησιμοποιείς.

```
mkdir -p ~/Termux-Workspace/{projects,notes,backups,exports,labs}
printf "Workspace created\n" > ~/Termux-Workspace/notes/status.txt
cd ~/Termux-Workspace
find . -maxdepth 2 -type d
cat notes/status.txt
```

## Γραμμή προς γραμμή εξήγηση:

- Φτιάχνει workspace.
- Γράφει status.
- Δείχνει folders.
- Πρόσθεσε ένα repo, ένα script, ένα export και ένα backup.

## Έλεγχος

Γράψε το αποτέλεσμα στις σημειώσεις σου πριν προχωρήσεις.

# Πίνακας αυτοαξιολόγησης

Χρησιμοποίησε αυτή τη σελίδα πριν πεις ότι ολοκλήρωσες το βιβλίο.

- Μπορώ να εγκαταστήσω APK από official πηγή.
- Μπορώ να εξηγήσω Allow from this source.
- Μπορώ να βρω Downloads από Termux.
- Μπορώ να εξηγήσω pwd, ls, cd, cat, cp, mv, rm.
- Μπορώ να κάνω git clone verified repo.
- Μπορώ να διαβάσω README και να βρω run command.
- Μπορώ να τρέξω Python file με spaces χρησιμοποιώντας quotes.
- Μπορώ να ξεκινήσω local server και να ανοίξω 127.0.0.1.
- Μπορώ να κρατήσω exact errors και να μην μαντεύω fixes.

## Τελικός κανόνας

Δεν χρειάζεται να τα θυμάσαι όλα απέξω. Πρέπει να ξέρεις πού να κοιτάξεις, τι να ελέγξεις πρώτο και πώς να μη χαλάσεις το περιβάλλον σου.

## Μέρος 3 - Android σε PC ή USB για Termux και Android apps

Αυτό το μέρος είναι για χρήστη που θέλει να δοκιμάσει Android σε υπολογιστή, να το τρέξει από USB ή να το εγκαταστήσει σε PC, ώστε να ανοίγει Android apps και Termux σε μεγαλύτερη οθόνη. Δεν αντικαθιστά τα προηγούμενα μαθήματα. Τα προηγούμενα μαθήματα σε μαθαίνουν Termux. Αυτό το μέρος σου δείχνει πώς να φτιάξεις περιβάλλον Android πάνω σε PC χωρίς να μαντεύεις. Η ασφαλέστερη αρχή είναι πάντα live USB test και όχι εγκατάσταση πάνω στον κύριο δίσκο.

### Πρώτος κανόνας ασφάλειας

Μην εγκαταστήσεις Android OS πάνω στον δίσκο των Windows αν δεν έχεις backup και δεν ξέρεις ακριβώς ποιο partition διαλέγεις. Για αρχή δοκίμασε live boot από USB. Το live boot δείχνει αν δουλεύουν Wi-Fi, πληκτρολόγιο, οθόνη, touchpad, ήχος και apps χωρίς να πειράξεις τον δίσκο.

### Τι θα μάθεις εδώ

Θα ξεχωρίζεις ISO, bootable USB, live boot, εγκατάσταση σε δίσκο, εγκατάσταση σε δεύτερο USB/SSD, BIOS/UEFI, Secure Boot, partition, GRUB, F-Droid, Termux σε Android-x86 περιβάλλον και γιατί κάποια APK δεν δουλεύουν σε PC.

# Ποιες επιλογές υπάρχουν και ποια να διαλέξεις

Υπάρχουν αρκετοί τρόποι να τρέξεις Android apps σε PC, αλλά δεν είναι όλοι ίδιοι. Κάποιοι είναι κανονικά Android-x86 λειτουργικά συστήματα, κάποιοι είναι ChromiumOS-like συστήματα με Android subsystem, και κάποιοι είναι containers μέσα σε Linux. Για να μην μπλέκεσαι, διάλεξε με βάση τον στόχο σου.

## Official links

[Bliss OS official site: https://blissos.org/](https://blissos.org/)

[Bliss OS documentation: https://docs.blissos.org/](https://docs.blissos.org/)

[Bliss OS live boot guide: https://docs.blissos.org/installation/live-boot-bliss-os/](https://docs.blissos.org/installation/live-boot-bliss-os/)

[Bliss OS bootable USB install guide: https://docs.blissos.org/installation/install-from-bootable-usb/](https://docs.blissos.org/installation/install-from-bootable-usb/)

[Android-x86 official site: https://www.android-x86.org/](https://www.android-x86.org/)

[Android-x86 download page: https://www.android-x86.org/download](https://www.android-x86.org/download)

[Android-x86 installation guide: https://www.android-x86.org/installhowto.html](https://www.android-x86.org/installhowto.html)

[FydeOS download page: https://fydeos.io/download/](https://fydeos.io/download/)

[FydeOS for PC: https://fydeos.io/download/pc/](https://fydeos.io/download/pc/)

[FydeOS Android apps guide:](https://fydeos.io/help/manual/manage-your-apps/add-apps-and-extensions/install-android-apps-on-fydeos/)

<https://fydeos.io/help/manual/manage-your-apps/add-apps-and-extensions/install-android-apps-on-fydeos/>

[Waydroid install instructions: https://docs.waydro.id/usage/install-on-desktops](https://docs.waydro.id/usage/install-on-desktops)

[Waydroid install/run apps: https://docs.waydro.id/usage/install-and-run-android-applications](https://docs.waydro.id/usage/install-and-run-android-applications)

[ChromeOS Flex differences: https://support.google.com/chromeosflex/answer/11542901](https://support.google.com/chromeosflex/answer/11542901)

## Αν θέλεις καθαρό Android σε PC

Πού βρίσκεσαι: Πριν κατεβάσεις οτιδήποτε.

Τι κάνεις: Ξεκίνα με Android-x86 ή Bliss OS. Κατέβασε μόνο από official site/docs. Φτιάξε bootable USB και κάνε live boot πρώτα.

Τι πρέπει να δεις: Βλέπεις Android desktop ή Android setup wizard από USB.

Γιατί το κάνεις: Αυτές οι επιλογές είναι πιο κοντά στο Android σε υπολογιστή.

Αν αποτύχει: Αν το Wi-Fi ή τα γραφικά δεν δουλεύουν σε live mode, μην κάνεις εγκατάσταση ακόμα.

## Αν θέλεις πιο laptop-like εμπειρία

Πού βρίσκεσαι: Πριν αποφασίσεις OS.

Τι κάνεις: Δες FydeOS for PC. Είναι πιο κοντά σε ChromeOS-like desktop με Android apps σε υποστηριζόμενο hardware.

Τι πρέπει να δεις: Βλέπεις desktop τύπου ChromiumOS και Android app section, αν το hardware το υποστηρίζει.

Γιατί το κάνεις: Είναι πιο φιλικό σε χρήστη που θέλει καθημερινό desktop, όχι μόνο Android test.

Αν αποτύχει: Αν το Android subsystem δεν υποστηρίζεται στο PC σου, δεν θα είναι καλή επιλογή για Termux/Android apps.

## Αν έχεις ήδη Linux laptop

Πού βρίσκεσαι: Σε Ubuntu/Debian/Fedora κλπ.

Τι κάνεις: Δες Waydroid αντί για νέο OS. Δεν είναι Android OS εγκατάσταση. Είναι Android container μέσα σε Linux.

Τι πρέπει να δεις: Τρέχει Android περιβάλλον μέσα στη Linux συνεδρία.

Γιατί το κάνεις: Είναι λιγότερο ριψοκίνδυνο από το να αλλάξεις partitions.

Αν αποτύχει: Αν είσαι σε Windows μόνο, το Waydroid δεν είναι η απλή διαδρομή.

## Τι να μην διαλέξεις για αυτόν τον στόχο

Πού βρίσκεσαι: Πριν χάσεις χρόνο με λάθος ISO.

Τι κάνεις: Μην διαλέξεις ChromeOS Flex αν ο στόχος σου είναι γενικά Android apps και Termux. Δεν είναι το σωστό μονοπάτι για αυτό το βιβλίο.

Τι πρέπει να δεις: Καταλαβαίνεις ότι ChromeOS Flex είναι άλλο πράγμα.

Γιατί το κάνεις: Το βιβλίο θέλει Android apps/Termux, όχι απλώς ελαφρύ OS για browsing.

Αν αποτύχει: Αν έχεις ChromeOS Flex ήδη, χρησιμοποίησε Linux environment μόνο αν το μοντέλο το υποστηρίζει, αλλά μην περιμένεις πλήρη Android app εμπειρία.

## Γιατί δεν βάζουμε τυχαία OS names

Παλιά projects όπως Remix OS είναι discontinued, και άλλα projects συχνά έχουν παλιές ή μπερδεμένες σελίδες. Το βιβλίο προτιμά verified official links αντί για fake downloads. Αν δεν υπάρχει καθαρή official download/install σελίδα, δεν μπαίνει σαν κύρια οδηγία.

# Λεξιλόγιο που πρέπει να ξέρεις πριν πειράξεις USB ή δίσκο

Αν δεν καταλαβαίνεις αυτές τις λέξεις, μην πας κατευθείαν σε install. Διάβασέ τις σαν μικρό λεξικό. Μετά όταν τις δεις σε installer, θα ξέρεις τι συμβαίνει.

- ISO/IMG: εικόνα λειτουργικού συστήματος. Δεν είναι απλό αρχείο που κάνεις copy στο USB. Πρέπει να γραφτεί με εργαλείο όπως Rufus, Etcher ή dd.
- Bootable USB: USB που ο υπολογιστής μπορεί να ξεκινήσει από αυτό πριν ανοίξουν τα Windows.
- Live boot: δοκιμή λειτουργικού από USB χωρίς εγκατάσταση στον δίσκο. Συνήθως οι αλλαγές χάνονται μετά το κλείσιμο.
- Installer mode: λειτουργία που γράφει Android στον δίσκο ή σε partition. Αυτό μπορεί να σβήσει δεδομένα αν διαλέξεις λάθος στόχο.
- BIOS/UEFI: το firmware με τις ρυθμίσεις εκκίνησης του PC. Εκεί αλλάζεις boot order, Secure Boot και storage mode.
- Boot menu: γρήγορο μενού επιλογής συσκευής εκκίνησης. Συχνά ανοίγει με F12, F9, F10, F11, Esc ή Del ανάλογα με τη μάρκα.
- Secure Boot: προστασία UEFI που μερικές φορές μπλοκάρει μη-Windows boot media. Αν το απενεργοποιήσεις, γράψε πώς ήταν πριν.
- BitLocker/recovery key: αν τα Windows έχουν drive encryption, αλλαγές σε Secure Boot/TPM μπορεί να ζητήσουν recovery key. Κράτα backup του key πριν πειράξεις BIOS.
- Partition: κομμάτι δίσκου. Λάθος partition σημαίνει πιθανή απώλεια Windows ή αρχείων.
- EXT4: Linux filesystem που συχνά προτείνεται για Android-x86/Bliss εγκατάσταση.
- NTFS: Windows filesystem. Μερικοί installers μπορούν να το χρησιμοποιήσουν, αλλά πρέπει να ακολουθείς ακριβώς τις οδηγίες.
- GRUB: bootloader που εμφανίζει επιλογές εκκίνησης, π.χ. Windows ή Android.
- x86\_64: αρχιτεκτονική PC 64-bit. Πολλά Android APK είναι ARM-only και μπορεί να μην τρέξουν σωστά σε x86\_64 Android.

## Μικρή άσκηση πριν συνεχίσεις

Γράψε σε χαρτί ή notes app: ποιο PC έχεις, πόσα GB RAM, αν έχεις SSD ή HDD, πόσο ελεύθερο χώρο έχεις, πώς ανοίγει το boot menu του μοντέλου σου και αν τα Windows έχουν BitLocker ενεργό.

# Πριν κατεβάσεις OS - έλεγχος hardware και backup

Το ίδιο ISO μπορεί να δουλεύει τέλεια σε ένα laptop και άσχημα σε άλλο. Ο λόγος είναι drivers: Wi-Fi, Bluetooth, GPU, ήχος, touchpad, sleep, brightness. Γι αυτό πρώτα κάνεις έλεγχο και μετά install.

## Βήμα 1 - γράψε τα στοιχεία του PC

Πού βρίσκεσαι: Μέσα στα Windows ή Linux.

Τι κάνεις: Σε Windows πήγαινε Settings -> System -> About. Γράψε processor, RAM και system type. Σε Linux τρέξε `uname -m` και `lscpu`.

Τι πρέπει να δεις: Ξέρεις αν είναι x64/x86\_64 και πόση μνήμη έχεις.

Γιατί το κάνεις: Τα Android PC builds συνήθως στοχεύουν x86/x86\_64, όχι ARM laptop.

Αν αποτύχει: Αν δεν ξέρεις architecture, μην κατεβάζεις ISO στην τύχη.

## Βήμα 2 - backup πριν από οποιοδήποτε install

Πού βρίσκεσαι: Windows ή Linux file manager.

Τι κάνεις: Αντιγραφή σημαντικών αρχείων σε εξωτερικό δίσκο ή cloud. Αν υπάρχει BitLocker, αποθήκευσε recovery key.

Τι πρέπει να δεις: Έχεις αντίγραφα εκτός του δίσκου που θα πειράξεις.

Γιατί το κάνεις: Partition mistakes μπορούν να σβήσουν δεδομένα. Το backup είναι η ασφάλειά σου.

Αν αποτύχει: Αν δεν έχεις backup, κάνε μόνο live boot, όχι install.

## Βήμα 3 - βρες boot menu key

Πού βρίσκεσαι: Browser ή manual του PC.

Τι κάνεις: Ψάξε: "μοντέλο laptop boot menu key". Σημείωσε το πλήκτρο.

Τι πρέπει να δεις: Ξέρεις πώς θα διαλέξεις USB στην εκκίνηση.

Γιατί το κάνεις: Χωρίς boot menu μπορεί να νομίζεις ότι το USB δεν δουλεύει ενώ απλώς δεν το διάλεξες.

Αν αποτύχει: Αν δεν ξέρεις το μοντέλο, κοίτα sticker στο laptop ή Windows About.

## Βήμα 4 - ετοίμασε ξεχωριστό USB

Πού βρίσκεσαι: Πάνω στο γραφείο πριν ανοίξεις Rufus/Etcher.

Τι κάνεις: Χρησιμοποίησε USB 16GB+ που δεν έχει σημαντικά αρχεία.

Τι πρέπει να δεις: Το USB μπορεί να σβηστεί χωρίς πρόβλημα.

Γιατί το κάνεις: Η δημιουργία bootable USB διαγράφει το περιεχόμενο του USB.

Αν αποτύχει: Αν έχει αρχεία, αντέγραφέ τα αλλού πρώτα.

```
# Linux checks, only if you are already on Linux
uname -m
lscpu | head
lsblk
```

## **Γραμμή προς γραμμή εξήγηση:**

- `uname -m` δείχνει architecture, π.χ. `x86_64`.
- `lscpu` δείχνει CPU πληροφορίες.
- `lsblk` δείχνει δίσκους/USB και βοηθά να μη διαλέξεις λάθος συσκευή.

# Bliss OS - τι να ξέρεις πριν το χρησιμοποιήσεις

Το Bliss OS είναι γνωστή επιλογή για Android σε PC, αλλά πρέπει να χρησιμοποιηθεί με προσοχή. Η official σελίδα αναφέρει ότι το project βρίσκεται υπό maintenance/lockdown και ότι παλιές public releases είναι snapshots χωρίς περαιτέρω updates/support. Άρα το βιβλίο δεν σου λέει "βάλε τυφλά Bliss". Σου λέει: έλεγξε την τρέχουσα official σελίδα, διάβασε docs, δοκίμασε live boot, και μετά αποφάσισε.

## Official links

Bliss OS: <https://blissos.org/>

Bliss OS docs: <https://docs.blissos.org/>

Live boot guide: <https://docs.blissos.org/installation/live-boot-bliss-os/>

Bootable USB install guide: <https://docs.blissos.org/installation/install-from-bootable-usb/>

Windows installer docs: <https://docs.blissos.org/installation/using-the-windows-installer/>

## Βήμα 1 - διάβασε status

Πού βρίσκεσαι: Browser.

Τι κάνεις: Άνοιξε blissos.org και docs.blissos.org. Δες αν υπάρχει νέα προειδοποίηση, νέο download ή νέα οδηγία.

Τι πρέπει να δεις: Ξέρεις αν οι παλιές οδηγίες ισχύουν ή αν πρέπει να περιμένεις νέο build.

Γιατί το κάνεις: Το project αλλάζει και οι παλιές οδηγίες μπορεί να είναι obsolete.

Αν αποτύχει: Αν η σελίδα λέει ότι τα docs είναι obsolete, μην κάνεις install σε κύριο δίσκο.

## Βήμα 2 - live boot πρώτα

Πού βρίσκεσαι: Windows με Rufus ή άλλο USB writer.

Τι κάνεις: Φτιάξε bootable USB και διάλεξε live boot από το boot menu.

Τι πρέπει να δεις: Το Android φορτώνει από USB χωρίς εγκατάσταση.

Γιατί το κάνεις: Ελέγχεις hardware χωρίς να πειράξεις Windows.

Αν αποτύχει: Αν μείνει black screen, δοκίμασε safe graphics/VESA/άλλο boot option αν υπάρχει.

## Βήμα 3 - έλεγχος πριν εγκατάσταση

Πού βρίσκεσαι: Μέσα στο live Bliss OS.

Τι κάνεις: Δοκίμασε Wi-Fi, keyboard, mouse/touchpad, sound, brightness, sleep, browser, F-Droid/Termux install.

Τι πρέπει να δεις: Ξέρεις τι δουλεύει και τι όχι.

Γιατί το κάνεις: Δεν εγκαθιστάς OS που δεν μπορεί να χρησιμοποιηθεί στο δικό σου PC.

Αν αποτύχει: Αν κάτι βασικό δεν δουλεύει, σταμάτα και κράτα σημειώσεις αντί να εγκαταστήσεις.

# Android-x86 - απλή επιλογή για δοκιμή Android σε PC

Το Android-x86 είναι κλασικό project για Android σε PC. Το official install guide εξηγεί ISO download, bootable USB, live boot, εγκατάσταση σε partition, GRUB, read-write /system και auto installation. Για αρχάριο, το πιο σημαντικό είναι: live boot πρώτα, install μόνο σε ασφαλή στόχο.

## Official links

Android-x86 official site: <https://www.android-x86.org/>

Android-x86 download: <https://www.android-x86.org/download>

Android-x86 install howto: <https://www.android-x86.org/installhowto.html>

### Βήμα 1 - κατέβασε ISO

Πού βρίσκεσαι: Browser στο PC.

Τι κάνεις: Άνοιξε την official download page και κατέβασε το νεότερο κατάλληλο ISO από official mirror.

Τι πρέπει να δεις: Έχεις αρχείο .iso στον Downloads folder.

Γιατί το κάνεις: Το ISO είναι η εικόνα που θα γραφτεί στο USB.

Αν αποτύχει: Αν κατέβει HTML ή λάθος αρχείο, ξαναπάτα official mirror και περίμενε να τελειώσει.

### Βήμα 2 - bootable USB

Πού βρίσκεσαι: Rufus/Etcher σε Windows ή dd σε Linux.

Τι κάνεις: Γράψε το ISO στο USB. Μην κάνεις απλό copy-paste του ISO στο USB.

Τι πρέπει να δεις: Το USB γίνεται bootable και τα παλιά αρχεία του σβήνονται.

Γιατί το κάνεις: Ο υπολογιστής χρειάζεται bootable δομή, όχι απλό αρχείο.

Αν αποτύχει: Αν το USB δεν εμφανίζεται στο boot menu, ξαναγράψε το ISO με άλλο tool ή άλλη θύρα USB.

### Βήμα 3 - live mode

Πού βρίσκεσαι: Boot menu του PC.

Τι κάνεις: Διάλεξε Live CD/Run Android-x86 without installation αν υπάρχει.

Τι πρέπει να δεις: Το Android ξεκινά χωρίς να γράψει στον δίσκο.

Γιατί το κάνεις: Το live mode είναι έλεγχος compatibility.

Αν αποτύχει: Αν η οθόνη μαυρίσει, δοκίμασε VESA/no hardware acceleration option αν υπάρχει.

### Βήμα 4 - install μόνο όταν είσαι σίγουρος

Πού βρίσκεσαι: Installer menu.

Τι κάνεις: Αν εγκαταστήσεις, διάλεξε σωστό partition/drive και διάβασε format prompts. Μην διαλέξεις auto installation στον κύριο δίσκο αν έχεις Windows που θες να κρατήσεις.

Τι πρέπει να δεις: Ο installer γράφει Android και πιθανόν bootloader.

Γιατί το κάνεις: Εδώ υπάρχει πραγματικός κίνδυνος απώλειας δεδομένων.

Αν αποτύχει: Αν δεν ξέρεις ποιο partition είναι ποιο, ακύρωσε και μάθε πρώτα με Disk Management/lslblk.

# FydeOS - επιλογή τύπου laptop με Android apps

Το FydeOS for PC είναι πιο κοντά σε ChromeOS-like εμπειρία. Μπορεί να είναι πιο φιλικό για καθημερινή χρήση, αλλά το Android subsystem δεν είναι εγγυημένο για κάθε hardware. Η official σελίδα για Android apps σημειώνει ότι Android subsystem/related functions μπορούν να είναι προσαρμοσμένα μόνο σε Intel graphics, άρα πρέπει να ελέγξεις compatibility πριν το θεωρήσεις λύση για Termux.

## Official links

[FydeOS download: https://fydeos.io/download/](https://fydeos.io/download/)

[FydeOS for PC: https://fydeos.io/download/pc/](https://fydeos.io/download/pc/)

[FydeOS Android apps guide:](#)

<https://fydeos.io/help/manual/manage-your-apps/add-apps-and-extensions/install-android-apps-on-fydeos/>

## Βήμα 1 - διάλεξε σωστή build κατηγορία

Πού βρίσκεσαι: FydeOS for PC download page.

Τι κάνεις: Διάλεξε Intel Slim, Intel Modern, AMD Graphics ή άλλη κατηγορία που ταιριάζει στο PC σου.

Τι πρέπει να δεις: Δεν κατεβάζεις τυχαία image.

Γιατί το κάνεις: Το σωστό image αυξάνει τις πιθανότητες να δουλέψουν graphics και Android subsystem.

Αν αποτύχει: Αν δεν ξέρεις CPU/GPU, γύρνα στο hardware checklist.

## Βήμα 2 - δοκίμασε πριν βασιστείς σε Android apps

Πού βρίσκεσαι: Live USB ή test install.

Τι κάνεις: Μετά την εκκίνηση άνοιξε Launcher -> Android και δες αν ξεκινά Android environment.

Τι πρέπει να δεις: Το Android environment ξεκινά και μπορείς να βάλεις apps.

Γιατί το κάνεις: Αν αυτό δεν δουλέψει, ο στόχος Termux/Android apps αποτυγχάνει.

Αν αποτύχει: Αν δεν υπάρχει Android ή δεν ξεκινά, μην συνεχίσεις ως Termux PC λύση.

## Βήμα 3 - εγκατάσταση Termux

Πού βρίσκεσαι: Μέσα στο Android environment.

Τι κάνεις: Χρησιμοποίησε app store ή F-Droid/APK μόνο από official source και εγκατέστησε Termux.

Τι πρέπει να δεις: Το Termux ανοίγει και δείχνει prompt.

Γιατί το κάνεις: Η ύπαρξη Android subsystem δεν σημαίνει αυτόματα ότι όλα τα APK δουλεύουν.

Αν αποτύχει: Αν το Termux δεν εγκαθίσταται, έλεγξε architecture/support και δοκίμασε F-Droid source.

# Waydroid - Android apps μέσα σε Linux χωρίς νέο Android OS

Το Waydroid δεν είναι ξεχωριστό Android OS για boot από USB. Είναι τρόπος να τρέξεις Android περιβάλλον μέσα σε Linux desktop, συνήθως σε Wayland session. Είναι χρήσιμο αν έχεις ήδη Linux laptop και δεν θες να πειράξεις partitions για Android-x86/Bliss.

## Official links

Waydroid docs: <https://docs.waydro.id/>

Install on desktops: <https://docs.waydro.id/usage/install-on-desktops>

Install/run Android applications: <https://docs.waydro.id/usage/install-and-run-android-applications>

### Βήμα 1 - έλεγξε ότι είσαι σε Linux/Wayland

Πού βρίσκεσαι: Linux desktop.

Τι κάνεις: Άνοιξε terminal και έλεγξε session. Σε Ubuntu 22.04+ συνήθως μπορείς να έχεις Wayland.

Τι πρέπει να δεις: Ξέρεις αν το περιβάλλον σου είναι κατάλληλο.

Γιατί το κάνεις: Το Waydroid χρειάζεται σωστό Linux desktop/container setup.

Αν αποτύχει: Αν είσαι σε Windows χωρίς Linux, αυτό δεν είναι η πρώτη επιλογή.

### Βήμα 2 - εγκατάσταση σε Ubuntu/Debian-like

Πού βρίσκεσαι: Linux terminal.

Τι κάνεις: Ακολούθησε τις official commands για curl repo script και apt install waydroid.

Τι πρέπει να δεις: Το package εγκαθίσταται και το Waydroid εμφανίζεται στο app menu.

Γιατί το κάνεις: Χρησιμοποιείς official repo αντί για τυχαία script blogs.

Αν αποτύχει: Αν το repo script αποτύχει, έλεγξε distro codename και official docs.

### Βήμα 3 - εγκατάσταση APK

Πού βρίσκεσαι: Linux terminal ή app menu.

Τι κάνεις: Χρησιμοποίησε waydroid app install xyz.apk ή F-Droid μέσα στο Android environment.

Τι πρέπει να δεις: Το app μπαίνει στο Android container.

Γιατί το κάνεις: Έτσι μπορείς να δοκιμάσεις Android apps χωρίς dual boot.

Αν αποτύχει: Αν APK είναι ARM-only, μπορεί να μην δουλέψει σε x86\_64.

```
sudo apt install curl ca-certificates -y
curl -s https://repo.waydro.id | sudo bash
sudo apt install waydroid -y
```

### Γραμμή προς γραμμή εξήγηση:

- Εγκαθιστά βασικά εργαλεία.
- Προσθέτει το official Waydroid repo script.
- Εγκαθιστά Waydroid. Τρέξε μόνο σε κατάλληλο Linux περιβάλλον και αφού διαβάσεις τα official docs.

```
waydroid app install xyz.apk
waydroid app launch com.example.app
```

### **Γραμμή προς γραμμή εξήγηση:**

- Εγκαθιστά ένα APK στο Waydroid container.
- Εκκινεί app αν ξέρεις το package name.

# ChromeOS Flex και άλλα ονόματα που πρέπει να προσέξεις

Μην μπερδεύεις κάθε ελαφρύ OS για PC με Android OS. Το ChromeOS Flex είναι χρήσιμο για να μετατρέψεις παλιό PC σε ChromeOS-like συσκευή, αλλά δεν είναι η σωστή επιλογή για αυτό το μέρος αν ο στόχος είναι γενικά Android apps και Termux. Η official Google σελίδα εξηγεί ότι ChromeOS Flex έχει διαφορές από ChromeOS και δεν υποστηρίζει γενικά Google Play/Android apps όπως ένα Chromebook.

## Official links

ChromeOS Flex differences official page: <https://support.google.com/chromeosflex/answer/11542901>

- ChromeOS Flex: καλό για browser/cloud use, όχι κύριος στόχος για Termux Android app lab.
- Remix OS: discontinued, άρα δεν το χρησιμοποιούμε σαν σύγχρονο install guide.
- Phoenix OS ή παλιά Android PC builds: συχνά stale/unclear sources. Μην βάζεις random ISO από reupload sites.
- PrimeOS: το official site που ελέγχθηκε προβάλλει PrimeOS 3.0/Primebook. Χωρίς καθαρή verified generic PC install σελίδα, δεν μπαίνει εδώ ως βήμα εγκατάστασης.

## Κανόνας επιλογής OS

Αν δεν μπορείς να βρεις official site, official download, official docs και πρόσφατη ένδειξη υποστήριξης, μην το βάζεις στον κύριο υπολογιστή σου. Μπορείς να το δοκιμάσεις μόνο σε spare μηχανήμα ή VM, αφού δεχτείς το ρίσκο.

# Πώς φτιάχνεις bootable USB σε Windows με Rufus

Το Rufus γράφει το ISO στο USB με τρόπο που ο υπολογιστής μπορεί να εκκινήσει. Αυτό σβήνει το USB. Μην βάλεις USB με φωτογραφίες ή σημαντικά αρχεία.

## Official links

Rufus official site: <https://rufus.ie/>

### Βήμα 1 - κατέβασε Rufus

Πού βρίσκεσαι: Windows browser.

Τι κάνεις: Άνοιξε <https://rufus.ie/> και κατέβασε την portable ή normal έκδοση.

Τι πρέπει να δεις: Έχεις Rufus .exe στο Downloads.

Γιατί το κάνεις: Το Rufus είναι εργαλείο δημιουργίας bootable USB.

Αν αποτύχει: Αν το κατέβασες από άλλο site, ακύρωσε και πάρε το official.

### Βήμα 2 - διάλεξε USB και ISO

Πού βρίσκεσαι: Rufus window.

Τι κάνεις: Βάλε USB, άνοιξε Rufus, στο Device διάλεξε το USB, πάτα Select και διάλεξε το Android ISO.

Τι πρέπει να δεις: Στο Rufus φαίνεται το USB και το ISO filename.

Γιατί το κάνεις: Πρέπει να επιβεβαιώσεις ότι γράφεις στο σωστό USB.

Αν αποτύχει: Αν βλέπεις εξωτερικό δίσκο με δεδομένα, σταμάτα.

### Βήμα 3 - start και warnings

Πού βρίσκεσαι: Rufus window.

Τι κάνεις: Άφησε default settings εκτός αν το official guide λέει κάτι άλλο. Πάτα Start. Αν ρωτήσει ISO mode, διάλεξε ISO mode όταν το προτείνει το guide.

Τι πρέπει να δεις: Το Rufus προειδοποιεί ότι θα σβήσει το USB και μετά γράφει το image.

Γιατί το κάνεις: Τα warnings προστατεύουν από λάθος επιλογή συσκευής.

Αν αποτύχει: Αν εμφανιστεί Syslinux download prompt, ακολούθησε το official guide για το συγκεκριμένο OS.

### Βήμα 4 - ασφαλής αφαίρεση

Πού βρίσκεσαι: Windows taskbar.

Τι κάνεις: Όταν γράφει Ready, κλείσε Rufus και κάνε safe eject στο USB.

Τι πρέπει να δεις: Το USB είναι έτοιμο για boot.

Γιατί το κάνεις: Το safe eject μειώνει την πιθανότητα corrupted USB.

Αν αποτύχει: Αν το boot αποτύχει, ξαναγράψε το USB και δοκίμασε άλλη θύρα.

# Πώς φτιάχνεις bootable USB σε Linux με dd χωρίς να σβήσεις λάθος δίσκο

Το dd είναι δυνατό και επικίνδυνο. Δεν ρωτάει αν είσαι σίγουρος. Αν βάλεις λάθος of=/dev/sdX, μπορείς να σβήσεις τον κύριο δίσκο. Χρησιμοποίησέ το μόνο αν καταλαβαίνεις ποια συσκευή είναι το USB.

## Βήμα 1 - βρες USB device

Πού βρίσκεσαι: Linux terminal.

Τι κάνεις: Τρέξε lsblk πριν και μετά βάλεις το USB. Σύγκρινε ποια νέα συσκευή εμφανίστηκε.

Τι πρέπει να δεις: Ξέρεις αν το USB είναι π.χ. /dev/sdb ή /dev/sdc.

Γιατί το κάνεις: Δεν πρέπει να γράφεις πάνω στο /dev/sda ή nvme0n1 αν αυτός είναι ο κύριος δίσκος.

Αν αποτύχει: Αν δεν είσαι 100% σίγουρος, μην χρησιμοποιήσεις dd.

## Βήμα 2 - γράψε το ISO

Πού βρίσκεσαι: Linux terminal.

Τι κάνεις: Τρέξε dd με if=το ISO και of=το USB device, όχι partition όπως /dev/sdb1.

Τι πρέπει να δεις: Το terminal μπορεί να αργήσει και μετά επιστρέφει prompt.

Γιατί το κάνεις: Γράφει raw image στο USB.

Αν αποτύχει: Αν βλέπεις permission error, χρειάζεται sudo. Αν βλέπεις λάθος device, ακύρωσε πριν πατήσεις Enter.

## Βήμα 3 - sync και αφαίρεση

Πού βρίσκεσαι: Linux terminal.

Τι κάνεις: Τρέξε sync και μετά κάνε eject/unmount.

Τι πρέπει να δεις: Το γράψιμο ολοκληρώνεται πριν βγάλεις το USB.

Γιατί το κάνεις: Αν βγάλεις USB πριν τελειώσει, μπορεί να χαλάσει το boot media.

Αν αποτύχει: Αν δεν bootάρει, ξαναγράψε από την αρχή.

```
lsblk
# Replace android.iso and /dev/sdX with your real ISO and USB device
sudo dd if=android.iso of=/dev/sdX bs=4M status=progress oflag=sync
sync
```

### Γραμμή προς γραμμή εξήγηση:

- lsblk δείχνει δίσκους και partitions.
- dd γράφει το ISO στο USB device.
- sync περιμένει να ολοκληρωθούν τα writes. Προσοχή στο /dev/sdX.

# Πώς κάνεις boot από USB χωρίς να χαθείς στο BIOS

Μετά το bootable USB, πρέπει να πεις στο PC να ξεκινήσει από το USB αντί για τον εσωτερικό δίσκο. Αυτό γίνεται από boot menu ή από BIOS/UEFI boot order.

## Βήμα 1 - γρήγορο boot menu

Πού βρίσκεσαι: PC κλειστό ή σε restart.

Τι κάνεις: Βάλε USB. Πάτα power και αμέσως πάτα επαναλαμβανόμενα F12/F9/F10/F11/Esc/Del ανάλογα με το μοντέλο.

Τι πρέπει να δεις: Ανοίγει boot menu με λίστα devices.

Γιατί το κάνεις: Δεν χρειάζεται να αλλάξεις μόνιμα BIOS settings.

Αν αποτύχει: Αν δεν ανοίγει, ψάξε το exact model + boot menu key.

## Βήμα 2 - διάλεξε USB σωστά

Πού βρίσκεσαι: Boot menu.

Τι κάνεις: Διάλεξε το USB. Αν βλέπεις UEFI: USB και απλό USB, ξεκίνα με UEFI αν το OS/PC είναι UEFI.

Τι πρέπει να δεις: Φορτώνει GRUB ή Android boot menu.

Γιατί το κάνεις: Το UEFI είναι η σύγχρονη εκκίνηση στους περισσότερους υπολογιστές.

Αν αποτύχει: Αν δεν ξεκινά, δοκίμασε την άλλη USB επιλογή ή ξαναγράψε το image.

## Βήμα 3 - Secure Boot/TPM με προσοχή

Πού βρίσκεσαι: BIOS/UEFI settings.

Τι κάνεις: Αν official guide το ζητά, απενεργοποίησε Secure Boot προσωρινά και σημείωσε την αρχική κατάσταση. Μην πειράζεις TPM/drive encryption χωρίς BitLocker recovery key.

Τι πρέπει να δεις: Το USB μπορεί να bootάρει.

Γιατί το κάνεις: Κάποια Android PC builds δεν είναι signed για Secure Boot.

Αν αποτύχει: Αν τα Windows ζητήσουν BitLocker recovery, χρησιμοποίησε το saved key ή επανέφερε τις προηγούμενες BIOS ρυθμίσεις.

## Μην αλλάζεις πολλά πράγματα μαζί

Αλλάξε μία ρύθμιση, δοκίμασε, γράψε τι έγινε. Αν αλλάξεις Secure Boot, TPM, AHCI, boot order και partitions όλα μαζί, μετά δεν θα ξέρεις τι χάλασε.

# Live boot test checklist - πριν από εγκατάσταση

Το live boot είναι το φίλτρο σου. Αν κάτι βασικό δεν δουλεύει στο live USB, πιθανότατα θα σε ταλαιπωρήσει και μετά την εγκατάσταση. Μην προσπερνάς αυτό το checklist.

- Οθόνη: σωστή ανάλυση, χωρίς black screen, χωρίς flicker.
- Πληκτρολόγιο: γράφει σε search/input fields.
- Mouse/touchpad: κλικ, scroll, δεξί κλικ αν χρειάζεται.
- Wi-Fi: βλέπει δίκτυα, συνδέεται, ανοίγει browser.
- Bluetooth: προαιρετικό, αλλά σημείωσε αν δουλεύει.
- Ήχος: ανοίγει βίντεο ή sound test.
- Μπαταρία/φόρτιση σε laptop: δείχνει κατάσταση ή όχι.
- Sleep/wake: κλείνει/ανοίγει χωρίς να παγώνει.
- USB storage: βλέπει άλλο USB αν το βάλεις.
- Android apps: μπορείς να εγκαταστήσεις F-Droid ή APK από official source.
- Termux: ανοίγει, δείχνει prompt, τρέχει echo hello και rka update.

```
echo hello
uname -m
getprop ro.product.cpu.abi
getprop ro.build.version.release
```

## Γραμμή προς γραμμή εξήγηση:

- echo hello ελέγχει terminal input.
- uname -m δείχνει kernel architecture.
- getprop ro.product.cpu.abi δείχνει Android ABI.
- getprop ro.build.version.release δείχνει Android version.

## Αποτέλεσμα checklist

Αν 8/11 δουλεύουν και τα βασικά που χρειάζεσαι είναι εντάξει, μπορείς να σκεφτείς εγκατάσταση. Αν λείπουν Wi-Fi, keyboard ή graphics, μην εγκαταστήσεις στο κύριο PC.

# Εγκατάσταση σε δίσκο ή δεύτερο USB/SSD - τι σημαίνει κάθε επιλογή

Η μόνιμη εγκατάσταση είναι το πιο επικίνδυνο σημείο. Ο installer μπορεί να γράφει partitions και bootloader. Για αρχάριο, η καλύτερη πρακτική είναι δεύτερος δίσκος, εξωτερικός SSD ή παλιό laptop χωρίς σημαντικά δεδομένα.

## Επιλογή 1 - μόνο live USB

Πού βρίσκεσαι: Χωρίς install.

Τι κάνεις: Κάθε φορά κάνεις boot από USB και δοκιμάζεις.

Τι πρέπει να δεις: Δεν αλλάζει ο δίσκος.

Γιατί το κάνεις: Ασφαλέστερη επιλογή για μάθηση.

Αν αποτύχει: Δεν κρατάει πάντα δεδομένα μετά το reboot, ανάλογα με OS/persistence.

## Επιλογή 2 - εγκατάσταση σε ξεχωριστό USB/SSD

Πού βρίσκεσαι: Installer target selection.

Τι κάνεις: Διάλεξε εξωτερικό δίσκο που έχεις αδειάσει και ξέρεις ότι είναι ο σωστός.

Τι πρέπει να δεις: Το Android OS ζει σε εξωτερικό μέσο.

Γιατί το κάνεις: Μειώνει τον κίνδυνο για τον κύριο δίσκο.

Αν αποτύχει: Αν διαλέξεις λάθος disk, μπορεί να σβήσεις Windows.

## Επιλογή 3 - dual boot σε εσωτερικό δίσκο

Πού βρίσκεσαι: Installer partition selection.

Τι κάνεις: Μόνο αν έχεις backup, ελεύθερο partition και καταλαβαίνεις GRUB/EFI.

Τι πρέπει να δεις: Στην εκκίνηση διαλέγεις Windows ή Android.

Γιατί το κάνεις: Είναι πιο πρακτικό αλλά πιο ριψοκίνδυνο.

Αν αποτύχει: Αν χαλάσει bootloader, χρειάζεσαι recovery USB για Windows/Linux.

## Επιλογή 4 - όλος ο δίσκος Android

Πού βρίσκεσαι: Auto installation ή wipe disk.

Τι κάνεις: Μόνο σε spare PC.

Τι πρέπει να δεις: Το PC ανοίγει απευθείας Android OS.

Γιατί το κάνεις: Απλό αλλά καταστροφικό για υπάρχοντα δεδομένα.

Αν αποτύχει: Μην το κάνεις σε κύριο laptop με προσωπικά αρχεία.

## Κόκκινη γραμμή

Αν ο installer γράφει erase disk, format, write partition table ή auto install, σταμάτα και βεβαιώσου ότι ο δίσκος είναι σωστός. Αυτές οι επιλογές μπορούν να σβήσουν τα πάντα.

# Πώς βάζεις Termux μέσα στο Android PC περιβάλλον

Αφού ανοίξει Android σε PC, το Termux εγκαθίσταται όπως σε Android συσκευή, με μία σημαντική διαφορά: το PC Android είναι συνήθως x86\_64. Κάποια apps/APKs είναι ARM-only και δεν δουλεύουν. Το Termux από F-Droid/GitHub official source είναι η πιο καθαρή αρχή.

## Official links

[Termux official website: https://termux.dev/en/](https://termux.dev/en/)

[Termux F-Droid package: https://f-droid.org/en/packages/com.termux/](https://f-droid.org/en/packages/com.termux/)

[Termux GitHub releases: https://github.com/termux/termux-app/releases](https://github.com/termux/termux-app/releases)

### Βήμα 1 - άνοιξε browser μέσα στο Android OS

Πού βρίσκεσαι: Android PC browser.

Τι κάνεις: Πήγαινε στο official Termux/F-Droid link. Μην ψάχνεις random APK site.

Τι πρέπει να δεις: Βλέπεις official page με package/release.

Γιατί το κάνεις: Η πηγή είναι το πιο σημαντικό κομμάτι της ασφάλειας.

Αν αποτύχει: Αν δεν έχεις internet, πρώτα φτιάξε Wi-Fi/Ethernet.

### Βήμα 2 - εγκατάστησε F-Droid ή Termux APK

Πού βρίσκεσαι: Android downloads/installer.

Τι κάνεις: Κατέβασε APK, άνοιξέ το, δώσε Allow from this source μόνο για τον browser/file manager, πάτα Install.

Τι πρέπει να δεις: Βλέπεις App installed.

Γιατί το κάνεις: Είναι η ίδια λογική sideload που έμαθες στο κινητό.

Αν αποτύχει: Αν δεις App not installed, έλεγξε παλιά εγκατάσταση, architecture ή corrupted download.

### Βήμα 3 - πρώτο Termux test

Πού βρίσκεσαι: Termux app.

Τι κάνεις: Άνοιξε Termux και τρέξε echo hello, uname -m, pkg update.

Τι πρέπει να δεις: Βλέπεις prompt, output και package update lines.

Γιατί το κάνεις: Αυτό αποδεικνύει ότι Termux λειτουργεί στο Android PC OS.

Αν αποτύχει: Αν pkg update αποτύχει, έλεγξε internet/date/time/DNS και Termux source.

```
echo hello
uname -m
pkg update
pkg install git python nano -y
python --version
```

### Γραμμή προς γραμμή εξήγηση:

- Βασικό input test.
- Δείχνει architecture.
- Ανανεώνει package lists.
- Βάζει βασικά εργαλεία.

- Ελέγχει Python.

# Γιατί κάποια Android apps δεν δουλεύουν σε PC Android

Ο χρήστης πρέπει να ξέρει ότι Android σε PC δεν σημαίνει ότι κάθε APK δουλεύει. Υπάρχουν διαφορές σε CPU architecture, Google Play Services, sensors, camera, GPS, DRM, graphics acceleration και input method.

- ARM-only APK: πολλά Android apps φτιάχνονται για ARM τηλέφωνα. Σε x86\_64 PC Android μπορεί να αποτύχουν ή να χρειάζονται translation layer.
- Google Play Services: apps που απαιτούν GMS μπορεί να μην ανοίγουν χωρίς Play Store/Services ή κατάλληλη πιστοποίηση.
- Camera/GPS/sensors: laptop hardware δεν έχει πάντα τα ίδια sensors με κινητό.
- Games: μπορεί να χρειάζονται Vulkan/OpenGL drivers που δεν δουλεύουν καλά στο PC build.
- Termux packages: σε x86\_64 μπορεί κάποια πακέτα να διαφέρουν από ARM Android phone.
- Storage paths: Downloads/Shared storage μπορεί να έχουν διαφορετική συμπεριφορά ανά OS.

## Έλεγχος architecture πριν κατηγορήσεις το app

Πού βρίσκεσαι: Μέσα στο Termux ή Android terminal.

Τι κάνεις: Τρέξε `uname -m` και `getprop ro.product.cpu.abi`.

Τι πρέπει να δεις: Ξέρεις αν είσαι x86\_64/x86 ή κάτι άλλο.

Γιατί το κάνεις: Το APK compatibility εξαρτάται από ABI.

Αν αποτύχει: Αν app είναι ARM-only, ψάξε x86 compatible version ή άλλη μέθοδο.

## Έλεγχος Google dependency

Πού βρίσκεσαι: Μέσα στο Android app.

Τι κάνεις: Αν app λέει Google Play Services required, σημείωσε το.

Τι πρέπει να δεις: Ξέρεις ότι δεν είναι απλό Termux error.

Γιατί το κάνεις: Δεν λύνεις Google dependency με `pkg install`.

Αν αποτύχει: Διάλεξε F-Droid alternatives ή OS/build με σωστή Play support αν το χρειάζεσαι νόμιμα.

# Troubleshooting για Android OS σε PC/USB

Αυτά είναι τα πιο συχνά προβλήματα. Μην αλλάζεις δέκα πράγματα μαζί. Κάνε ένα fix, ξαναδοκίμασε, γράψε αποτέλεσμα.

## Το USB δεν εμφανίζεται στο boot menu

Πού βρίσκεσαι: Boot menu/BIOS.

Τι κάνεις: Δοκίμασε άλλη θύρα USB, ξαναγράψε ISO με Rufus/Etcher, έλεγξε αν το PC είναι UEFI/Legacy.

Τι πρέπει να δεις: Το USB εμφανίζεται σαν UEFI: όνομα USB ή USB HDD.

Γιατί το κάνεις: Συχνά φταίει κακό γράψιμο ISO ή λάθος boot mode.

Αν αποτύχει: Αν συνεχίζει, δοκίμασε άλλο USB stick.

## Black screen μετά το boot

Πού βρίσκεσαι: Android boot menu.

Τι κάνεις: Δοκίμασε safe graphics/VESA/no hardware acceleration αν υπάρχει. Δοκίμασε άλλο build.

Τι πρέπει να δεις: Το σύστημα προχωράει σε desktop/setup.

Γιατί το κάνεις: Τα GPU drivers είναι συχνό πρόβλημα σε Android-x86 builds.

Αν αποτύχει: Αν δεν υπάρχει επιλογή που δουλεύει, το hardware δεν είναι καλός στόχος.

## Δεν δουλεύει Wi-Fi

Πού βρίσκεσαι: Μέσα στο live Android.

Τι κάνεις: Δοκίμασε USB tethering από κινητό, Ethernet USB adapter ή άλλο build.

Τι πρέπει να δεις: Έχεις internet έστω προσωρινά.

Γιατί το κάνεις: Πολλά Wi-Fi chipsets δεν έχουν driver στο συγκεκριμένο kernel.

Αν αποτύχει: Αν χρειάζεσαι Wi-Fi καθημερινά, μην κάνεις μόνιμη εγκατάσταση μέχρι να λυθεί.

## Δεν εγκαθίσταται Termux

Πού βρίσκεσαι: Android installer.

Τι κάνεις: Έλεγξε source, architecture, προηγούμενη εγκατάσταση και αν το APK ολοκληρώθηκε.

Τι πρέπει να δεις: App installed ή καθαρό error message.

Γιατί το κάνεις: Το App not installed είναι clue, όχι απάντηση.

Αν αποτύχει: Κατέβασε ξανά από F-Droid/GitHub official και μην αναμειγνύεις sources.

## pkg update αποτυγχάνει στο Android PC

Πού βρίσκεσαι: Termux.

Τι κάνεις: Έλεγξε internet, date/time, DNS, proxy/VPN, και μετά δοκίμασε termux-change-repo.

Τι πρέπει να δεις: Τα repositories απαντούν και το update τελειώνει.

Γιατί το κάνεις: Λάθος ώρα ή DNS μπορεί να σπάσει HTTPS/repositories.

Αν αποτύχει: Αν όλα αποτύχουν, κράτα exact error και δοκίμασε άλλο δίκτυο.

## Τα Windows δεν ξεκινούν μετά από install

Πού βρίσκεσαι: Boot menu/BIOS/GRUB.

Τι κάνεις: Μην πανικοβληθείς. Δες αν τα Windows υπάρχουν στο boot menu. Αν όχι, θα χρειαστεί Windows recovery USB ή επαναφορά bootloader.

Τι πρέπει να δεις: Μπορείς να ξαναμπεές Windows ή να επισκευάσεις boot.

Γιατί το κάνεις: Το GRUB/EFI μπορεί να άλλαξε boot order.

Αν αποτύχει: Αν δεν έχεις εμπειρία, ζήτα βοήθεια πριν γράψεις commands σε EFI partition.

## BitLocker recovery screen

Πού βρίσκεσαι: Windows boot.

Τι κάνεις: Βάλε το saved recovery key ή επανέφερε BIOS settings που άλλαξες.

Τι πρέπει να δεις: Τα Windows ξεκλειδώνουν.

Γιατί το κάνεις: Αλλαγές σε Secure Boot/TPM μπορούν να ενεργοποιήσουν BitLocker recovery.

Αν αποτύχει: Αν δεν έχεις key, μην κάνεις άλλες αλλαγές. Βρες το key από Microsoft account/backup.

## Ασκήσεις Μέρους 3 - Android PC/USB lab

Αυτές οι ασκήσεις δεν απαιτούν μόνιμη εγκατάσταση. Ο στόχος είναι να μάθεις σωστή διαδικασία έρευνας, boot USB και live test. Μόνιμη εγκατάσταση κάνεις μόνο αν έχεις backup και spare δίσκο/PC.

### Άσκηση 13 - OS επιλογή χωρίς fake links

Πού βρίσκεσαι: Browser και notes app.

Τι κάνεις: Άνοιξε τα official links για Bliss OS, Android-x86, FydeOS και Waydroid. Για κάθε ένα γράψε: official download, official docs, target hardware, μπορεί να τρέξει Android apps/Termux;

Τι πρέπει να δεις: Έχεις πίνακα σύγκρισης 4 επιλογών.

Γιατί το κάνεις: Μαθαίνεις να διαλέγεις με evidence, όχι με YouTube hype.

Αν αποτύχει: Αν δεν βρίσκεις official download/docs, γράψε "όχι ασφαλής επιλογή για τώρα".

### Άσκηση 14 - φτιάξε bootable USB

Πού βρίσκεσαι: Windows/Rufus ή Linux/dd.

Τι κάνεις: Κατέβασε ένα ISO από official source και γράψτο σε άδειο USB.

Τι πρέπει να δεις: Το USB εμφανίζεται ως bootable στο boot menu.

Γιατί το κάνεις: Μαθαίνεις τη διαφορά copy ISO και write image.

Αν αποτύχει: Αν δεν bootάρει, ξαναγράψε το USB και σημείωσε tool/settings.

### Άσκηση 15 - live boot checklist

Πού βρίσκεσαι: PC από USB.

Τι κάνεις: Μπες σε live mode και συμπλήρωσε το checklist hardware. Μην εγκαταστήσεις.

Τι πρέπει να δεις: Ξέρεις τι δουλεύει στο PC σου.

Γιατί το κάνεις: Το live test σε προστατεύει από κακή εγκατάσταση.

Αν αποτύχει: Αν τρία βασικά πράγματα αποτύχουν, δοκίμασε άλλο OS/build.

### Άσκηση 16 - Termux σε Android PC

Πού βρίσκεσαι: Live Android περιβάλλον.

Τι κάνεις: Άνοιξε browser, κατέβασε F-Droid/Termux από official source, εγκατέστησε και τρέξε echo hello + pkg update.

Τι πρέπει να δεις: Το Termux ανοίγει και τρέχει βασικές εντολές.

Γιατί το κάνεις: Αυτό αποδεικνύει τον στόχο του Μέρους 3.

Αν αποτύχει: Αν αποτύχει, γράψε ακριβές error και architecture.

### Άσκηση 17 - σύγκριση κινητού και PC

Πού βρίσκεσαι: Notes app.

Τι κάνεις: Γράψε διαφορές: πληκτρολόγιο, οθόνη, paths, performance, app compatibility, storage, errors.

Τι πρέπει να δεις: Έχεις προσωπικό συμπέρασμα ποιο περιβάλλον σε βολεύει.

Γιατί το κάνεις: Η καλή επιλογή εξαρτάται από χρήση, όχι από όνομα OS.

Αν αποτύχει: Αν δεν ξέρεις τι να επιλέξεις, κράτα το PC setup ως test lab και το κινητό ως κύριο Termux.

## Άσκηση 18 - τελική αναφορά εγκατάστασης

Πού βρίσκεσαι: Notes app ή text file.

Τι κάνεις: Γράψε μία σελίδα: ποιο OS δοκίμασες, από πού το κατέβασες, τι δούλεψε, τι απέτυχε, ποια errors είδες, τι θα έκανες διαφορετικά.

Τι πρέπει να δεις: Έχεις καθαρή τεχνική αναφορά.

Γιατί το κάνεις: Αυτό είναι πραγματική μάθηση, όχι απλώς εγκατάσταση.

Αν αποτύχει: Αν χρειαστείς βοήθεια, αυτή η αναφορά δίνει όλα τα στοιχεία.

## Τελικό αποτέλεσμα

Μετά το Μέρος 3 δεν χρειάζεται να έχεις κάνει install. Χρειάζεται να μπορείς να εξηγήσεις ποια επιλογή είναι ασφαλής για το δικό σου PC και να έχεις κάνει τουλάχιστον ένα σωστό live boot test.