📘 xenexAi Termux-Befehlssystem Dokumentation

Diese Anleitung beschreibt detailliert, wie du dein eigenes Befehlssystem unter Termux mit einer JSON-Datenbank aufbaust. Die Ausgabe erfolgt in einer übersichtlichen, farblich gestalteten Terminalansicht.

# 1. Voraussetzungen & Installation

- Installiere Termux aus dem Google Play Store oder F-Droid  
- Starte Termux  
- Aktualisiere Pakete:  
  
 pkg update && pkg upgrade  
  
- Installiere jq und nano:  
  
 pkg install jq nano

# 2. JSON-Befehlsliste vorbereiten

Erstelle eine Datei namens 'befehlsliste.json' im Home-Verzeichnis. Diese Datei speichert alle benutzerdefinierten Befehle in folgendem Format:  
  
{  
 "termux-clipboard-set": "Kopiert Text in Zwischenablage",  
 "nano ~/.bashrc": "bashrc bearbeiten"  
}

# 3. Bash-Skript erstellen

Speichere folgendes Bash-Skript unter ~/bin/befehle (ggf. Ordner anlegen):

#!/data/data/com.termux/files/usr/bin/bash  
# (Vollständiger Skript-Inhalt siehe Anhang):

Mache es ausführbar:  
  
chmod +x ~/bin/befehle

# 4. Nutzung

- Starte das Tool mit: befehle  
- Filtere nach Stichwörtern: befehle clipboard  
- Öffne Editor: befehle bearbeiten  
- Neuen Eintrag hinzufügen: befehle hinzufügen  
- Eintrag löschen: befehle löschen

# 5. Beispielausgabe

📦 Systembefehle (Infos)  
🔹 termux-clipboard-set Fügt Text zur Zwischenablage hinzu  
🔹 nano ~/.bashrc Öffnet bashrc im Editor  
  
🧠 Benutzerdefinierte Befehle  
🔸 mein-befehl Mein eigener Befehl

# 6. Tipps & Erweiterungen

- Ergänze Farben mit \e[1;32m (Grün), \e[1;34m (Blau), usw.  
- Füge weitere Abschnitte hinzu wie Backup oder Sortierung  
- Erstelle einen eigenen GitHub-Link für andere Nutzer

# 7. Fehlerbehebung

- ❌ jq ist nicht installiert → pkg install jq  
- ❌ Datei nicht gefunden → lege sie mit echo '{}' > befehlsliste.json an  
- 🔐 Datei ist nicht ausführbar → chmod +x

# Anhang: Beispiel Bash-Code

Der vollständige Shell-Code ist modular aufgebaut und farblich strukturiert, einschließlich Unterstützung für Filterung, Hinzufügen, Löschen und Hilfe-Anzeige. Ein Hintergrundbild ist im Terminal leider nicht darstellbar, aber Designfarben sind bereits integriert.

Code Anhang:

~/bin/befehle:

#!/data/data/com.termux/files/usr/bin/bash

# 📁 Pfade & Farben

JSON=”$HOME/befehlsliste.json”

GRUEN=’\e[1;32m’

BLAU=’\e[1;34m’

GRAU=’\e[0;37m’

ROT=’\e[1;31m’

RESET=’\e[0m’

BOLD=’\e[1m’

# ✅ Voraussetzungen prüfen

If ! command -v jq > /dev/null; then

Echo -e “${ROT}❌ ‘jq’ ist nicht installiert. Mit ‘pkg install jq’ nachholen.${RESET}”

Exit 1

Fi

If [ ! -f “$JSON” ]; then

Echo -e “${ROT}❌ Datei $JSON nicht gefunden. Erstelle leere Datei…${RESET}”

Echo “{}” > “$JSON”

Fi

# ⚙️ Spezialfunktionen

Case “$1” in

Bearbeiten)

Echo -e “${BLAU}📝 Öffne JSON-Datei in nano…${RESET}”

Nano “$JSON”

Exit 0

;;

Hinzufügen)

Echo -e “${GRUEN}➕ Neuen Befehl hinzufügen:${RESET}”

Read -p “🔹 Befehl (z. B. termux-clipboard-set): “ CMD

Read -p “📄 Beschreibung: “ DESC

Jq –arg cmd “$CMD” –arg desc “$DESC” ‘. + {($cmd): $desc}’ “$JSON” > “$JSON.tmp” && mv “$JSON.tmp” “$JSON”

Echo -e “${GRUEN}✅ Hinzugefügt!${RESET}”

Exit 0

;;

Löschen)

Echo -e “${ROT}🗑️ Befehl löschen:${RESET}”

Read -p “🔹 Befehl (genau wie in der Liste): “ CMD

Jq “del(.\”$CMD\”)” “$JSON” > “$JSON.tmp” && mv “$JSON.tmp” “$JSON”

Echo -e “${GRUEN}✅ Gelöscht (falls vorhanden).${RESET}”

Exit 0

;;

Hilfe|--help|-h)

Echo -e “\n${BLAU}📖 Verfügbare Optionen:${RESET}”

Echo -e “ ${GRUEN}befehle${RESET} → Zeigt alle Befehle”

Echo -e “ ${GRUEN}befehle SUCHWORT${RESET} → Filtert beide Sektionen”

Echo -e “ ${GRUEN}befehle bearbeiten${RESET} → Öffnet JSON-Datei in nano”

Echo -e “ ${GRUEN}befehle hinzufügen${RESET} → Fügt neuen Befehl hinzu”

Echo -e “ ${GRUEN}befehle löschen${RESET} → Löscht vorhandenen Befehl”

Exit 0

;;

Esac

# 🧾 Format-Funktion für Umbruch & Layout

Format\_output() {

Local cmd=”$1”

Local desc=”$2”

Local width=50

Local pad=26

Wrapped=$(echo “$desc” | fold -s -w $width)

First\_line=1

While IFS= read -r line; do

If [[ $first\_line -eq 1 ]]; then

Printf “${GRUEN}🔹 %-${pad}s${RESET} ${GRAU}%s${RESET}\n” “$cmd” “$line”

First\_line=0

Else

Printf “%-${pad}s ${GRAU}%s${RESET}\n” “ “ “$line”

Fi

Done <<< “$wrapped”

}

# 📦 System-Befehle (Info)

Declare -A SYSTEM\_BEFEHLE=(

[“termux-clipboard-set”]=”Fügt Text zur Zwischenablage hinzu”

[“termux-open URL”]=”Öffnet eine Datei oder Webseite”

[“nano ~/.bashrc”]=”Bearbeitet die Bash-Startdatei”

[“rm datei”]=”Löscht eine Datei dauerhaft aus dem Dateisystem”

[“mv a.txt b.txt”]=”Benennt eine Datei um oder verschiebt sie”

[“cp quelle ziel”]=”Kopiert eine Datei von Quelle nach Ziel”

[“exit”]=”Beendet die aktuelle Termux-Sitzung sofort”

[“ls -la”]=”Zeigt Dateien mit Details und versteckten Einträgen”

)

FILTER=”$1”

Clear

# 📋 Ausgabe: Systembefehle

Echo -e “${GRUEN}╔══════════════════════════════════════════════════════╗${RESET}”

Echo -e “${GRUEN}║ 📦 Systembefehle (Infos) ║${RESET}”

Echo -e “${GRUEN}╚══════════════════════════════════════════════════════╝${RESET}”

FOUND\_SYS=0

For cmd in “${!SYSTEM\_BEFEHLE[@]}”; do

Desc=”${SYSTEM\_BEFEHLE[$cmd]}”

If [[ -z “$FILTER” || “$cmd” == \*”$FILTER”\* || “$desc” == \*”$FILTER”\* ]]; then

Format\_output “$cmd” “$desc”

FOUND\_SYS=1

Fi

Done

[[ $FOUND\_SYS -eq 0 ]] && echo -e “${GRAU}(Keine passenden Systembefehle gefunden)${RESET}”

Echo -e “\n${GRUEN}──────────────────────────────────────────────────────${RESET}\n”

# 🧠 Benutzerdefinierte Befehle

Echo -e “${GRUEN}╔══════════════════════════════════════════════════════╗${RESET}”

Echo -e “${GRUEN}║ 🧠 Benutzerdefinierte Befehle ║${RESET}”

Echo -e “${GRUEN}╚══════════════════════════════════════════════════════╝${RESET}”

MATCH\_USER=0

Jq -r ‘to\_entries[] | “\(.key)|\(.value)”’ “$JSON” | sort | while IFS=”|” read -r cmd desc; do

If [[ -z “$FILTER” || “$cmd” == \*”$FILTER”\* || “$desc” == \*”$FILTER”\* ]]; then

Format\_output “$cmd” “$desc"

MATCH\_USER=1

fi

done

[[ "$MATCH\_USER" == 0 ]] && echo -e "${GRAU}(Keine passenden benutzerdefinierten Befehle gefunden)${RESET}"

echo -e "\n${GRAU}Nutze '${GRUEN}befehle hilfe${RESET}${GRAU}' für alle Optionen.${RESET}\n"

~/befehlsliste.json:

{

"ls": "📁 Listet alles im aktuellen Verzeichnis auf",

"cd": "📂 Verzeichnis wexhseln ('cd Downloads')",

"pwd": "📌 Zeigt aktuelles Verzeichnis an",

"cp": "📄 Kopiert Dateien ('cp quelle.txt ziel.txt')",

"mv": "✏️ Verschiebt oder benennt Dateien um ('mv alt.txt neu.txt')",

"rm": "🗑️ Löscht Dateien ('rm datei.txt')",

"rm -r": "🧹 Löscht Ordner rekursiv ('rm -r meinOrdner')",

"cat": "📖 Zeigt Inhalt einer Datei ('cat notes.txt')",

"nano": "📝 Öffnet Datei im Nano-Editor ('nano datei.txt')",

"termux-clipboard-get": "📋 Holt Inhalt aus Clipboard",

"termux-clipboard-set": "📋 Fügt Text in Clipboard ('echo hallo | termux-clipboard-set')",

"termux-open": "🌐 Öffnet Datei oder URL mit passender App",

"termux-sms-send": "📨 Sendet SMS ('termux-sms-send -n 0123456789 \"Hallo\"')",

"termux-toast": "🔔 Zeigt Toast-Nachricht auf dem Screen",

"termux-notification": "🔔 Sendet Benachrichtigung im Android-System",

"termux-session-start": "🚀 Startet eine Termux-Session",

"exit": "❌ Beendet aktuelle Termux-Session",

"chmod +x": "⚙️ Macht ein Skript ausführbar ('chmod +x befehle')",

"pkg install": "📦 Installiert ein Paket ('pkg install jq')",

"apt update && upgrade": "🔄 Aktualisiert alle installierten Pakete",

"clear": "🧽 Löscht Terminal-Bildschirm",

"history": "📜 Zeigt zuletzt verwendeten Befehle",

"befehle hilfe": "befehle ausgabe hilfe&information",

"befehle": "📋 Zeigt gespeicherte Liste benutzerdefinierter Befehle an",

"befehle hinzufügen": "➕ Fügt neuen Befehl zur Liste hinzu",

"befehle löschen": "🗑️ Entfernt gespeicherten Befehl",

"befehle bearbeiten": "📝 Öffnet Befehlsliste zur manuellen Bearbeitung"

}