from reportlab.lib.pagesizes import letter

from reportlab.lib.styles import getSampleStyleSheet, ParagraphStyle

from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer, Table, TableStyle

from reportlab.lib import colors

# Datei definieren

file\_path = "/mnt/data/Mac\_Terminal\_CheatSheet\_Advanced.pdf"

# PDF-Dokument erstellen

doc = SimpleDocTemplate(file\_path, pagesize=letter)

styles = getSampleStyleSheet()

style\_title = styles['Heading1']

style\_subtitle = styles['Heading2']

style\_normal = styles['BodyText']

elements = []

# Titel

elements.append(Paragraph("Mac Terminal Mini-Cheat-Sheet", style\_title))

elements.append(Spacer(1, 12))

# Basisbefehle

elements.append(Paragraph("📌 Basisbefehle", style\_subtitle))

basic\_data = [

["Befehl", "Beschreibung"],

["pwd", "Zeigt den aktuellen Verzeichnispfad an"],

["ls", "Listet Dateien und Ordner im aktuellen Verzeichnis auf"],

["cd <pfad>", "Wechselt in ein anderes Verzeichnis"],

["cd ~", "Springt ins Home-Verzeichnis"],

["cd ..", "Geht eine Ebene zurück"],

["mkdir <name>", "Erstellt einen neuen Ordner"],

["touch <datei>", "Erstellt eine leere Datei"],

["cp <quelle> <ziel>", "Kopiert Dateien oder Ordner"],

["mv <quelle> <ziel>", "Verschiebt oder benennt Dateien/Ordner um"],

["rm <datei>", "Löscht eine Datei"],

["rm -r <ordner>", "Löscht einen Ordner mit Inhalt"],

["clear", "Leert das Terminalfenster"],

["open .", "Öffnet den aktuellen Ordner im Finder"]

]

basic\_table = Table(basic\_data, colWidths=[120, 370])

basic\_table.setStyle(TableStyle([

('BACKGROUND', (0, 0), (-1, 0), colors.lightgrey),

('GRID', (0, 0), (-1, -1), 0.5, colors.grey),

('FONTNAME', (0, 0), (-1, 0), 'Helvetica-Bold'),

('ALIGN', (0, 0), (-1, -1), 'LEFT')

]))

elements.append(basic\_table)

elements.append(Spacer(1, 18))

# Fortgeschrittene Befehle

elements.append(Paragraph("⚡ Fortgeschrittene Befehle", style\_subtitle))

advanced\_data = [

["Befehl", "Beschreibung"],

["grep 'text' <datei>", "Sucht nach einem Textmuster in einer Datei"],

["find . -name '\*.py'", "Sucht Dateien mit bestimmtem Namen im aktuellen Verzeichnis"],

["nano <datei>", "Öffnet eine Textdatei im Terminal-Editor"],

["cat <datei>", "Zeigt den Inhalt einer Datei an"],

["head <datei>", "Zeigt die ersten Zeilen einer Datei an"],

["tail <datei>", "Zeigt die letzten Zeilen einer Datei an"],

["chmod 755 <datei>", "Ändert die Zugriffsrechte einer Datei/Ordner"],

["top", "Zeigt eine Live-Übersicht der laufenden Prozesse"],

["ps aux", "Listet alle laufenden Prozesse auf"],

["kill <PID>", "Beendet einen Prozess mit seiner PID"],

["history", "Zeigt die Befehls-Historie"],

["!!", "Führt den letzten Befehl erneut aus"]

]

advanced\_table = Table(advanced\_data, colWidths=[160, 330])

advanced\_table.setStyle(TableStyle([

('BACKGROUND', (0, 0), (-1, 0), colors.lightgrey),

('GRID', (0, 0), (-1, -1), 0.5, colors.grey),

('FONTNAME', (0, 0), (-1, 0), 'Helvetica-Bold'),

('ALIGN', (0, 0), (-1, -1), 'LEFT')

]))

elements.append(advanced\_table)

# PDF speichern

doc.build(elements)

file\_path

`~

 Home.py (damit ich sicher den richtigen Sidebar-Code habe, ohne Tippfehler aus dem ZIP)

 modules/app.py (Wirtschaftliche Verluste + Energie)

 optional modules/LCP\_app.py und modules/esg\_data\_main.py