



Contao
KONFERENZ

2025



**Entdecke die Macht der Konsole –
für jeden nutzbar, nicht nur für Nerds!**

Contao Konferenz 2025 | Christian Feneberg



01

VORAUSSETZUNGEN

HOSTING MIT SSH-ZUGANG

- Bei vielen Hostern bereits vorhanden
- ggf. Tarif prüfen und Upgrade durchführen
- Zugangsdaten prüfen (Server, User, Port)
- muss teilweise zunächst aktiviert werden



Verwaltung der Serverbenutzer

Suchen

ck25

[+ Neuer Serverbenutzer](#)

Weitere Serverbenutzer



CK25

Hosting

1 GB Speicherplatz

Server: r1.hostingwerk.de

Benutzer: ck25

 Bearbeiten

[phpMyAdmin](#) ▶ [Neues Passwort](#) ▶ [Löschen](#) ▶

PHP-Version

Legt die PHP-Version für das Hosting fest.

PHP 8.4 ▼

Details ▶

Fehler anzeigen

Aktiviert die Anzeige von PHP-Fehlern.

AUS

Details ▶

ionCube-Loader

Führt verschlüsselte PHP-Dateien aus.

AUS

Details ▶

SSH-Zugang

Aktiviert den SSH-Zugang zum Server.

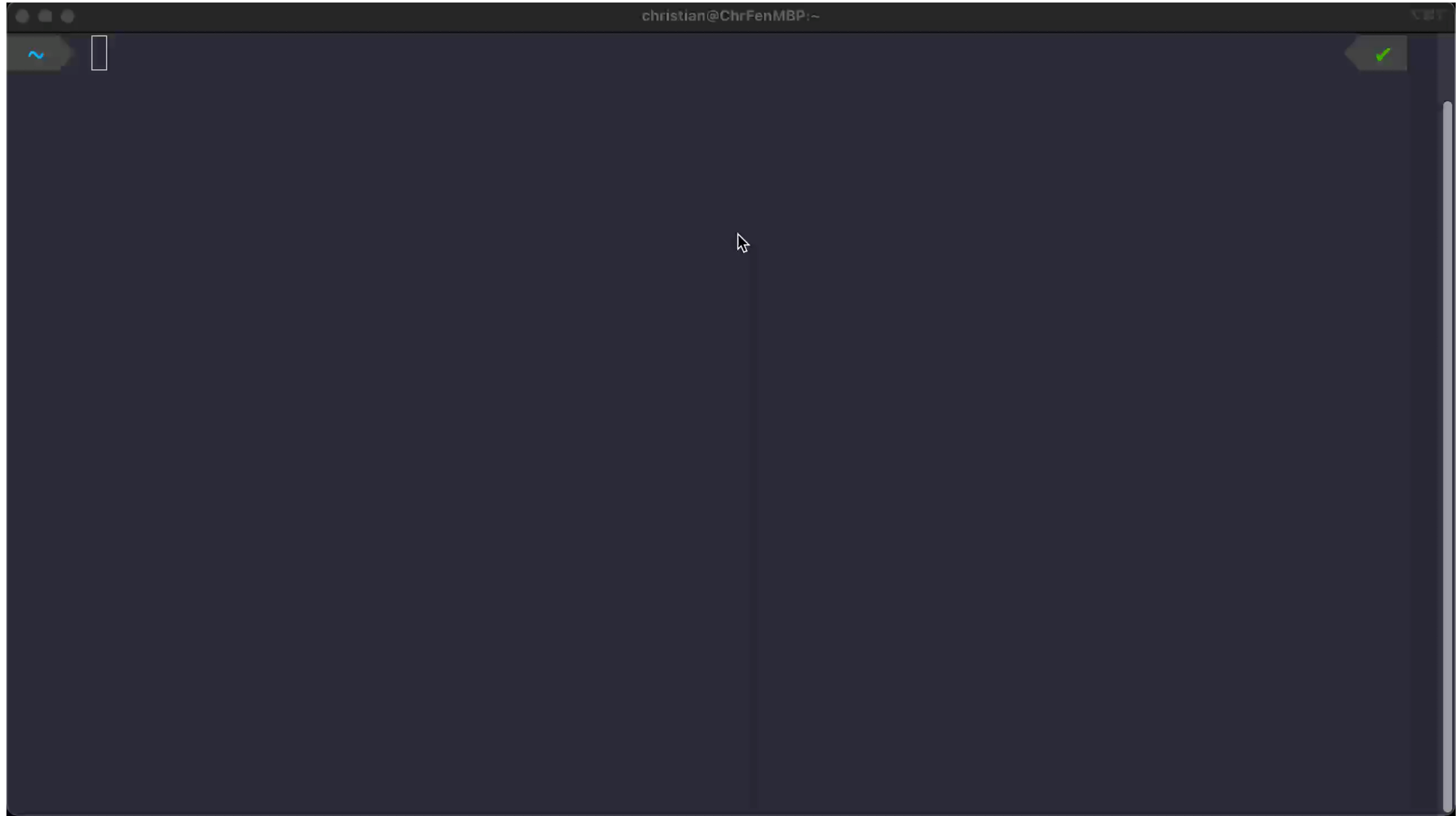
AN

Details ▶

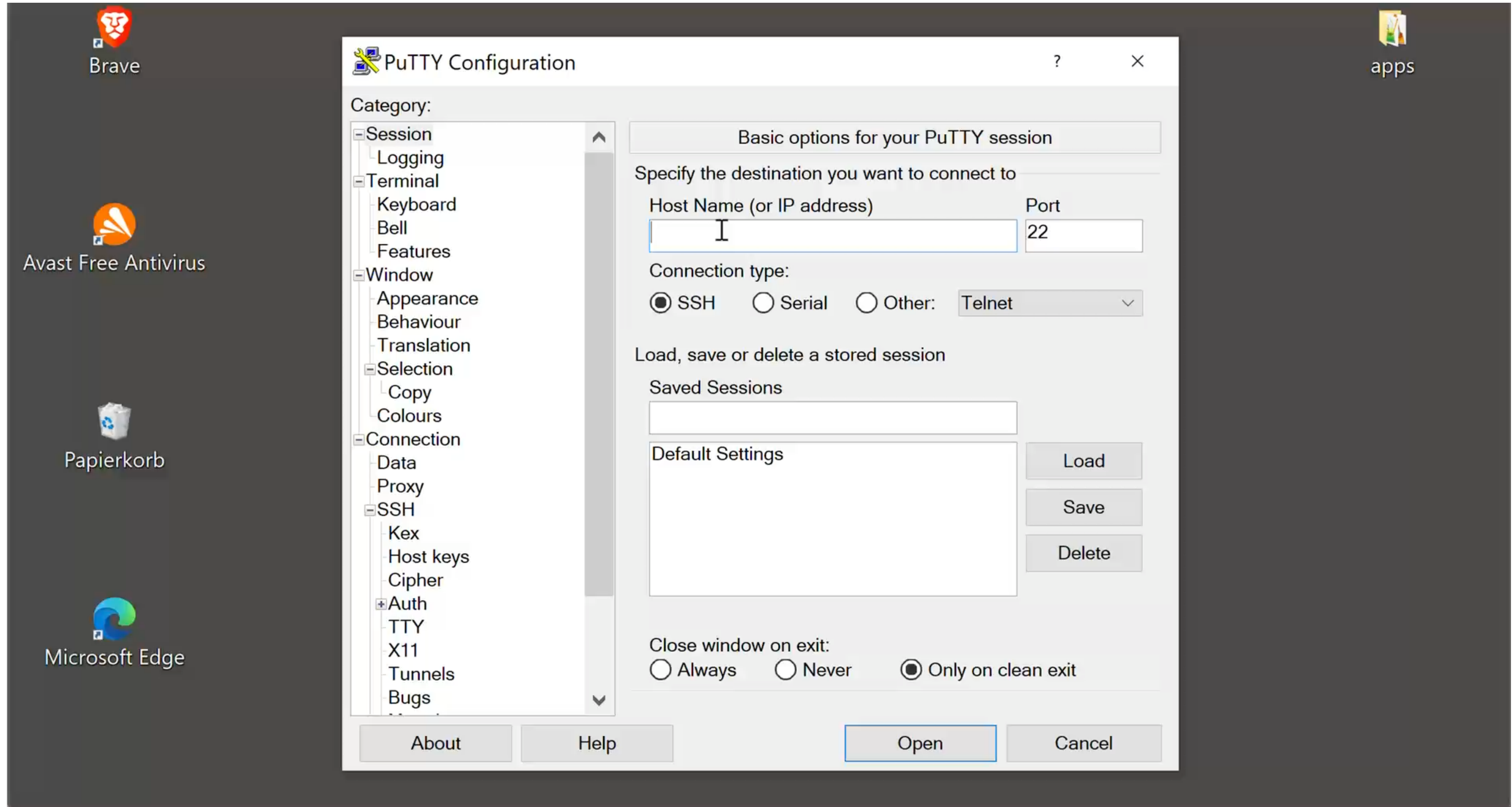
SSH-CLIENT

- **Terminals:**
 - PowerShell (Win)
 - Terminal (Mac)
 - iTerm2 (Mac)
- **Grafische SSH-Clients**
 - PuTTY (Win)
 - Termius (Win, Mac, Linux, iOS)
 - MobaXterm (Win)
 - Core Shell (Mac)

TERMINAL



PUTTY



CORE SHELL

Profiles

Tags

4.13

5.3

backup

demo

eigene

kunde

matomo

Search

ck25

ck25@r1.hostingwerk.de : 22

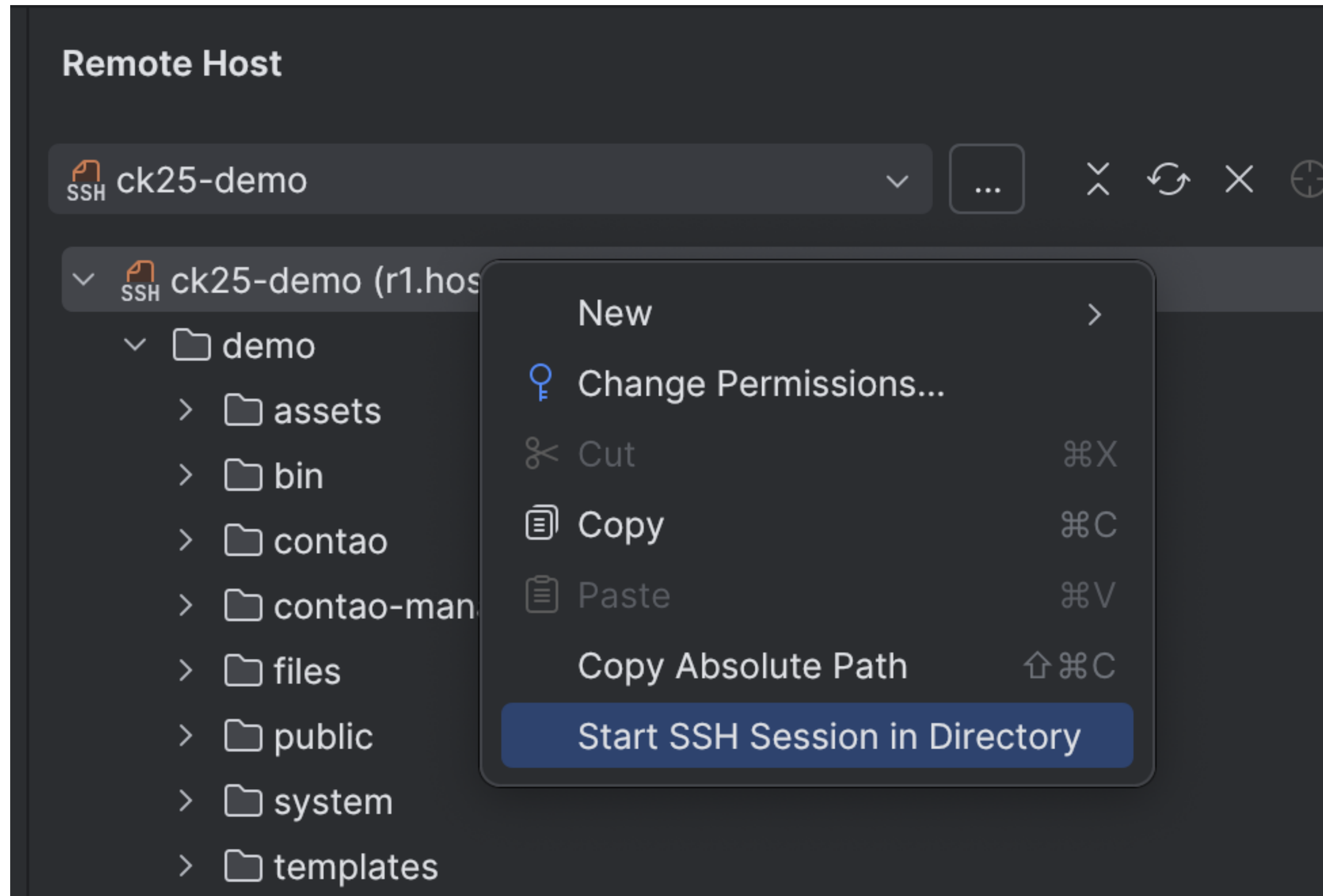
demo

webhostone

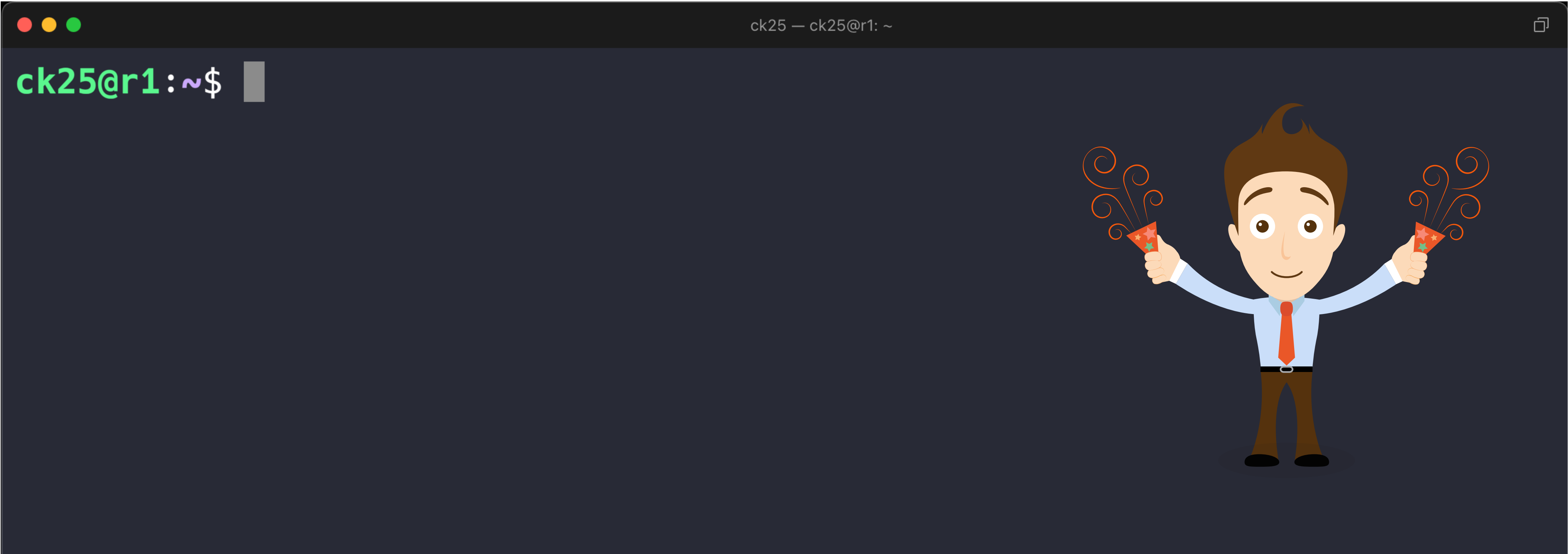
32297-1@sa9.serverdomain.org : 22

demo, eigene

AUS PHPSTORM



ERFOLGREICH VERBUNDEN



02

CRASHKURS

Befehl	Zweck	Beispiel
<code>pwd</code>	Zeigt das aktuelle Verzeichnis (Pfad) an	<code>pwd</code> → <code>/home/christian/projects</code>
<code>ls</code>	Listet Dateien und Verzeichnisse auf	<code>ls -l</code> → detaillierte Liste <code>ls -a</code> → auch versteckte Dateien (<code>.htaccess</code>) <code>ls -lh</code> → Dateigrößen lesbar
<code>cd</code>	Wechselt in ein anderes Verzeichnis	<code>cd /var/www/html</code> <code>cd ..</code> → ein Verzeichnis zurück
<code>mkdir</code>	Erstellt ein neues Verzeichnis	<code>mkdir css</code> <code>mkdir -p assets/img</code> → mehrere Ebenen erstellen
<code>cp</code>	Kopiert Dateien oder Verzeichnisse	<code>cp index.html backup.html</code> <code>cp -r images/ backup/</code> → rekursiv kopieren
<code>mv</code>	Verschiebt oder benennt Dateien/Verzeichnisse um	<code>mv style.css css/style.css</code> <code>mv old.html index.html</code>
<code>rm</code>	Löscht Dateien oder Verzeichnisse	<code>rm old.html</code> <code>rm -i index.html</code> → mit Nachfrage <code>rm -r backup/</code> → rekursiv (Ordner + Inhalte)
<code>cat</code>	Gibt den Inhalt einer Datei direkt im Terminal aus	<code>cat index.html</code>
<code>more</code>	Zeigt den Inhalt einer Datei seitenweise an	<code>more access.log</code> (mit Leertaste weiterblättern)
<code>history</code>	Zeigt eine Liste der zuletzt eingegebenen Befehle	<code>history</code> → Befehlsübersicht <code>!42</code> → führt Befehl Nr. 42 erneut aus <code>`history`</code>
<code>exit</code>	Beendet die aktuelle Shell-Sitzung	<code>exit</code>



ZUSAMMENGEFASST

- Es genügt, nur **wenige Befehle** zu lernen (Spickzettel 🤗)
- **Löschen ist endgültig**, kein Papierkorb!
- **Autovervollständigung nutzen**: zweimal TAB-Taste drücken
- **history nutzen**: zeigt die Liste der zuletzt verwendeten Befehle
- **Vorherige Befehle aufrufen**: mit den Pfeiltasten hoch und runter

03

COMPOSER

Pakete werden aktualisiert



▼ ● composer require terminal42/notification_center --no-update

```
1 $ /usr/bin/php8.4 -q -dmax_execution_time=0 -dmemory_limit=-1 -ddisplay_errors=0 -
  ddisplay_startup_errors=0 -derror_reporting=0 -dallow_url_fopen=1 -ddisable_functions= -
  ddate.timezone=Europe/Berlin /srv/www/ck25/public/demo/public/contao-manager.phar.php composer require
  terminal42/notification_center --no-update --no-scripts --prefer-stable --sort-packages --no-ansi --no-
  interaction
```

▼ ● composer update terminal42/notification_center contao/conflicts --no-install

▼ ● vendor/bin/contao-console contao:maintenance-mode enable

▼ ● composer install

▼ ● vendor/bin/contao-console contao:maintenance-mode disable

Befehl	Zweck	Beispiel
<code>composer search</code>	Sucht nach einem Paket im Composer-Repository (Packagist)	<code>composer search notification_center</code>
<code>composer require</code>	Fügt ein Paket zum Projekt hinzu und trägt es in der <code>composer.json</code> ein	<code>composer require terminal42/notification_center</code>
<code>composer install</code>	Installiert alle in der <code>composer.json</code> definierten Abhängigkeiten (z. B. beim Aufsetzen eines Projekts)	<code>composer install</code>
<code>composer update</code>	Aktualisiert alle Abhängigkeiten auf die neuesten Versionen entsprechend den Versionsangaben in der <code>composer.json</code>	<code>composer update</code>
<code>composer remove</code>	Entfernt ein Paket aus dem Projekt und aus der <code>composer.json</code>	<code>composer remove terminal42/notification_center</code>
<code>composer self-update</code>	Aktualisiert Composer selbst auf die neueste Version	<code>composer self-update</code>



ZUSAMMENGEFASST

- composer.phar herunterladen (z. B. mit wget)
- PHP-Version: muss zur Contao-Version passen
- Besonders nützlich, wenn Updates im Contao Manager Probleme bereiten

04

CONTAO-CONSOLE

Neuerstellung des Contao-Caches



- ▼ ✓ vendor/bin/contao-console contao:maintenance-mode enable
- ▼ ● rm -rf var/cache/prod
- ▼ ● vendor/bin/contao-console cache:clear --no-warmup
- ▼ ● vendor/bin/contao-console cache:warmup
- ▼ ● vendor/bin/contao-console contao:maintenance-mode disable

Befehl	Zweck
<code>contao:migrate</code>	Führt Datenbank-Migrationen aus
<code>cache:clear</code>	Löscht den Anwendungs-Cache
<code>cache:warmup</code>	Baut den Cache neu auf (für bessere Performance)
<code>contao:setup</code>	Führt das Contao-Setup aus (Installation/Konfiguration)
<code>contao:backup:create</code>	Erstellt ein Datenbank-Backup
<code>contao:backup:list</code>	Listet vorhandene Backups auf
<code>contao:backup:restore</code>	Stellt ein Backup wieder her
<code>contao:crawl</code>	Startet den Suchcrawler und indexiert die Seiten
<code>contao:user:create</code>	Erstellt einen neuen Contao-Backend-Benutzer
<code>contao:user:list</code>	Listet alle vorhandenen Contao-Backend-Benutzer auf
<code>contao:user:password</code>	Ändert das Passwort eines bestehenden Backend-Benutzers



ZUSAMMENGEFASST

- **Pfad:** *vendor/bin/contao-console*
- **PHP-Version:** muss zur Contao-Version passen
- **Backups:** Wiederherstellung nur per Konsole möglich
- **Zeitersparnis:** besonders bei migrate und Cache-Befehlen
- **Backend-Passwort:** Unkompliziert das Admin-Passwort zurücksetzen

05

PRAXISTIPPS

SNIPPETS VERWALTEN

- **Einfache Textdatei** mit häufigen Befehlen erstellen
- **Snippet-Tools** nutzen: z. B. *PhraseExpress*, *TextExpander* oder *Alfred*
- **Integrierte Snippet-Funktion** von Termius



KONSOLE CLEVER EINSETZEN

- Contao Manager per wget herunterladen
- Komplette Contao-Installation kopieren
- Backup-Archiv aller Dateien erstellen
- Datenbank-Backup per mysqldump erstellen
- Skripte und crontab für wiederkehrende Aufgaben



SUMMARY

KONSOLE

MACHT DER KONSOLE NUTZEN



01

GRAFISCHEN SSH-CLIENT NUTZEN

Zugänge für Projekte mit Tools wie Core Shell, Termius oder PuTTY einrichten.

MACHT DER KONSOLE NUTZEN



01

GRAFISCHEN SSH-CLIENT NUTZEN

Zugänge für Projekte mit Tools wie Core Shell, Termius oder PuTTY einrichten.

02

SNIPPETS NUTZEN

Wichtige Befehle stets griffbereit halten und regelmäßig ergänzen.

MACHT DER KONSOLE NUTZEN



01

GRAFISCHEN SSH-CLIENT NUTZEN

Zugänge für Projekte mit Tools wie Core Shell, Termius oder PuTTY einrichten.

02

SNIPPETS NUTZEN

Wichtige Befehle stets griffbereit halten und regelmäßig ergänzen.

03

ÜBEN UND AUSPROBIEREN

Die Konsole regelmäßig nutzen, um Aufgaben zu identifizieren, die eine Zeitersparnis bringen.



<https://fene.link/ck25>



<https://fene.link/ck25>

06

ZEIT FÜR FRAGEN





Christian Feneberg
<https://contao-academy.de>
kontakt@contao-academy.de