

Shortcuts & Menü-Navigation

Ctrl = Strg | Shift = Umschalt | Num = Ziffernblock | → im Menüpfad = Untermenü | Kürzel gelten für die Standard-Keymap

01 Suchen & navigieren

- Shift zweimal** Search Everywhere: überall suchen.
- Ctrl+Shift+A** Find Action: Aktion oder Einstellung suchen.
- Ctrl+N** / **Ctrl+Shift+N** Klasse / Datei öffnen.
- Ctrl+Alt+Shift+N** Symbol finden.
- Ctrl+F** / **Ctrl+R** In Datei suchen / ersetzen.
- Ctrl+Shift+F** / **Ctrl+Shift+R** In Dateien suchen / ersetzen.
- F3** / **Shift+F3** Nächster / vorheriger Treffer.
- Ctrl+F3** Wort am Cursor suchen.
- Ctrl+B** / **Ctrl+Alt+B** Deklaration / Implementierung.
- Ctrl+U** Übergeordnete Methode.
- Alt+F7** / **Ctrl+Alt+F7** Verwendungen suchen / kurz anzeigen.
- Ctrl+Shift+F7** Verwendungen in Datei markieren.
- Ctrl+F12** / **Ctrl+G** Dateistruktur / Zeile und Spalte.
- Ctrl+E** / **Ctrl+Shift+E** Letzte Dateien / letzte Orte.
- Ctrl+Alt+←** / **Ctrl+Alt+→** Navigationsverlauf zurück / vor.
- Ctrl+Shift+Backspace** Letzte Bearbeitungsstelle.
- Alt+↑** / **Alt+↓** Vorherige / nächste Funktion.
- Alt+F1** Aktuelle Datei in einer Ansicht finden.
- Alt+Home** Navigationsleiste öffnen.
- F4** Zur Quelle, z. B. aus der Projektansicht.

02 Auswahl & Mehrfachcursor

- Ctrl+W** / **Ctrl+Shift+W** Auswahl erweitern / verkleinern.
- Alt+J** / **Alt+Shift+J** Nächstes Vorkommen / Auswahl lösen.
- Ctrl+Alt+Shift+J** Alle gleichen Vorkommen auswählen.
- Alt+Klick** Cursor hinzufügen / entfernen.
- Alt+Shift+Insert** Spaltenauswahl umschalten.
- Ctrl+Shift+←** / **→** Wortweise markieren.
- Shift+Home** / **Shift+End** Bis Zeilenanfang / -ende markieren.
- Esc** Mehrfachcursor verlassen.

03 Schreiben & bearbeiten

- Ctrl+Space** Vervollständigung öffnen.
- Ctrl+Shift+Space** Typbezogene Vervollständigung.
- Ctrl+Shift+Enter** Anweisung vervollständigen.
- Alt+Enter** Quick-Fix / Import / Code-Aktion.
- Ctrl+P** / **Ctrl+Q** Parameter / Dokumentation.
- Ctrl+Shift+I** / **Shift+F1** Definition / externe Dokumentation.
- Ctrl+D** / **Ctrl+Y** Duplizieren / Zeile löschen.
- Ctrl+Z** / **Ctrl+Shift+Z** Rückgängig / wiederholen.
- Ctrl+Shift+V** Zwischenablage-Verlauf.
- Ctrl+Alt+Shift+V** Als reinen Text einfügen.
- Ctrl+ /** Zeilen mit # kommentieren.
- Tab** / **Shift+Tab** Einrücken / ausrücken.
- Alt+Shift+↑** / **↓** Zeile verschieben.
- Ctrl+Shift+↑** / **↓** Anweisung strukturell verschieben.
- Shift+Enter** / **Ctrl+Alt+Enter** Neue Zeile darunter / darüber.
- Ctrl+Shift+J** / **Ctrl+Enter** Zeilen verbinden / Zeile teilen.
- Ctrl+Shift+U** Groß-/Kleinschreibung umschalten.
- Ctrl+Delete** / **Ctrl+Backspace** Wort rechts / links löschen.

04 Refactoring & Qualität

- Shift+F6** Symbol mit Verwendungen umbenennen.
- Ctrl+F6** Funktionssignatur ändern.
- Ctrl+Alt+V** / **Ctrl+Alt+P** Variable / Parameter einführen.
- Ctrl+Alt+M** / **Ctrl+Alt+N** Funktion extrahieren / Inline.
- F5** / **F6** / **Alt+Delete** Kopieren / verschieben / sicher löschen.
- Ctrl+Alt+Shift+T** Refactor This: Auswahl anzeigen.
- Ctrl+Alt+T** / **Alt+Insert** Umgeben mit... / Code erzeugen.
- Ctrl+Alt+L** / **Ctrl+Alt+I** Formatieren / automatisch einrücken.
- Ctrl+Alt+O** Imports optimieren.
- F2** / **Shift+F2** / **Ctrl+F1** Nächster / voriger Fehler / Erklärung.

05 Run, Tests & Debugging

- Shift+F10** / **Shift+F9** Run / Debug; Konfiguration prüfen.
- Ctrl+Shift+F10** Datei oder Test im Kontext starten.
- Alt+Shift+F10** / **Alt+Shift+F9** Run- / Debug-Konfiguration wählen.
- Ctrl zweimal** Run Anything öffnen.
- Ctrl+F2** Programm stoppen.
- Ctrl+F8** / **Ctrl+Shift+F8** Breakpoint setzen / verwalten.
- F8** / **F7** / **Shift+F7** Step Over / Into / Smart Into.
- Shift+F8** / **F9** Step Out / Programm fortsetzen.
- Alt+F9** / **Ctrl+Alt+F9** Bis Cursor / erzwungen bis Cursor.
- Alt+F8** / **Ctrl+Alt+F8** Ausdruck / schnell auswerten.
- Alt+F10** Aktuelle Ausführungsstelle zeigen.

Debugging in sechs Schritten

- 1. Vorbereitung** Run → Edit Configurations: Interpreter, Skript, Argumente und Arbeitsverzeichnis prüfen.
- 2. Haltepunkt** Cursor in die gewünschte Zeile → Ctrl+F8. Per Rechtsklick auf den Punkt eine Bedingung setzen.
- 3. Start** Shift+F9 bzw. Alt+Shift+F9. Variablen sind erst an einem Haltepunkt im jeweiligen Kontext sichtbar.
- 4. Untersuchen** Variables / Watches verwenden; Alt+F8 für Ausdrücke. Funktionen können beim Auswerten Nebenwirkungen haben.
- 5. Schrittweise** F8 bleibt im aktuellen Aufruf; F7 geht hinein; Shift+F8 verlässt die Funktion.
- 6. Fortsetzen** F9 läuft bis zum nächsten Halt. Ctrl+F2 beendet den Prozess.

Umfang: Erweiterte Auswahl der wichtigen Standardaktionen. Vollständige eigene Belegung: Settings → Keymap; Plugins und Kontextaktionen variieren. Menüpfade: englische UI. Grundlagen: beide Vorlagen sowie JetBrains Keymap Reference, Configure Keyboard Shortcuts und Menus and Toolbars; geprüft 09.10.2026.

Druck: A3 quer bei 100 %. A4 quer: „An Seitengröße anpassen“ (ca. 70,7 %), drehen und zentrieren; eine PDF-Seite pro Blatt. Quellen: jetbrains.com/help/pycharm/keymap-reference.html und [.../configuring-keyboard-and-mouse-shortcuts.html](https://www.jetbrains.com/help/pycharm/configuring-keyboard-and-mouse-shortcuts.html)

Menüwege, Fenster & Praxis

PyCharm • Windows/Linux • Einstellungen, Projektumgebung und Bedienung

06 Fenster, Faltung & Lesezeichen

- Alt+1 / Alt+3 / Alt+4** Projekt / Suchergebnisse / Run.
- Alt+5 / Alt+6 / Alt+7** Debug / Probleme / Struktur.
- Alt+9 / Alt+0** Versionskontrolle / Commit.
- Alt+F12 / Ctrl+Alt+S** Terminal / Einstellungen.
- Ctrl+Tab / Alt+← oder →** Fensterwechsler / Editor-Tab.
- Esc / F12** Zum Editor / letzten Werkzeugfenster.
- Shift+Esc / Ctrl+Shift+F12** Aktives / alle Werkzeugfenster verbergen.
- Ctrl+F4 / Ctrl+Shift+F4** Editor-Tab / aktiven Werkzeug-Tab schließen.
- Ctrl+Num+ / Ctrl+Num-** Block auf- / zuklappen.
- Ctrl+Shift+Num+ oder Num-** Alle auf- / zuklappen; Ctrl+Shift halten.
- F11 / Ctrl+F11 / Shift+F11** Lesezeichen / Kennung / Liste.
- Ctrl+0...9** Zum nummerierten Lesezeichen.

07 Git / Versionskontrolle

- Ctrl+K / Ctrl+Shift+K** Commit / Push.
- Ctrl+T** Projekt aus VCS aktualisieren.
- Alt+`** VCS-Aktionen; Taste ggf. neu belegen.
- Alt+Shift+C** Letzte Änderungen anzeigen.

08 Wichtige Menüwege

- File → New / Open** Datei oder Projekt erstellen / öffnen.
- File → Settings → Keymap** Alle Aktionen, Belegungen und Konflikte.
- Settings → Python → Interpreter** Virtuelle Umgebung / Pakete; ältere Versionen: Project → Python Interpreter.
- Settings → Editor → Font** Schriftart und Schriftgröße.
- Settings → Editor → Code Style** Python-Einrückung und Formatierung.
- Settings → Editor → Live Templates** Codevorlagen und deren Kürzel.
- Settings → Plugins** Erweiterungen verwalten.
- View → Tool Windows** Werkzeugfenster einblenden.
- View → Appearance** Ansichten, z. B. Zen / Distraction Free.
- Navigate / Code / Refactor** Navigation, Code-Aktionen, Umbau.
- Run → Edit Configurations** Skript, Interpreter, Argumente, Arbeitsordner.
- Run → Run / Debug** Konfiguration ausführen / debuggen.
- Tools → Python Console** Python interaktiv im Projekt nutzen.
- Git → Branches** Branch auswählen oder anlegen; auch Branch-Widget.
- Datei-Rechtsklick → Local History** Show History: lokale Änderungen ansehen.
- Help → Keyboard Shortcuts PDF** Offizielle Kurzreferenz öffnen.
- Help → Find Action** Aktion suchen; alternativ Ctrl+Shift+A.

Weitere nützliche Menüaktionen

- Editor-Tab → Rechtsklick** Split Right / Split Down: Dateien nebeneinander ansehen.
- Projektordner → Rechtsklick** Mark Directory as: z. B. Sources Root oder Excluded. Nur passend zur Projektstruktur verwenden.
- Edit → Find** Suchen, Ersetzen und projektweite Suche; Regex und Suchbereich im Suchdialog prüfen.
- Code → Reformat Code** Aktuelle Datei oder Auswahl formatieren; Code Style legt die Regeln fest.
- View → Appearance** Statusleiste und Navigationsleiste einrichten; Anzeigoptionen sind versionsabhängig.
- Tools → Deployment** Remote-Dateien und Deployment, sofern die Funktion verfügbar ist.

09 Menüs bedienen & Praxis

- Tastatur** Im geöffneten Menü: ↑/↓ wählen, → Untermenü, ← zurück, Enter ausführen, Esc schließen. Buchstaben-Zugriff hängt von UI und Sprache ab.
- Menü öffnen** Alt+F öffnet in der englischen UI das File-Menü. Bei anderer Sprache können Buchstaben abweichen. Hauptmenü-Button oben links nutzen. Ctrl+Shift+A erreicht Aktionen auch ohne sichtbare Menüleiste.
- Eigene vollständige Belegung** Settings → Keymap ist maßgeblich. Nicht jede Aktion hat ein Kürzel; Plugins und Version verändern das Angebot.
- Python / PyTorch** Im Debugger x.shape, x.dtype, x.device, x.numel() prüfen; Bedingung / Logpoint per Breakpoint-Rechtsklick.
- Deutsche Tastatur / KDE** / und ` können abweichen. KDE fängt ggf. Ctrl+Alt+L, Ctrl+Alt+T oder Alt+F7 ab; Konflikt in Keymap / KDE lösen.
- Sicher arbeiten** Ersetzen: Suchbereich prüfen. Refactor für Namen verwenden. Local History ergänzt Git und Backups.

Umfang: Erweiterte Auswahl der wichtigen Standardaktionen. Vollständige eigene Belegung: Settings → Keymap; Plugins und Kontextaktionen variieren. Menüpfade: englische UI. Grundlagen: beide Vorlagen sowie JetBrains Keymap Reference, Configure Keyboard Shortcuts und Menus and Toolbars; geprüft 09.10.2026.

Druck: A3 quer bei 100 %. A4 quer: „An Seitengröße anpassen“ (ca. 70,7 %), drehen und zentrieren; eine PDF-Seite pro Blatt. Quellen: jetbrains.com/help/pycharm/keymap-reference.html und [.../configuring-keyboard-and-mouse-shortcuts.html](https://jetbrains.com/help/pycharm/configuring-keyboard-and-mouse-shortcuts.html)