```
#########################
### TMUX CLI COMMANDS ###
#########################

#########################
### https://github.com/r4ven-me/dots/blob/main/.config/tmux/tmux.conf
#########################


# Reload tmux configuration               tmux source ~/.config/tmux/tmux.conf
# Attach to an existing session           tmux attach -t session_name
# Kill all tmux processes                 pkill -f tmux
# Install specified plugins via TPM       prefix + I
# Run custom shell script at startup      run-shell "~/your/command.sh"
# (in tmux.conf)


########################
### GENERAL SETTINGS ###
########################

# set-option -g default-terminal "screen-256color" # Enable 256 colors in terminal
set -g default-terminal 'tmux-256color'
#set -g default-terminal 'xterm-256color'
set -ga terminal-overrides ',*:Tc'   # Enable TrueColors
set -ga terminal-overrides '*:Ss=\E[%p1%d q:Se=\E[ q' # Support cursor shape changes
set -g base-index 1                  # Start window numbering at 1
set -g pane-base-index 1             # Start pane numbering at 1
set -g aggressive-resize on          # Automatically resize panes on window size change
set -g history-limit 10000           # Keep 10,000 lines in scrollback buffer
set -g set-titles on                 # Enable automatic title updates
set -g mouse on                      # Enable mouse support
set -g mode-style "fg=black,bg=blue" # Customize text selection colors in copy mode
set -g status-position top           # Place status bar at top
set -g default-shell /usr/bin/zsh    # Set default shell
set -g default-command /usr/bin/zsh  # Set default shell command
set -g mode-keys vi                  # Use vi keybindings in copy mode
set -g status-keys vi                # Use vi keybindings for tmux prompts
set -g @shell_mode 'vi'              # Explicitly set shell mode to vi
set-environment -g LC_CTYPE "ru_RU.UTF-8" # Set environment encoding for Russian
#set -g pane-active-border-style fg=blue # Set active board color: example with bg
'fg=blue,bg=green'
#set -s set-clipboard off             # Clipboard mode on|off|external
#set -g display-time 100              # Set deley after changing focus of pane (def 750 ms)

## Extra status bar line
#setw -g pane-border-status bottom    # Border placement
#setw -g pane-border-format '—'       # Border style


#########################
### MOUSE KEYBINDINGS ###
#########################

# Open a new window with double-click on status bar
bind-key -n DoubleClick1Status new-window

# X11-specific bindings
%if "#{==:#{XDG_SESSION_TYPE},x11}"
   # Middle click pastes from clipboard
   unbind-key MouseDown2Pane
   bind-key -n MouseDown2Pane \
      run 'tmux set-buffer "$(xclip -out -sel clipboard)"; tmux paste-buffer'

   # Middle click cancels copy mode and pastes from clipboard
   bind-key -T copy-mode-vi MouseDown2Pane \
      select-pane \; \
```

```
      send -X cancel \; \
      run 'tmux set-buffer "$(xclip -out -sel clipboard)"; tmux paste-buffer'

   # Copy selection to clipboard without clearing it
   bind -T copy-mode-vi MouseDragEnd1Pane \
      select-pane \; \
      send-keys -X copy-pipe-no-clear "xclip -in -sel clipboard"

   # Cancel selecting on LBM click
   bind -T copy-mode-vi MouseDown1Pane \
      select-pane \; \
      send-keys -X clear-selection

   # Double and triple LBM click for word/line selection in copy mode
   bind-key -T copy-mode-vi DoubleClick1Pane \
      select-pane \; \
      send-keys -X select-word \; \
      send-keys -X copy-pipe-no-clear "xclip -in -sel clipboard"

   bind-key -T copy-mode-vi TripleClick1Pane \
      select-pane \; \
      send-keys -X select-line \; \
      send-keys -X copy-pipe-no-clear "xclip -in -sel clipboard"
%else # Non-X11 sessions bindings
   # Middle click pastes from buffer
   unbind-key MouseDown2Pane
   bind-key -n MouseDown2Pane paste-buffer

   # Middle click cancels copy mode and pastes from buffer
   bind-key -T copy-mode-vi MouseDown2Pane \
      select-pane \; \
      send -X cancel \; \
      paste-buffer

   # Copy selection without clearing it
   bind -T copy-mode-vi MouseDragEnd1Pane \
      select-pane \; \
      send-keys -X copy-selection-no-clear

   # Cancel selecting on LBM click
   bind -T copy-mode-vi MouseDown1Pane \
      select-pane \; \
      send-keys -X clear-selection

   # Double and triple LBM click for word/line selection in copy mode
   bind-key -T copy-mode-vi DoubleClick1Pane \
      select-pane \; \
      send-keys -X select-word \; \
      send-keys -X copy-selection-no-clear

   bind-key -T copy-mode-vi TripleClick1Pane \
      select-pane \; \
      send-keys -X select-line \; \
      send-keys -X copy-selection-no-clear
%endif


############################
### KEYBOARD KEYBINDINGS ###
############################

bind-key 'v' copy-mode # 'prefix+v' to enable copy mode
bind-key -T copy-mode-vi 'v' send -X begin-selection # 'v' to start selecting in copy mode

# X11-specific bindings
%if "#{==:#{XDG_SESSION_TYPE},x11}"
   # 'Enter' and 'y' in copy mode, copy selection to clipboard using xclip and cancel copy mode
```

```
   bind-key -T copy-mode-vi 'Enter' \
      select-pane \; \
      send -X copy-pipe-and-cancel "xclip -in -sel clipboard"

   bind-key -T copy-mode-vi 'y' \
      select-pane \; \
      send -X copy-pipe-and-cancel "xclip -in -sel clipboard"

   # ']' and 'p' in normal mode, paste from clipboard buffer using xclip
   bind-key ']' \
      run 'tmux set-buffer "$(xclip -out -sel clipboard)"; tmux paste-buffer'

   bind-key 'P' \
      run 'tmux set-buffer "$(xclip -out -sel clipboard)"; tmux paste-buffer'

   # ']' and 'p' in copy mode, cancel copy mode and paste from clipboard buffer using xclip
   bind-key -T copy-mode-vi ']' \
      select-pane \; \
      send -X cancel \; \
      run 'tmux set-buffer "$(xclip -out -sel clipboard)"; tmux paste-buffer'

   bind-key -T copy-mode-vi 'p' \
      select-pane \; \
      send -X cancel \; \
      run 'tmux set-buffer "$(xclip -out -sel clipboard)"; tmux paste-buffer'

%else # If not in X11
   # 'y' in copy mode, copy selection to tmux buffer and cancel copy mode
   bind-key -T copy-mode-vi 'y' \
      select-pane \; \
      send -X copy-selection-and-cancel

   # When pressing 'p' in normal mode, paste from tmux buffer
   bind-key 'P' paste-buffer

   # When pressing 'p' in copy mode, cancel copy mode and paste from tmux buffer
   bind-key -T copy-mode-vi 'p' \
      select-pane \; \
      send -X cancel \; \
      paste-buffer
%endif

# vi-like navigation for panes
bind-key -r -T prefix h select-pane -L
bind-key -r -T prefix j select-pane -D
bind-key -r -T prefix k select-pane -U
bind-key -r -T prefix l select-pane -R

# Resize panes using vi-like keys
bind-key -r -T prefix H resize-pane -L 2
bind-key -r -T prefix J resize-pane -D 2
bind-key -r -T prefix K resize-pane -U 2
bind-key -r -T prefix L resize-pane -R 2

# Manual save/restore bindings for tmux-resurrect
# 'prefix+F5' to save
bind-key -T prefix F5 run-shell \
    '~/.config/tmux/plugins/tmux-resurrect/scripts/save.sh && mv -f $(find
~/.local/share/tmux/resurrect -type f -name "tmux_resurrect_*.txt" -size +0c | sort | tail -1)
~/.local/share/tmux/resurrect/main.txt && tmux display-message "Environment saved manually!"'

# 'prefix+F6' to load
bind-key -T prefix F6 run-shell \
    'ln -sf ~/.local/share/tmux/resurrect/main.txt ~/.local/share/tmux/resurrect/last &&
~/.config/tmux/plugins/tmux-resurrect/scripts/restore.sh'
```

```
####################
### PLUGINS LIST ###
####################

# Plugins
set -g @plugin 'tmux-plugins/tpm'            # Plugins manager
set -g @plugin 'tmux-plugins/tmux-sensible'  # Optimal tmux default settings
set -g @plugin 'arcticicestudio/nord-tmux'   # Nord theme for tmux
set -g @plugin 'tmux-plugins/tmux-resurrect' # Save and restore tmux env
set -g @plugin 'tmux-plugins/tmux-continuum' # Automates save and restore actions of tmux-
resurrect
set -g @plugin 'sainnhe/tmux-fzf'            # Fuzzy finder


######################
### PLUGINS CONFIG ###
######################

# [tmux-resurrect]
# Cleanup old session files
run-shell \
   'find ~/.local/share/tmux/resurrect/ -type f -name "*.txt" ! -name "main.txt" -mtime +3 -
delete'
#set -g @resurrect-capture-pane-contents 'on' # Restore previous sessions content
set -g @resurrect-strategy-nvim 'session' # Use sessions for restoring Neovim
set -g @resurrect-processes 'ssh ranger mc tmux cmatrix ipython' # Processes to restore

# [tmux-continuum]
set -g @continuum-boot 'on'          # Auto-start tmux via systemd
set -g @continuum-restore 'on'       # Auto-restore tmux sessions
set -g @continuum-save-interval '60' # Auto-save every 60 minutes

# [tmux-fzf]
# Deault options
set-environment -g TMUX_FZF_OPTIONS "
    -p -w 65% -h 35% -m
    --exact
    --color=fg:#e5e9f0,bg:#2e3440,hl:#81a1c1
    --color=fg+:#e5e9f0,bg+:#2e3440,hl+:#81a1c1
    --color=info:#eacb8a,prompt:#bf6069,pointer:#b48dac
    --color=marker:#a3be8b,spinner:#b48dac,header:#a3be8b
    --color=border:#5e81ac
    --border=rounded
"

# Quick commands
bind-key C-b run-shell "
    command_list=\"~/.config/tmux/commands.list\"
    command=\$(cat \$command_list | fzf-tmux \$TMUX_FZF_OPTIONS --reverse --prompt=\"⚡ Command:
\") && \
    tmux send-keys \"\${command#*] }\" || \
    exit 0
"

# Quick SSH connection from ~/.ssh/config
bind-key B run-shell "
    host=\$(grep -E \"^Host \" ~/.ssh/config | sed \"s/^Host //\" | fzf-tmux \$TMUX_FZF_OPTIONS -
-reverse --prompt=\"🌐 SSH: \") && \
    tmux new-window -n \"SSH [\$host\"] \"ssh \$host\" || \
    exit 0
"

############################
### PLUGINS AUTOINSTALL ###
############################
```

```
# Ensure TPM plugin is installed
if "test ! -d ~/.config/tmux/plugins/tpm" \
   "run 'git clone https://github.com/tmux-plugins/tpm ~/.config/tmux/plugins/tpm'"

# Autoinstall other plugins on new machine
run '~/.config/tmux/plugins/tpm/tpm'
run '~/.config/tmux/plugins/tpm/bin/install_plugins'


####################
### PLUGINS INIT ###
####################

# Initialize TPM (keep this line at very bottom of tmux.conf)
run '~/.config/tmux/plugins/tpm/tpm'
```