# kubectl □□ □□ (□□ □)

□□□□□ □□ □□ □□□ □ □□□□□ □□□ □□□ □□ □□ □□□ □ □□□.

□□□ □□□□

□□

```
type _init_completion
```

□□□□ □□□□ □□ □□□ □□ □□□ □□ □□

□□□ □□ □□:

```
type _init_completion
-bash: type: _init_completion: not found
```

□□ □□□:

```
source /usr/share/bash-completion/bash_completion
[root@main ~]# exec bash
```

> □□ □ □□□□ □□ □ 'Not Found' □□ □□ □□, □□ 'bash-completion'□ □ □□ □□ □□ □□ □□□

> RHEL □□:
> dnf install bash-completion
>
> Ubuntu □□:
> apt install bash-completion

□□

```
type _init_completion
_init_completion is a function
_init_completion ()
{
    local exclude= flag outx errx inx OPTIND=1;
    while getopts "n:e:o:i:s" flag "$@"; do
        case $flag in
            n)
                exclude+=$OPTARG
            ;;
            e)
                errx=$OPTARG
            ;;
            o)
                outx=$OPTARG
            ;;
            i)
                inx=$OPTARG
            ;;
            s)
                split=false;
                exclude+==
            ;;
        esac;
    done;
    COMPREPLY=();
    local redir="@(?([0-9])<|?([0-9&])>?(>)|>&)";
    _get_comp_words_by_ref -n "$exclude<>&" cur prev words cword;
    _variables && return 1;
    if [[ $cur == $redir* || $prev == $redir ]]; then
        local xspec;
        case $cur in
            2'>'*)
                xspec=$errx
            ;;
            *'>'*)
                xspec=$outx
            ;;
            *'<'*)
                xspec=$inx
            ;;
            *)
                case $prev in
                    2'>'*)
                        xspec=$errx
                    ;;
```

```
            *'>'*)
                xspec=$outx
            ;;
            *'<'*)
                xspec=$inx
            ;;
        esac
      ;;
    esac;
    cur="${cur##$redir}";
    _filedir $xspec;
    return 1;
  fi;
  local i skip;
  for ((i=1; i < ${#words[@]}; 1))
  do
    if [[ ${words[i]} == $redir* ]]; then
      [[ ${words[i]} == $redir ]] && skip=2 || skip=1;
      words=("${words[@]:0:i}" "${words[@]:i+skip}");
      [[ $i -le $cword ]] && cword=$(( cword - skip ));
    else
      i=$(( ++i ));
    fi;
  done;
  [[ $cword -eq 0 ]] && return 1;
  prev=${words[cword-1]};
  [[ -n ${split-} ]] && _split_longopt && split=true;
  return 0
}
```

□□ □□□□□ □ □□ □□ □□. □□ kubectl □□ □□ □□□□□ □□ □ □□□□□ □□□□□ □□ □□. □□ □□□ □ □□ □ □□ □□□ □□.

□□ □□□□□□ □ □□□

echo 'source <(kubectl completion bash)' >>~/.bashrc

□□ □□□□□□ □□

```
ubectl completion bash | sudo tee /etc/bash_completion.d/kubectl > /dev/null
[root@node1 ~]# exec bash
[root@node1 ~]# kubectl get ns
default         kube-node-lease  kube-public      kube-system
```

[root@node1 ~]# kubectl get ns

// □□□

---