Software & Repositories

DNF	for rpm packages of Red Hat, Fedora and CentOS distributions
dnf list --installed [<application ...>]	lists installed packages
dnf list --available [<application ...>]	lists available packages
dnf list [<application ...>]	lists all packages (installed and available)
dnf group list [<package_group>]	lists all groups of packages (installed and available)
dnf list --upgrades [<application ...>]	lists available updates of installed packages, **--security** security updates only
dnf list --extras [<application ...>]	lists installed packages that are not available in any configured repository
dnf list --recent [<application ...>]	lists packages recently added into the repositories
dnf list --showduplicates [<application ...>]	lists available versions of packages
dnf repoquery --deplist <application ...>	lists the dependencies (files including required packages) that are needed for the specified package
dnf repoquery -l <application ...>	prints the files contained in the installed or available package
dnf search <keyword>	lists packages (installed or available) whose metadata contains the keyword
dnf provides <file>	prints the package (installed or available) to which the file belongs
dnf info <application ...>	prints detailed information about the package
dnf group info <package_group ...>	prints information about the group of packages including dependencies
dnf install --downloadonly --downloaddir=<directory> <application> **dnf download** <application>	downloads the package from the repository
dnf download --resolve <application>	downloads the package from the repository including dependencies
dnf install <application ...>	installs the package including dependencies, **-y** automatically confirms the operation
dnf install <package ...>	installs the package from the specified directory or URL, potential dependencies are resolved by repositories, **-y** automatically confirms the operation
dnf group install <package_group ...>	installs the group of packages including dependencies, **-y** automatically confirms the operation `# dnf group install GNOME`
dnf reinstall <application ...>	reinstalls the package with the same version, **-y** automatically confirms the operation
dnf upgrade	updates all installed packages, **--security** security updates only, **--exclude=**<application> excludes the package from the update process (permanent settings are configured in */etc/dnf/dnf.conf* using the "exclude=<application>" option), **-y** automatically confirms the operation

DNF	**for rpm packages of Red Hat, Fedora and CentOS distributions**
dnf upgrade <application ...>	updates the package, **-y** automatically confirms the operation, **--nogpgcheck** skips checking GPG signatures on packages
dnf upgrade <package ...>	updates the package from the specified directory, possible dependencies are resolved by repositories, **-y** automatically confirms the operation
dnf group upgrade <package_group ...>	updates the group of packages including dependencies, **-y** automatically confirms the operation
dnf downgrade <application ...>	downgrades the current version of the installed package to the previous version, **-y** automatically confirms the operation
dnf remove | **erase** <application ...>	uninstalls the package including dependencies, **-y** automatically confirms the operation
dnf group remove <package_group ...>	uninstalls the group of packages including dependencies, **-y** automatically confirms the operation
dnf check	checks the local RPM database and prints information on any problems it finds, **--dependencies** problems with dependencies between packages only, **--duplicates** problems with duplicate packages only, **--obsoleted** problems with obsolete packages only, **--provides** problems with the files or properties that a package contains only
dnf history	displays the transaction history (data from */var/log/dnf.rpm.log*)
dnf history info <ID>	displays detailed information about the specified transaction
dnf history undo <ID>	reverts all changes within the specified transaction (installs or uninstalls software)
dnf versionlock add <application ...>	locks the current versions of the packages
dnf versionlock delete <application ...>	removes the package version locks
dnf versionlock clear	removes all package version locks
dnf versionlock list	lists all locked package versions
dnf clean all	removes all cached data and temporary files associated with the package management transactions (*/var/cache/dnf*)
dnf repolist	lists enabled repositories (their ID and name), **--enabled** enabled repositories (by default), **--disabled** disabled repositories, **--all** all repositories, **-v** detailed output
dnf --enablerepo=<repo_id>	enables the repository (permanent settings are configured in */etc/yum.repos.d/<repository>*) `# dnf install mplayer* --enablerepo=rpmfusion-free-updates`
dnf --disablerepo=<repo_id>	disables the repository (permanent settings are configured in */etc/yum.repos.d/<repository>*)

YUM	**for rpm packages of Red Hat, Fedora and CentOS distributions**
yum list installed [<application ...>]	lists installed packages
yum list available [<application ...>]	lists available packages
yum list [<application ...>]	lists all packages (installed and available)

YUM	**for rpm packages of Red Hat, Fedora and CentOS distributions**
yum grouplist [<package_group>]	lists all groups of packages (installed and available)
yum list updates [<application ...>]	lists available updates of installed packages, **--security** security updates only
yum list extras [<application ...>]	lists installed packages that are not available in any configured repository
yum list recent [<application ...>]	lists packages recently added into the repositories
yum list --showduplicates [<application ...>]	lists available versions of the packages
yum deplist <application ...>	lists the dependencies (files including required packages) that are needed for the specified package
yum search <keyword>	lists packages (installed or available) whose metadata contains the keyword
yum [**what**]**provides** <file>	prints the package (installed or available) to which the file belongs
yum info <application ...>	prints detailed information about the package
yum groupinfo <package_group ...>	prints information about the group of packages including dependencies
yum install --downloadonly --downloaddir=<directory> <application> **yumdownloader** <application>	downloads the package from the repository
yumdownloader --resolve <application>	downloads the package from the repository including dependencies
yum install <application ...>	installs the package including dependencies, **-y** automatically confirms the operation
yum [**local**]**install** <package ...>	installs the package from the specified directory or URL, potential dependencies are resolved by repositories, **-y** automatically confirms the operation
yum groupinstall <package_group ...>	installs the group of packages including dependencies, **-y** automatically confirms the operation `# yum groupinstall GNOME`
yum reinstall <application ...>	reinstalls the package with the same version, **-y** automatically confirms the operation
yum update	updates all installed packages, **--security** security updates only, **--exclude=**<application> excludes the package from the update process (permanent settings are configured in */etc/yum.conf* using the "exclude=<application>" option), **-y** automatically confirms the operation
yum update <application ...>	updates the package, **-y** automatically confirms the operation, **--nogpgcheck** skips checking GPG signatures on packages
yum [**local**]**update** <package ...>	updates the package from the specified directory, possible dependencies are resolved by repositories, **-y** automatically confirms the operation
yum groupupdate <package_group ...>	updates the group of packages including dependencies, **-y** automatically confirms the operation
yum downgrade <application ...>	downgrades the current version of the installed package to the previous version, **-y** automatically confirms the operation
yum remove | **erase** <application ...>	uninstalls the package including dependencies, **-y** automatically confirms the operation
yum groupremove <package_group ...>	uninstalls the group of packages including dependencies, **-y** automatically confirms the operation

YUM	**for rpm packages of Red Hat, Fedora and CentOS distributions**
yum check	checks the local RPM database and prints information on any problems it finds, **dependencies** problems with dependencies between packages only, **duplicates** problems with duplicate packages only, **obsoleted** problems with obsolete packages only, **provides** problems with the files or properties that a package contains only
yum-complete-transaction	completes failed or aborted transactions
yum history	displays the transaction history (data from */var/log/yum.log*)
yum history info <ID>	displays detailed information about the specified transaction
yum history undo <ID>	reverts all changes within the specified transaction (installs or uninstalls software)
yum versionlock add <application ...>	locks the current versions of the packages
yum versionlock delete <application ...>	removes the package version locks
yum versionlock clear	removes all package version locks
yum versionlock list	lists all locked package versions
yum clean all	removes all cached data and temporary files associated with the package management transactions (*/var/cache/yum/repository/packages*)
yum repolist	lists enabled repositories (their ID and name), **enabled** enabled repositories (by default), **disabled** disabled repositories, **all** all repositories, **-v** detailed output
yum-config-manager --add-repo <repo>	adds and enables the repository `# yum-config-manager --add-repo "https://dl.fedoraproject.org/pub/epel/7/x86_64/"`
yum-config-manager --enable <repo>	enables the repository
yum-config-manager --disable <repo>	disables the repository
yum --enablerepo=<repo>	enables the repository (permanent settings are configured in */etc/yum.repos.d/<repository>*) `# yum install mplayer* --enablerepo=rpmfusion-free-updates`
yum --disablerepo=<repo>	disables the repository (permanent settings are configured in */etc/yum.repos.d/<repository>*)

RPM	**for rpm packages of Red Hat, Fedora, CentOS and SUSE distributions**
rpm -qa	lists installed packages
rpm -q <application ...>	prints the name and version of the installed package
rpm -qi <application ...>	prints detailed information about the installed package
rpm -qip <package ...>	prints detailed information about the uninstalled package
rpm -ql <application ...>	prints the files contained in the installed package
rpm -qlp <package ...>	prints the files contained in the uninstalled package
rpm -qc <application ...>	prints configuration files contained in the installed package
rpm -qcp <package ...>	prints configuration files contained in the uninstalled package
rpm -qf <file ...>	prints the installed package to which the file belongs
rpm -qR <application ... | package ...>	lists the dependencies for the specified application or package

RPM	**for rpm packages of Red Hat, Fedora, CentOS and SUSE distributions**
rpm -qp --scripts <package ...>	prints scripts contained in the uninstalled package (useful before installing a package that does not come from a trusted repository)
rpm -ivh <package ...>	installs the package from the specified directory or URL
rpm -ivh --test <package ...>	checks and reports potential dependencies and conflicts among the packages (useful before installing the package itself)
rpm -ivh --nodeps <package ...>	installs the package without resolving dependencies
rpm -Uvh <package ...>	installs the package or updates an already installed version
rpm -Uvh --replacepkgs <package ...>	reinstalls the package
rpm -Fvh <package ...>	updates the installed package
rpm -e <application ...>	uninstalls the package, **--justdb** removes only the package record in the RPM database `# rpm -e --justdb nomachine`
rpm -ivh <repository ...>	installs the repository `# rpm -ivh rpm.livna.org/release7.rpm`
rpm --import <public_key>	imports the public GPG key of the specified repository to ensure that the package really comes from the original repository `# rpm --import rpm.livna.org/RPM-LIVNA-GPG-KEY`
rpm -qa gpg-pubkey	prints imported public GPG keys
rpm --checksig | **-K** <package ...>	verifies the package signature to ensure its integrity and origin (this happens automatically during the package installation if the required public GPG key has been imported before)
rpm -V <application ...>	verifies information about the installed package files with information in the RPM database
rpm -Vf <file ...>	verifies information about the installed package file with information in the RPM database `# rpm -Vf /bin/ls`
rpm -Va	verifies information about the files of all installed packages with information in the RPM database
rpm --restore <application ...>	restores the owner, group and permissions of the installed package files

UP2DATE	**for rpm packages of older Red Hat distributions**
up2date --show-available	lists available packages
up2date --showall	lists all packages (installed and available)
up2date --show-orphans	lists installed packages that are not available in any configured repository
up2date -l	lists available updates of installed packages
up2date -d	downloads packages to be installed or updated
up2date -i	installs downloaded packages
up2date -u	updates downloaded packages
up2date <application ...>	updates the specified package

UP2DATE	**for rpm packages of older Red Hat distributions**
up2date --whatprovides <file | application>	prints the package to which the file or application belongs
up2date --configure	sets the program options; the configuration file is */etc/sysconfig/rhn/up2date*, the packages are stored in */var/spool/up2date* by default

APT	**for deb packages of Debian and Ubuntu distributions**
dpkg -l	lists installed packages
apt-cache pkgnames	lists all packages (installed and available)
apt-cache depends <application>	lists the dependencies (required packages) that are needed for the specified package
apt-cache whatdepends <application>	lists packages that depend on the specified package
apt-cache show <application>	prints detailed information about the package
apt-cache search <keyword>	lists packages (installed or available) whose metadata contains the keyword
apt-cache policy	lists all installed repositories
apt-file list <application>	prints the files that the package (installed or available) contains
apt-file search <file>	prints the package (installed or available) to which the file belongs
apt-file update	updates the database with information about the packages and their files
apt-get install <application ...>	installs the package including dependencies, **-d** only downloads the package
apt-get update	updates the local list of packages (required before each package update)
apt-get -u upgrade	updates all installed packages without resolving new dependencies, **-u** prints updated packages
apt-get -u dist-upgrade	updates all installed packages including resolving new dependencies (the entire distribution can be updated as well), **-u** prints updated packages
apt-get install --only-upgrade <application ...>	updates only the specified package
apt-get remove <application ...>	uninstalls the package (configuration files remain)
apt-get --purge remove <application ...>	uninstalls the package including configuration files
apt-get check	checks the consistency of the local package database and verifies whether all dependencies for installed packages are met
apt-get clean	removes cached package files associated with the package management transactions (*/var/cache/apt/archives*)
apt-show-versions	prints the versions of all installed packages and possibly their available updates, **-p** <application> applies for the specified package only, **-u** prints only packages with available updates, **-a** prints all available versions of the package

ZYPPER	for rpm packages of SUSE distribution
zypper packages --installed-only	lists installed packages
zypper packages --not-installed-only	lists available packages
zypper packages	lists all packages (installed and available)
zypper list-updates	lists available updates of installed packages
zypper search <keyword>	lists packages (installed or available) whose metadata contains the keyword
zypper what-provides <file>	prints the package (installed or available) to which the file belongs
zypper info <application>	prints detailed information about the package
zypper install --download-only <application ...>	downloads the package from the repository
zypper install <application ...>	installs the package including dependencies
zypper install -f <application ...>	reinstalls the package with the same version
zypper update	updates all installed packages; **zypper al** <application> excludes the package from the update or installation process, **zypper rl** <application> cancels the previous exclusion of the package from the update or installation process, **zypper ll** prints packages that are excluded from the update or installation process
zypper update <application ...>	updates the package
zypper remove <application ...>	uninstalls the package including dependencies
zypper clean --all	removes all cached data and temporary files associated with the package management transactions (*/var/cache/zypp/packages*)
cat /var/log/zypp/history	displays the transaction history
zypper repos	lists all repositories (their ID, name and status)
zypper addrepo <directory> <repo_name>	creates a local repository from the specified directory containing rpm packages

SRC	for compiling packages from the source code
./configure	executes the source package configuration script (from the package directory containing the script) which checks the system, loads the variables and creates associated files needed for the package compilation, **--prefix=**<directory> specifies the directory for the installation (*/usr/local* by default)
make	creates a binary executable program from the source code (compilation) based on data from the associated files created during the package configuration
make install	installs the program
checkinstall	creates a package from the binary program, which is added to the database of installed packages after the installation, **-R** rpm package, **-D** deb package; if no option is specified, it starts in interactive mode
make clean	removes unnecessary, mostly compiled files, created during the compilation
make distclean	removes all files created during the package configuration and compilation

SRC	for compiling packages from the source code
make uninstall	uninstalls the program (if the "Makefile" file exists, otherwise the program files have to be removed manually)

From:
https://prompt.cz/en/ - **Prompt.cz**

Permanent link:
https://prompt.cz/en/software-and-repositories

Last update: **2024/02/18 18:53**