Linux

Package management

Package management overview

A **package manager** or **package-management system** is a collection of software tools that automates the process of installing, upgrading, configuring, and removing computer.

Package managers provide the following benefits:

- Manage software sources
- Eliminate manual installation steps
- Provide consistent updates and upgrades
- Control software footprint

A **software package** is computer software and its metadata packaged in an archived format. It includes:

- Software files with pre-defined paths and permissions
- Dependency list
- Installation scripts

Software sources

Packages are distributed via **package repositories**. Repositories may contain multiple versions of packages. Repositories may be maintained by the Linux distribution developers, software development companies or users.

Maintainer may split repositories based on specific criteria, such as stability **(stable / testing / experimental).**

Adding, removing, enabling and disabling repositories gives users more control over the software that is being installed on their systems.

Package formats

Different distribution of Linux use different packaging system and **package intended for one distribution is not compatible with another distribution**.

Package format	Linux Distributions
APK	Alpine Linux
AppImage	Linux distribution-agnostic
Deb	Debian based distros
Flatpak	All Linux systems
RPM	RHEL based distros
SNAP	Linux + macOS
MSI	Windows

Package managers

- **RPM (Red Hat Package Manager).** Originally developed by Red Hat for Red Hat Linux, RPM is now used by many Linux distributions.
- **YUM (The Yellow dog Updater, Modified).** An open-source command line package management utility for RPM-compatible Linux operating systems.

- **Deb** is the extension of the Debian software package format and the most often used name for such binary packages.
- **APT** or **The Advanced Packaging Tool**, is a free user interface that works with core libraries to handle the installation and removal of software on the Debian distribution and its variants.

Common tasks

• **Finding** a package:	`$ yum search search_string`	`$ apt search search_string`
• **Installing** a package	`$ yum install package_name`	`$ apt install package_name`
• **Removing** a package	`$ yum erase package_name`	`$ apt remove package_name`
• **Updating** packages	`$ yum update`	`$ apt update; apt upgrade`
• **Listing** installed packages	`$ yum list installed`	`$ apt list --installed`
• **Displaying** package information	`$ yum info package_name`	`$ apt show package_name`

Home task

1. Find the list of repositories set up on your system.
2. Add the official Nginx repository and install Nginx.
3. Add the official MySQL Community repository and install the MySQL server package.
4. Set nginx and mysql-server to run on boot, check service status after rebooting the system.
5. Read nginx and mysql-server startup messages with journalctl.

THANK YOU