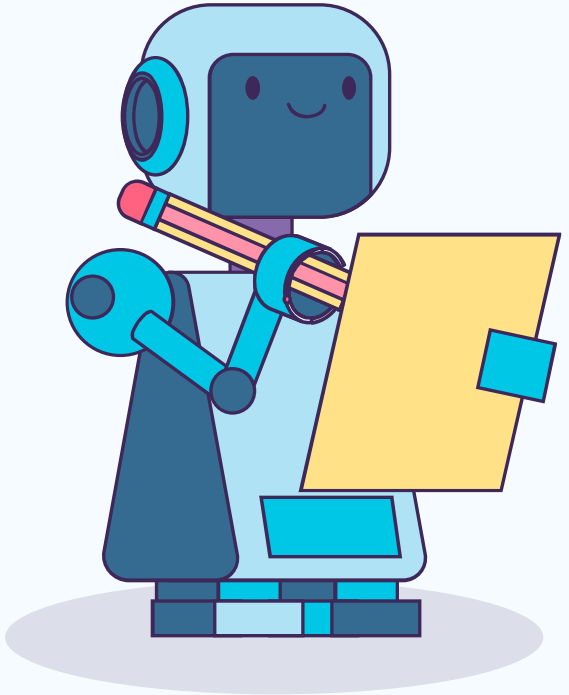
WORK-

CASE №4

Виконали:
Чех Іван
Дзизиль Денис

01

Theoretical material



В ході роботи досить часто виникає необхідність встановлювати нові програми та додатки. Для цього необхідно в терміналі вміти працювати з менеджерами пакетів

Дайте розгорнуте визначення таким поняттям як «пакет» та «репозиторій»

Package

A package is like a treasure chest, filled with files, software code, documentation, libraries, and more. It's all bundled together for a specific software product or component. Packages make it easy to share and install software on different operating systems and development environments. Inside, you'll find executable code, configuration files, and all the bits and bobs needed for the program to work smoothly. Packages usually come in neat, standardized formats, making them a breeze to install, update, and manage.

Repository

A repository, on the other hand, is like a secret hideout for software. It's a place - either central or scattered - where software code, files, change history, and other info about a software product's development are tucked away. Repositories are superheroes in the world of version control systems like Git, Subversion, Mercurial, and more. They let developers join forces on projects, keep an eye on code changes, lend a hand, and maintain a record of the software's life story. Repositories can be open for all to see or locked up. They're the ultimate team players in the world of software development, fostering collaboration among developers.

Надайте короткий огляд існуючих менеджерів пакетів у Linux. Охарактеризуйте їх основні можливості.

Linux provides several package managers to help users manage programs and their dependencies. In this article, we'll give you a quick overview of some of the most popular package managers, along with their key features:

APT (Advanced Package Tool) 💼:
APT is used by Debian and many Debian-derived distributions such as Ubuntu. It provides a wide selection of packages in repositories and uses commands like apt-get and apt to install, update, and remove packages. 📦

YUM (Yellowdog Updater Modified) 🐶:
YUM is used in Red Hat-based distributions such as CentOS and Fedora. It provides powerful package management and dependency resolution and uses commands like yum to manage packages. 🍽️

Pac Man 🕹️:
Pac Man is used in Arch Linux and other Arch-derivative distributions. It's a simple and fast package manager with a strong emphasis on simplicity and speed. It uses commands like pacman to install and manage packages. 🍒

dnf (Dandified YUM) 🌀:
dnf is used in modern versions of Fedora and CentOS. It's an evolution of YUM and provides better performance and speed. It uses commands such as dnf to manage packages. 🚀

ZYpp (Zenworks Package Management) ☯️:
ZYpp is used in SUSE distributions such as openSUSE. It supports package management, dependency resolution, and system updates. It uses commands like zypper to manage packages. 🧘

Portage 🐃:
Portage is used in Gentoo Linux. It's based on the concept of "ephemeral packages," where packages are built from source codes. It uses commands like emerge to manage packages. 🌿

Визначте який менеджер пакетів використовує ваш дистрибутив Linux. Опишіть основні команди для роботи з ним

My Linux distribution uses yum

To find, download and install the necessary packages using the YUM package manager, you can use the following commands:

Searching for a package in repositories 🔍:

Use the search command to search for a package by keyword, name, or description:

```
sudo yum search  package_name
```

Example:

```
sudo yum search nginx
```

// This command will list the packages that match your query.

Installing the package :

Once you've found the package you need, install it using the install command:

```
sudo yum install package_name
```

Where package_name is the name of the package you want to install.

Updating the list of repositories 🔄:

Make sure your repository list is up-to-date before searching for and installing packages :

```
sudo yum update
```

Installing a package from another repository 🏗:

If the package is not in the default repository, you can add a new repository and install the package from there.
Add the repository, for example:

```
sudo yum-config-manager --add-repo repository_url
```

Where repository_url is the URL of the repository.
After adding the repository, install the package:

```
sudo yum install package_name
```

Downloading the package without installation 💾🚫:

Sometimes you may need to download a package without installing it. Use the download command:

```
sudo yum install --downloadonly package_name
```

The package will be downloaded but not installed on the system.

View package information :

Display information about the installed package:

```
yum information about installed_package_name
```

Display information about the available package:

```
yum info available_package_name
```

Removing packages  :

Release of the installed package

```
sudo yum remove package_name
```

Update the YUM package manager itself if new versions are available 🔄🔧:

```
sudo yum update yum
```