CASE1 : install Apache Tomcat as a service and create a dedicated system user to manage all Tomcat-related operations

- Create a system user: This command creates a new system user named tomcat

  sudo adduser --system --shell /bin/false --group tomcat

  ❖ sudo: Allows the command to be run with superuserprivileges.
  ❖ adduser: Command to add a new user.
  ❖ --system: Indicates that this is a system user.
  ❖ --shell /bin/false: Specifies that this user does not have a login shell, enhancing security.
  ❖ --group: Creates a new group with the same name as the user.
  ❖ tomcat : name of the user

- Download and extract Tomcat: This command downloads the Apache Tomcat distribution and extracts it to /opt/tomcat:

  wget -P /tmp
  https://downloads.apache.org/tomcat/tomcat9/v9.0.50/bin/apache-tomcat-9.0.50.tar.gz

  sudo mkdir /opt/tomcat

  sudo tar xzvf /tmp/apache-tomcat-9.0.50.tar.gz -C /opt/tomcat --strip-components=1

  ❖ wget: Command-line utility for downloading files from the web.
  ❖ -P /tmp: Specifies the directory where the downloaded file will be saved.
  ❖ mkdir: Creates a new directory.
  ❖ tar: Command-line utility for compressing or decompressing files.
  ❖ xzvf: Flags to extract a tarball.
  ❖ /tmp/apache-tomcat-9.0.50.tar.gz: Path to the downloaded Tomcat archive.
  ❖ -C /opt/tomcat: Specifies the directory where the contents will be extracted.
  ❖ --strip-components=1: Removes the first component from the file paths in the archive, simplifying the directory structure.

- Change ownership and permissions: This command changes ownership of the Tomcat directory to the tomcat user and group:

  sudo chown -R tomcat: /opt/tomcat

  ❖ chown: Command to change file ownership.
  ❖ -R: Recursively changes ownership of all files and directories within /opt/tomcat.
  ❖ tomcat:: Specifies the user and group to which ownership is changed.

- Configure Tomcat: Edit the Tomcat configuration files located in /opt/tomcat/conf/ to adjust settings as needed. Common configuration files include server.xml, web.xml, etc.

- Create a systemd service unit file: Create a new file named tomcat.service in /etc/systemd/system/ directory with the following content:

```
[Unit]
Description=Apache Tomcat Web Application Container
After=network.target
[Service]
Type=forking
User=tomcat
Group=tomcat
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
SuccessExitStatus=143
StandardOutput=null
[Install]
WantedBy=multi-user.target
```

- ❖ Change file paths according to your installation
- ❖ This unit file specifies the systemd service configuration for Tomcat.
- ❖ It sets up environment variables such as CATALINA_PID, CATALINA_HOME, and CATALINA_BASE.
- ❖ ExecStart and ExecStop define the commands to start and stop Tomcat respectively.
- ❖ SuccessExitStatus=143 ensures that the service is considered successfully stopped when it exits with status 143.
- ❖ StandardOutput=null redirects standard output to null to prevent it from cluttering the system logs.

- Reload systemd: After creating the service unit file, reload systemd to apply the changes:

```
sudo systemctl daemon-reload
```

- ❖ systemctl: Command to control the systemd system and service manager.
- ❖ daemon-reload: Reloads systemd to read the latest configuration changes.

- Enable the Tomcat service: This command enables the Tomcat service to start automatically on system boot:

```
sudo systemctl enable tomcat
```

❖ enable: Command to enable a service to start automatically.

● Start the Tomcat service: Finally, start the Tomcat service:

```
sudo systemctl start tomcat
```

❖ start: Command to start a service.