**・コマンド履歴を削除する(　history 　–c　)**

**●実習１２－１．シェル環境**

**（１）エイリアス**

① エイリアスを表示する　　　　　　　　　　　　　　　　　　　**alias**

画面表示内容　エイリアス「 ll 」の行のみ記載→　　　**alias　ll＝'ls –alF’**

②　エイリアス「 ll 」を実行　 　 　**ll**

③　ホームディレクトリのパスを表示するコマンド「　echo $HOME　」を実行できる「home」という名前の

エイリアスを作成する。

　　　入力コマンド　→　　　**alias home='echo $HOME'**　　（＝の前後スペースあけない）

④　エイリアスを表示し、③で作成したエイリアス「home」が登録されていることを確認する

（①と同じコマンド実行）

⑤　エイリアス「 home 」を実行

　　入力コマンド　→　　**home**  　 画面表示内容例　→　　**/home/ie2a99**

⑥ エイリアスの表示結果をリダイレクト機能により、「　alias.kad12　」の名前で保存する

入力コマンド　→　　　**alias**　**> alias.kad12**

⑦ 「　alias.kad12　」の内容を確認する　  **cat　alias.kad12**

⑧　「Tera Term」→「新しい接続」→「IPアドレス」で別のウィンドウを開く

⑨　もう一つの端末でエイリアス「 home 」を実行 コマンドhomeが見つかりません...

　※エイリアスは設定したシェルでしか使えない

⑩　もう一つの端末を閉じる  **exit**

**（２）シェル変数・環境変数**

①　変数HOSTNAMEの内容を表示する

**入力コマンド　→　　　echo $HOSTNAME　　　画面表示内容例　→　　　ie2a99**

②　「　linux 」という名前の変数に　「　ubuntu 」の文字を設定する。

入力コマンド　→　　**linux=ubuntu**

③　②で作成した、変数「 linux 」の内容を表示する

　入力コマンド　→　　　　**echo $linux**　　　　　　　 画面表示内容　→　　　**ubuntu**

④　もう一つbashを起動する 　 　**bash**

⑤　変数「 linux 」の内容を表示する

　入力コマンド　→　　　　**echo $linux**　　　　　　　 画面表示内容　→　　　

　※シェル変数は設定したシェルでしか使えない

⑥　もう一つのbashを閉じる 　 　**exit**

⑦　②で作成した、変数「 linux 」を環境変数に設定する

　入力コマンド　→　　　　**export linux**

⑧　設定されている環境変数を表示する

入力コマンド　→　　　　 **printenv　| grep linux**

画面表示内容中、④で設定した環境変数の行の内容　→　　**linux=ubuntu**

⑨ 環境変数の表示結果をリダイレクト機能により、「　printenv.kad12　」の名前で保存する

入力コマンド　→　　　**printenv**　**> printenv.kad12**

⑩ 「 printenv.kad12　」の内容を確認する  **cat　 printenv.kad12**

⑪　もう一つbashを起動する 　 　**bash**

⑫　もう一つのbashで変数「 linux 」の内容を表示する

　入力コマンド　→　　　　**echo $linux**　　　　　　　 画面表示内容　→　　　**ubuntu**

　　※環境変数にexportしたら子シェルでも使えるようになる

⑬　もう一つのbashを閉じる 　 　**exit**

**●実習１２－２．アップデートの確認とアップデート**

**（１）アップデートの確認**

①　アップデートリストを確認する。　　　　 **apt list --upgradable**

②　systemdのバージョンを確認する  **sudo systemctl --version**

表示されたバージョン、リリース番号　　→　  **249.11-0ubuntu3.9**

③　rsyslogのバージョンを確認する  **rsyslogd -version**

表示されたバージョン、リリース番号　　→　  **8.2112.0**

　④　アップデートリストを更新する　　　　　　　　　 **sudo apt update**

⑤　アップデートを実行する。 **sudo apt -y upgrade**

⑥　rsyslogのバージョンを確認する  **rsyslogd -version**

表示されたバージョン、リリース番号　　→　  **8.2112.0**

**●実習１２－３．パッケージインストール**

**（１）コマンド操作練習１**

①パッケージのリストをアップデートする　入力コマンド→　**sudo apt update**

②パッケージをアップデートする　入力コマンド→　**sudo apt upgrade -y**

　③インストール可能なmysqlのバージョンを確認するコマンドを実行する。

　　　　　入力コマンド→ 　　**sudo apt show mysql-server**

　④インストールできるmysqlのバージョン　 **mysql-8.0**

　⑤mysqlをインストールする　 入力コマンド→　 　 **sudo apt install -y mysql-server**

　⑥mysqlを起動する　　 入力コマンド→　  **sudo systemctl start mysql**

　⑦mysqlのバージョンを確認するコマンドを実行する 　　 **mysql –-version**

**●実習１２－４．Webサーバーを立てる**

**（１）Apacheをインストールする**

①Apacheをインストールする（パッケージ名：**apache2**）

Sudo apt install -y apache2

②Apacheを起動する　　　　　　　　　　　　　　　 **sudo　　systemctl start apache2**

③Apacheの状態を確認する　　　　　　　　　　　　 **sudo　　systemctl status apache2**

　　→**active**になっていること！　→ｑｷｰ入力で　コマンド入力に戻る

④仮想マシンのIPアドレスを確認する　　　　　　　 **ip a**

192.168.64.4

　　→IPアドレス

⑤ブラウザの表示　（メニューより選択）

**サーバ側「****Firefox　ウェブ・ブラウザ」クリック**

⑥アドレスバーに下記URLを入力

　　　http://④で調べたIPアドレス/

**カレンダー が含まれている画像

自動的に生成された説明**

**●実習１２－５　ジョブ演習**

（１）ジョブの実行モードの動きを確認する

①　バックグラウンドでジョブを６０秒間一時停止する。

　　　入力コマンド　→　　**sleep 60 &**

　　　 画面表示内容　→　　　　**[1] 2766　　(数字は、各マシンによって違う)**

②　ジョブを表示する。

入力コマンド　→　　　　　**jobs**

　画面表示内容　→　　　　**[1]+ 実行中 sleep 60 &**

（「　[1]+ Done ････ 」と表示された場合は、ジョブが終了しているので①よりもう一度行う）

③　①で起動したバックグラウンドのジョブをフォアグラウンドへ移す。

入力コマンド　→　　　　　**fg**　　**または　fg　%1**

　　　 画面表示内容　→　 　　　**sleep　 60**

（何も表示されない場合は、60秒間経過しジョブが終了していると思われるので、①よりもう一度行う）

④　60秒後プロンプトが表示された後、ジョブを表示する。

入力コマンド　→　　 　　**jobs**

　 　　 画面表示内容　→　　何も表示されない

**（２）**ジョブの実行モードの動きを確認し、強制終了する

①　3000秒一時停止する

　　　入力コマンド　→　　**sleep 3000**

　②　control＋Zを押して、ジョブを一時中断する

　画面表示内容　→　　　　**[1]+ 停止 sleep 3000**

③　jobを表示する

入力コマンド　→　　 　　**jobs**

　画面表示内容　→　　　　**[1]+ 停止 sleep 3000**

④　一時停止したジョブをバックグラウンドで実行する。

入力コマンド　→　　　　　**bg**　　**または　bg　%1**

　画面表示内容　→　 　　 **[1]+ sleep 3000 &**

⑤　ジョブを表示する。

入力コマンド　→　　 　　**jobs**

　画面表示内容　→　　　　**[1]+ 実行中 sleep 3000 &**

⑥　プロセスを表示する。

入力コマンド　→　　　　　**ps**

　　　画面表示内容　→　 PID TTY TIME CMD

2778 pts/0 00:00:00 su

2785 pts/0 00:00:00 bash

2829 pts/0 00:00:00 sleep

　　　　　　　　　　　　2838 pts/0 00:00:00 ps

⑦　バックグラウンドで実行中のsleepプロセスに通常に終了する信号を送信する。

入力コマンド　→　　　　　**kill 15　(表示されたsleepのプロセスID：例では2829)**

　⑧　プロセスを表示する。

　入力コマンド　→　　　　　**ps**

　　　画面表示内容　→　 PID TTY TIME CMD

2778 pts/0 00:00:00 su

2785 pts/0 00:00:00 bash

2838 pts/0 00:00:00 ps

　 [1]+ Terminated sleep 3000

⑨　システム上で実行されている全てのプロセスを表示する。

入力コマンド　→　　　　　**ps aux**

　※下記実行前に（top実行中の場合qキーを押して止め）ssh　の設定（/etc/ssh/sshd\_config ）において

　AllowUsers XXXXを無効（左端に#）、PasswordAuthentication　no を無効（左端に# ※yesの場合はその

まま）に変更し、sshサービス再起動(sudo systemctl restart ssh)

⑩　プロセスの実行状況を監視する。　入力コマンド　→　　　　　**top**

⑪　TeraTerm「新しい接続」を使用し、SSH接続し、chiroでログインを試みる（パスワードはchpass01）

⑫　監視しているプロセスの実行状況にUSERがchiroのものが増えたことを確認する。

　　（矢印キーの上下でスクロールして確認する）

　　 PID USER COMMAND

　画面表示内容例（抜粋）　→　　　**3454 chiro bash**

⑬　qキーを押してプロセスの実行状況の監視を終了する。chiroでログインのTeraTermも閉じる。

**●実習１２－６　サーバー起動状態の確認**

（１）　サーバーを構築する準備として、起動時に設定されている状態を確認する。

　　①　ランレベルを確認する。

入力コマンド→  **runlevel**　　　　　　　→　　画面表示内容→　　　　　　**N 5**

②　ターゲット（target）状態を確認する。

入力コマンド→  **systemctl get-default**

画面表示内容→　　　**graphical.target**

③　システム起動時のサービス全体状況を　(１画面（ページ）単位で表示する)

　　入力コマンド　→　　 　　**systemctl list-unit-files | less**

④　システム起動時のサービス全体状況の結果から「ｓｓｈｄ」　のみ表示する)

　　　　　入力コマンド　→　　 　　**systemctl list-unit-files | grep ssh**

（２）「ssh」　サービス（ssh　デーモン）の起動状況を確認し、変更します。

①　「ｓｓｈ」　サービス（ssh　デーモン）のみのシステム起動時の設定状況を確認します。

入力コマンド　→　 　 **systemctl　　is-enabled　ssh**

・画面表示内容→　　 **enabled**

　 ②　sshサービス（ssh　デーモン）の状態を確認する。

入力コマンド　→　 　 **systemctl　　is-active 　ssh**

・画面表示内容→　　 **active**

③　sshサービス（ssh　デーモン）を停止する。

　　　　　　入力コマンド　→　　　　　　**sudo systemctl　　stop 　ssh**

④　sshサービス（ssh　デーモン）の状態を確認する。（コマンドは②と同じ）

　　　　　　　　　　　　　　　　　　　　　　　　　・画面表示内容→　　 **inactive**

　　　　・「ssh」　サービス（ssh　デーモン）のみのシステム起動時の設定状況を確認します。

　（コマンドは①と同じ）

・画面表示内容→　　 **enabled**

⑤　「ssh」　サービス（ssh　デーモン）、システム起動時の設定を無効にし、設定状況を確認する。

入力コマンド　→　 　 **sudo systemctl　　　disable 　ssh**　　　　　　　　　　　　　　　　　　

　 「ssh」　サービス（ssh　デーモン）のみのシステム起動時の設定状況を確認します。

　（コマンドは①と同じ）

・画面表示内容→　　　　**disabled**

⑥　「ssh」　サービス（sshd　デーモン）、システム起動時の設定を有効にし、設定状況を確認する。

入力コマンド　→　 　 **sudo systemctl　　　enable 　ssh**　　　 　　　　　　　　　　　　　　　

　 「ssh」　サービス（sshd　デーモン）のみのシステム起動時の設定状況を確認します。

　（コマンドは①と同じ） ・画面表示内容→　　　　**enabled**

**●課題１２－７．ftpサーバーの構築と動作確認**

**（１）　ftpサーバーの構築と起動**

①ftpサーバーをインストールする（パッケージ名：**vsftpd**）

**sudo apt install -y vsftpd**

②ftpサーバーを起動する （パッケージ名：**vsftpd**）

**sudo　systemctl start vsftpd**

③ftpサーバーの状態を確認する（パッケージ名：**vsftpd**）

**sudo　　systemctl status vsftpd**

**→activeになっていること！**

**（２）　Ａｐａｃｈｅの起動＆起動確認**

**sudo　systemctl start apache2**

①Apacheを起動する（パッケージ名：**apache2**）

②Apacheの状態を確認する

**sudo　systemctl status apache2**

**→activeになっていること！**

③Apacheをシステム起動時に起動する設定にする

runlevel

**（３）　ドキュメントルートの確認**

①Apacheの設定ファイルからDocumentRoot

（ドキュメントルート：Web上に公開できるディレクトリのトップ）を検索

**grep DocumentRoot /etc/apache2/sites-enabled/000-default.conf**

　　→DocumentRootの値

/var/www/html

②ドキュメントルートのパーミション、所有者、確認

ls -ld /var/www/html

　　→パーミッション、所有者　　　　　　　　　　  **drwxr-xr-x root root**

③ドキュメントルートのディレクトリをホームディレクトリとする、データ管理ユーザーを作成

　ユーザー名は、webne、id番号は、400とする （パスワードは、ecc）

**sudo adduser –-home 　/var/www/html --no-create-home --uid 400 webne**

**又は　sudo useradd –d 　/var/www/html –u 400 –M webne**

④　③をuseraddでした場合は、上記作成ユーザー（webne）のパスワード作成（パスワードは、ecc）

Sudo passwd webne

⑤ドキュメントルートのディレクトリ以下すべての所有者、所有グループをwebneに変更

Sudo chown -R webne:webne /var/www/html

⑥ドキュメントルートのパーミション、所有者、確認

ls -ld /var/www/html

　　→パーミッション、所有者　　　　　　　　　 　  **rwxr-xr-x webne webne**

※下記実行前に　sudo vim /etc/vsftpd.conf　→　# write\_enable=YES   →　write\_enable=YES　コメント#を削除

ftpサーバーを再起動する（パッケージ名：**vsftpd**）

**（４）　ftpサーバへのアクセス**

①Windowsでfilezillaを起動し、仮想マシンのIPアドレスに接続する

　[新規ホスト]-ホストの設定名：UNIXSv（なんでもいい）

ホスト名：仮想マシンのIPアドレス　ユーザー名：webne パスワード：ecc

②作成者がわかる内容のhtmlファイルを作成し、転送する(例kad12.html （クラス名出席番号入れる等））

③ブラウザからアクセスして、②で入れたhtmlファイルを表示させる

<http://仮想マシンのIPアドレス/html>ファイル名

**●課題提出　下記**3つのファイルをsv23へ提出する。

（1）Tera Termをいったん終了して、本日のログが記録されているか確認する。

ファイル名：クラス番号日付.log

（2）作成したファイルalias.kad12、printenv.kad12を表示してコピー、PC上でペーストして保存する。