

UNIXサーバー構築

第8章 簡単なシェルプログラミング 1

シェル環境

■エイリアス

頻繁に使うコマンドに名前(別名)を付ける機能

■エイリアス作成

alias 作成する別名=‘コマンド’

(例) `alias dir = 'ls -l'`

`dir /home` (`ls -l /home` と同じ)

■エイリアスの表示

alias

シェル環境

■エイリアスの削除

`unalias` エイリアス名 (エイリアスを削除)

■エイリアスの保存

エイリアスは、シェル(bash)プロセスがメモリ上に保存
シェルの終了(ログアウト)すると設定は失われる

`~/.bashrc` にエイリアスの設定を記述しておくと
永続的に使用できる

シェル変数と環境変数

■ シェル変数

コマンドの実行時に参照や代入が行うことができる変数のこと。

■ 主なシェル変数

HOME : ユーザーのホームディレクトリ

LANG : ユーザーの言語環境

PATH : コマンドサーチパス

PS1 : プロンプトの書式

HOSTNAME : ホスト名 (コンピュータ名) 等

シェル変数と環境変数

■変数に値を設定

変数名=値 (例) `var=Linux`

(注) **=の前後にスペースを入れない**

■変数の内容を表示

`echo $変数名` (例) `echo $var`

*変数の有効範囲は、変数を定義したシェルの中のみ

シェル変数と環境変数

■環境変数

変数を定義したシェル上、及びそのシェル上で実行されるプログラムにも引き継がれる変数

■環境変数の設定

変数を**エクスポート**すると環境変数になる

export 変数名=値

(例) **export** VR1=Ubuntu

シェル変数と環境変数

- 設定されている環境変数を表示

printenv

- 設定されている、シェル変数と環境変数をすべて表示

set

シェルスクリプト

複数のコマンドを実行する手順を記述したもの

シェルは、シェルスクリプトという方法で、実行させるコマンドを列挙させることができます

特定の条件での実行など体系的に制御ができます

これをシェルスクリプトプログラミングといいます

シェルスクリプトの作成1

(例1)文字列の表示

#!/bin/bash

echo "Hello Shell!"

- ... 使用するシェル指定
- ... echoは表示(PHPと同じです)

(例2)変数の表示

#!/bin/bash

name='野島'

echo \$name

echo "\${name}先生、こんにちは！"

- ... 入力する時は\$不要
- ... 参照する時は\$変数
- ... 文字列と連結

シェルスクリプトの実行

シェルスクリプトの実行方法にはいくつかの方法ある。

■ コマンドを使用して実行

sh シェルスクリプト名

bash シェルスクリプト名

■ ファイル名での実行

./シェルスクリプト名

*** chmod +x シェルスクリプト名で実行権限を与える必要がある。**

シェルスクリプトの作成2

(例3)入力文字列の表示

`#!/bin/bash`

`read name`

`echo "${name}さん、こんにちは"`

・・・入力されたものを変数へ

シェルスクリプトの作成3

(例4)if文

```
#!/bin/bash
```

```
echo 'Please input h or b.:'
```

```
read a
```

```
if [ $a = "h" ]
```

```
    then echo "Hello"
```

```
elif [ $a = "b" ]
```

```
    then echo "Bye bye"
```

```
else
```

```
    echo "NG"
```

```
fi
```

... [条件式]

... 条件一致の場合は**then**が必要

... **else if**ではないので注意

... if文の終了

*** 他の言語とは違う所が多いので注意してください。**

シェルスクリプトの作成4

(例5)ループ処理(for)

`#!/bin/bash`

```
for i in $@      ... $@は特殊変数(すべての引数を取得)
do
    echo $i      ... 繰り返し実行されるコマンド
done
```

* ループ処理には他の言語と同様にwhile文、until文も使用することができます。

* **in**の後には配列のようなリスト形式のものを配置します。

シェルスクリプトの作成5

(例6)正規表現

#!/bin/bash

read x

if [[\$x =~ ^[0-9]]] ...xが0～9|にいずれかで始まる

 then echo '数字が入力'

elif [[\$x =~ ^[a-z]]] ...xがaからz|にいずれかで始まる

 then echo 'アルファベットが入力'

else

 echo 'その他が入力'

fi

* [変数=~ 正規表現]は、変数の正規表現でチェック。

シェルスクリプトの作成6

(例7)関数

#!/bin/bash

```
hello(){  
    echo "Hello $1."  
}
```

... 関数の定義
... 第1引数

echo 'Input your name?'

read name

hello \$name

... 関数の呼び出し、関数名 引数