

参考

<http://technique.sonots.com/?UNIX%2F%E5%9F%BA%E7%A4%8E%E7%9F%A5%E8%AD%98%2F%E3%83%AA%E3%83%80%E3%82%A4%E3%83%AC%E3%82%AF%E3%83%88%E3%80%81%E3%83%91%E3%82%A4%E3%83%97>

<http://sonic64.com/2004-03-28.html>

<https://qiita.com/laikuaut/items/e1cc312ffc7ec2c872fc>

まとめ

基本

記述	意味
>	リダイレクト。標準出力をファイルに出力
>>	標準出力をファイルへ追記
<	ファイルの内容を標準入力にする
	パイプ。コマンドの出力結果を次のコマンドの標準入力にする

エラー関連

記述	意味
command >log.txt 2>&1	log.txt へ標準出力と標準エラーを出力
command &> log.txt	上記と同じ
command &>> log.txt	上記と同じ（追記）
command 1> stdout.log 2> stderr.log	標準出力を stdout.log へ、標準エラーを stderr.log へ出力
command tee -a log.txt	標準出力を log.txt と画面へ出力
command & tee -a log.txt	標準出力と標準エラーを log.txt と画面へ出力

exec

exec にコマンドを引数にせずにリダイレクトのみを指定すると現在のシェルのリダイレクト先を変更できる

記述	意味
exec > log.txt	以降の標準出力を log.txt にする（最初にファイルをクリア）
exec >> log.txt	以降の標準出力を log.txt にする（最初から追記）
exec > /dev/tty	以降の標準出力を画面にする
exec 2> log.txt	以降の標準エラーを log.txt にする
exec &>> log.txt	以降の標準出力、標準エラーを log.txt にする

exec &> >(tee -a log.txt)	以降の標準出力、標準エラーを log.txt と画面に出力する
---------------------------	---------------------------------

説明

sh, bash 系での標準エラー出力の扱い

1 = 標準出力
2 = 標準エラー出力

のように番号づけされています。

標準出力と標準エラー出力を両方ともファイルに書き出すには

```
% [command] > [filename] 2>&1
```

のようにします。順番が微妙に逆なことに注意。() を使うのなら

```
% ( [command] 2>&1 ) > [filename]
```

ほかのコマンドに標準エラー出力と標準出力両方をパイプで渡したいなら、

```
% [command] 2>&1 | [command]
```

csh, tcsh 系での標準エラー出力の扱い

標準エラー出力もファイルに出力する

```
% [command] >& [filename]
```

パイプも同様

```
% [command] |& [command]
```

/dev/null に出力

```
% [command] >& /dev/null
```

time コマンドも含めたりダイレクト

http://blog.livedoor.jp/hack_cripple/archives/50800592.html

```
$ (time COMMAND) 2>&1 > output
```