

UNIX/Linux①入門編 テキスト

2020 年 7 月版

明治大学

生田メディア支援事務室

はじめに

本テキストは、UNIX/Linux を初めて使う人のためのテキストです。UNIX/Linux 環境は、生田キャンパス内で使用できる CentOS を前提としています。

生田情報メディア HP に『UNIX 利用の手引き』がありますので、必要に応じてそちらも参照してください。

<http://www.meiji.ac.jp/isys/doc/UNIX2019.pdf>

目次

1. UNIX/Linux の基礎知識.....	4
1.1. UNIX とは	4
1.2. Linux とは	4
2. UNIX/Linux のファイルシステム.....	6
2.1. ファイルとディレクトリ	6
2.2. ファイルについて.....	7
2.3. ディレクトリについて	7
2.3.1. ホームディレクトリとカレントディレクトリ	7
2.3.2. ディレクトリの移動.....	8
2.4. 絶対パスと相対パス	8
2.4.1. 絶対パスとは.....	8
2.4.2. 相対パスとは.....	8
2.5. ファイル操作命令.....	9
2.6. ディレクトリ操作命令	10
3. CentOS の使い方	11
3.1. CentOS の起動.....	11
3.2. 各画面の説明.....	13
3.2.1. アプリケーションメニューの説明.....	14
3.2.2. 場所メニューの説明	15
3.2.3. システムメニューの説明.....	16
3.3. CentOS で操作できるソフトウェア	17
3.3.1. インターネットブラウザを使う	17
3.3.2. テキストエディターを使う	19
3.3.3. ソフトウェアの紹介	22
3.4. CentOS の終了.....	27

4. コマンド操作とマウス操作の比較.....	28
4.1. ホームディレクトリに移動.....	29
4.2. ディレクトリにあるファイルやディレクトリを表示.....	31
5. シェルの機能.....	32
5.1. コマンドシェル.....	32
5.2. 入出力の切り替え.....	33
5.2.1. リダイレクト機能.....	33
5.2.2. パイプ機能.....	35
6. UNIX/Linux②基礎編に向けて.....	36
付録.....	37
Mozilla Firefox が起動しない場合.....	37
プロセス、ジョブ、バックグラウンド処理について.....	37
ファイル、ディレクトリ操作について.....	38
ファイルやディレクトリ操作の基本.....	39
参考文献.....	40

1. UNIX/Linux の基礎知識

1.1. UNIX とは

UNIX は、1969 年に AT&T のベル研究所で Ken Thompson、Dennis Ritchie らが開発を開始したオペレーティングシステム(OS)のことです。現在は The Open Group という団体によって管理されています。

1.2. Linux とは

Linux は、1991 年に Linus Torvalds が開発したオペレーティングシステム(OS)です。Linux は UNIX と比較して似た動作をする OS ですが、厳密には UNIX とは別物です。Linux は誰でも自由に無償で使えることを観点に置いています。Linux は企業や学校が使うシステムに採用されているだけではなく、一般的なユーザも気軽に使うことができます。本テキスト、UNIX/Linux 基礎編・中級編、UNIX/Linux 管理者編それぞれのテキストで扱う基本操作においては、UNIX と Linux の差異を厳密に考慮する必要はないため、今後、両者をまとめて UNIX/Linux と表記します。両者の区別が重要な場合は、別途説明します。

以下に代表的な UNIX/Linux を紹介します。

表 1 UNIX/Linux 一覧

UNIX/Linux の種類	説明	分類 ¹
HP-UX	HP 社の UNIX	UNIX
IBM AIX	IBM 社の UNIX	UNIX
Solaris	Oracle 社の UNIX	UNIX
Red Hat Enterprise Linux	民間企業や研究機関などで広く用いられている商用 OS。RedHat 社が販売している	RedHat Linux 系
Fedora	最新技術を積極的に取り入れると言われているディストリビューション	RedHat Linux 系
CentOS	一般に安定性を重視していると言われている。Fedora や CentOS は、Red Hat Enterprise Linux 派生ディストリビューションと呼ばれる	RedHat Linux 系

¹ Linux ディストリビューション(Linux を使用する上で必要な OS やソフトウェアをひとまとめにしたもの)は、開発の歴史から、(1) RedHat Linux 系、(2) Debian 系、(3) Slackware 系、(4) 独立系に大別できる。

UNIX/Linux の種類	説明	分類 ¹
Debian	「100%フリーソフトウェアが理念である OS」として、広く用いられている。多くの開発者がプロジェクトに参加している	Debian 系
Ubuntu	Debian の派生ディストリビューションである。「誰にでも使いやすい最新かつ安定した OS」として広く用いられている	Debian 系
Linux Mint	ユーザビリティを意識したディストリビューションである。複数のデスクトップ環境を選択することができる	Debian 系
Slackware	長い歴史を持つディストリビューション。RedHat 系、Debian 系と並び Linux ディストリビューションとして受け入れられてきた	Slackware 系
OpenSUSE	商用 OS である SUSE Linux Enterprise をベースにコミュニティが開発した Linux ディストリビューション	Slackware 系
GentooLinux	Portage というパッケージ管理システムを採用している Linux ディストリビューション	独立系
ArchLinux	簡潔さや汎用性を重視されている Linux ディストリビューション	独立系

2. UNIX/Linux のファイルシステム

ファイルシステムとは、ファイルを管理する方式のことをいいます。
本章では、UNIX/Linux のファイルシステムについて説明します。

2.1. ファイルとディレクトリ

UNIX/Linux は、プログラムや文章、画像など全てをファイルという単位で管理しています。これに加えて、多数のファイルを効率良く管理するために、ディレクトリという仕組みも用意しています。

ファイルとディレクトリは、Windows や MacOS などのファイルとフォルダの概念とほぼ同じです。

UNIX/Linux のディレクトリとファイルは、下図のような構造になっています。

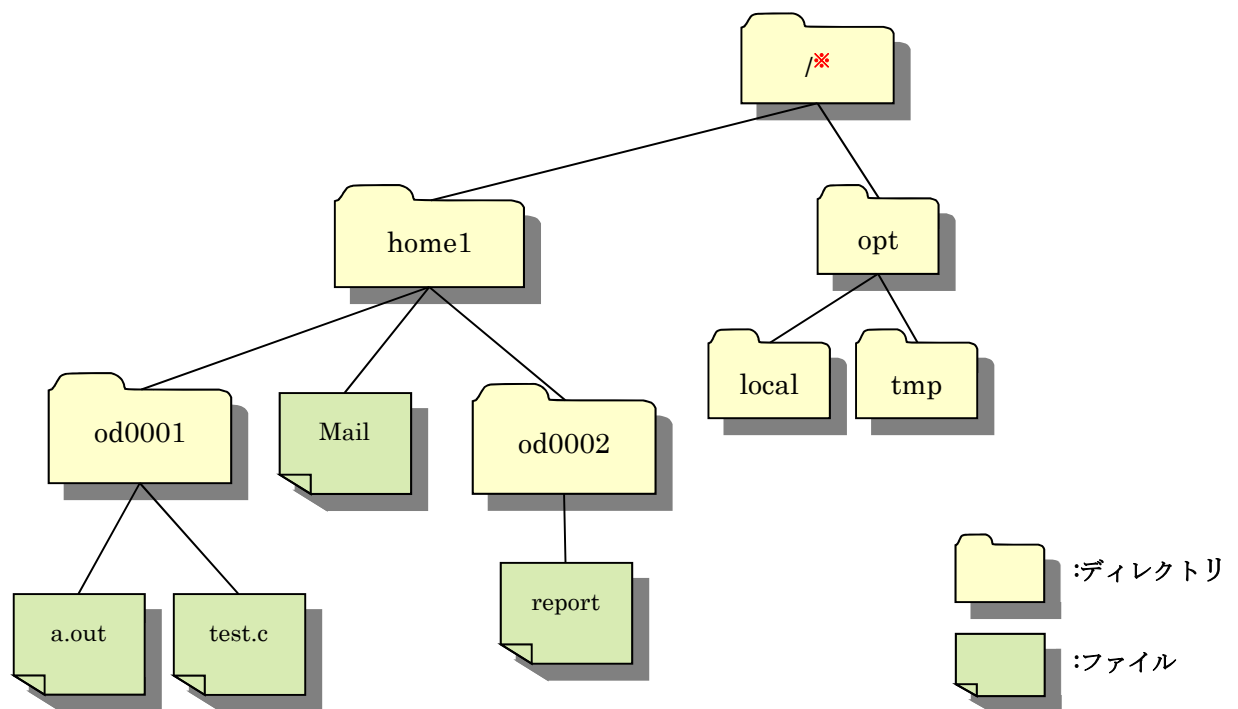


図 2-1 UNIX/Linux のディレクトリとファイル

※「/」であらわされるディレクトリは、ルートディレクトリ (root directory) と呼ばれます。
ルートディレクトリはどのような UNIX/Linux システムでも最上位のディレクトリであり、一つしか存在しません。

2.2. ファイルについて

UNIX/Linux でファイルに名前をつけるときは、以下の点に注意しましょう。

- 1) ファイル名は英数半角文字 255 字以内でつけます。基本的に全角文字は使えません²。
- 2) 英字の大文字(A～Z)と小文字(a～z)は区別されます。(a と A は別の文字と認識される)
[例] aaa.txt と AAA.txt は異なるファイルです。
- 3) 数字(0～9)がファイル名の先頭にあっても問題ありません。
- 4) ファイル名の先頭が「.(ドット)」のファイルは、**不可視ファイル**です³。

※なお、UNIX/Linux にはいくつものバージョンがあり、バージョンによってファイル名の制限が異なります。

2.3. ディレクトリについて

UNIX/Linux では、ファイルをディレクトリで効率的に管理します。MS-Windows や MacOS で「フォルダ」と呼ばれているものと同じです。このディレクトリの中に関連性のあるファイルを集めることにより、ユーザがファイルの管理を効率的に行うことを実現します。

ディレクトリは、以下のような特徴があります。

- 1) 必ず名前がある。
- 2) 好きなファイルやディレクトリを格納できる。ただし、1つのディレクトリの中に同じ名前のファイルやディレクトリを 2つ以上置くことはできない。

2.3.1. ホームディレクトリとカレントディレクトリ

UNIX/Linux 環境にログインした直後は、ある決まったディレクトリを参照している状態になります。この特別なディレクトリをホームディレクトリ(home directory)と呼び、ユーザ毎に専用のディスク領域が割り当てられています。

自分のホームディレクトリの中であれば、ユーザはファイルやディレクトリを自由に作成することができます。生田の情報処理教室の環境では、ホームディレクトリの名前はユーザ名(ログイン名)と同じになっています。

また、現在参照しているディレクトリのことをカレントディレクトリ(current directory)といいます。UNIX/Linux にログインした直後は、ホームディレクトリがカレントディレクトリとなります。

² 「端末」での入力が可能だが、ファイル名が表示されなかったり、文字化けが生じたりすることがある。

³ ファイルやディレクトリの詳細を表示させる方法については、『UNIX 利用の手引き』p.100 の ls コマンドを参照。

生田の情報処理教室の利用環境は以下の通りです

➤ ホームディレクトリの容量：1GB

文書や計算結果、ホームページ、受信したメールなどの各種ファイルがこのディレクトリに保存されます。1GB を越えてしまうと、それ以上ファイルやディレクトリを作成することはできません。ホームディレクトリの容量が 1GB を超えないように普段から気をつけてください。

➤ 「.windows2000」ディレクトリ

ホームディレクトリに「.windows2000」というディレクトリがあります。これは、Windows 環境の「マイドキュメント」で、UNIX/Linux からも参照や中のファイルの編集ができるようになっています。

➤ 「.snapshot」ディレクトリ

ホームディレクトリに「.snapshot」というディレクトリがあります。「.snapshot」には、前日から 14 日前までの日ごとのバックアップが自動保存されています。もしファイルを間違えて削除してしまった場合も、「.snapshot」ディレクトリのファイルからコピーすることで復元ができます。

2.3.2. ディレクトリの移動

作業するディレクトリの移動は `cd` コマンドで行います。コマンド操作については、p.28「4.コマンド操作とマウス操作の比較」で紹介します。

2.4. 絶対パスと相対パス

コマンドの実行など、各種の操作を行う場合に、目的のファイルやディレクトリの位置を指定しなければならないことがよくあります。UNIX/Linux では、その指定の方法が 2 つあります。ひとつが絶対パスによる指定、もうひとつが相対パスによる指定です。

2.4.1. 絶対パスとは

UNIX/Linux 環境に必ず存在し、しかもひとつしか無いルートディレクトリから辿った時の経路を記述する方法です。例えば、図 2-1 の場合、一番下にあるファイル `a.out` を絶対パスで指定すると、

```
/home1/od0001/a.out
```

となります。ディレクトリ名の間は「/」で区切ります。

2.4.2. 相対パスとは

カレントディレクトリを起点にして辿った時の経路を記述する方法です。例えば、図 2-

1 の **home1** というディレクトリで作業中の場合、先程と同じ **a.out** ファイルを相対パスで指定すると、

`./od01/a.out`

となります。

最初の「.(ドット)」は、カレントディレクトリを表す特別な記号です。また、カレントディレクトリのひとつ上のディレクトリは、「..(ドットドット)」という特別な記号で表すことができます。

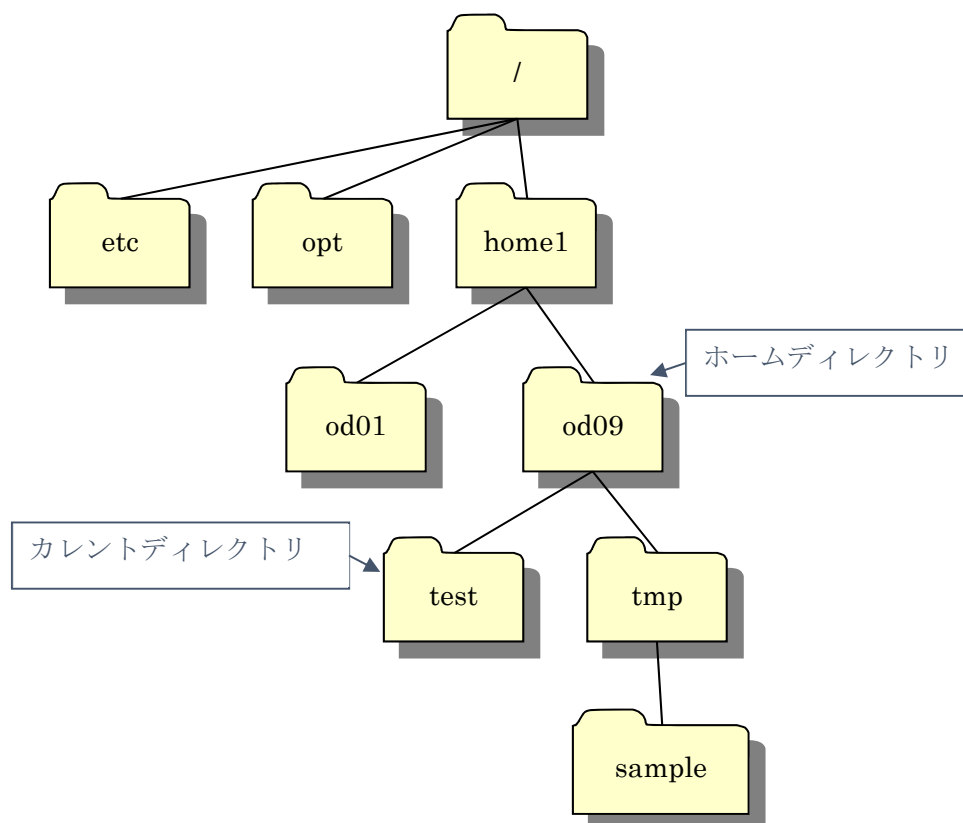


図 2-2 ホームディレクトリとカレントディレクトリ

図 2-2 を例にとると、「`.`」が表すディレクトリは **test** となり、「`..`」が表すディレクトリは **od09** となります。

絶対パスと相対パスのどちらを使うべきか迷うかもしれませんが、通常のファイル操作ではどちらでも構いません。使いやすい方を使ってください。

2.5. ファイル操作命令

ファイルに対して行える基本的な操作としては、以下のようなものがあります。

- ファイルの内容を表示する

- ファイルを連結する
- ファイルをコピーする
- ファイル名を変更する
- ファイルを移動する
- ファイルを削除する

これらの操作については、p.37 の「付録」を参照してください。

2.6. ディレクトリ操作命令

ディレクトリに対して行える基本的な操作としては、以下のようなものがあります。

- カレントディレクトリを表示する
- カレントディレクトリを変更する
- ディレクトリの内容一覧を表示する
- ディレクトリを作成する
- ディレクトリ名を変更する
- ディレクトリを移動する
- ディレクトリを削除する

これらの操作については、p.37 の「付録」を参照してください。

3. CentOS の使い方

ここでは CentOS の基本的な使い方として以下を紹介します。

- CentOS の起動
- 各画面の説明
- CentOS で操作できるソフトウェア
 - ・ インターネットブラウザ
 - ・ テキストエディター (gedit の使い方)
 - ・ その他
- CentOS の終了

3.1. CentOS の起動

PC の電源をつけると、OS の選択画面が表示されます。CentOS を選択してください。
仮想デスクトップを用いる場合は、VMware を使用します。詳細は、「生田仮想デスクトップ PC の利用方法について」を参照してください。

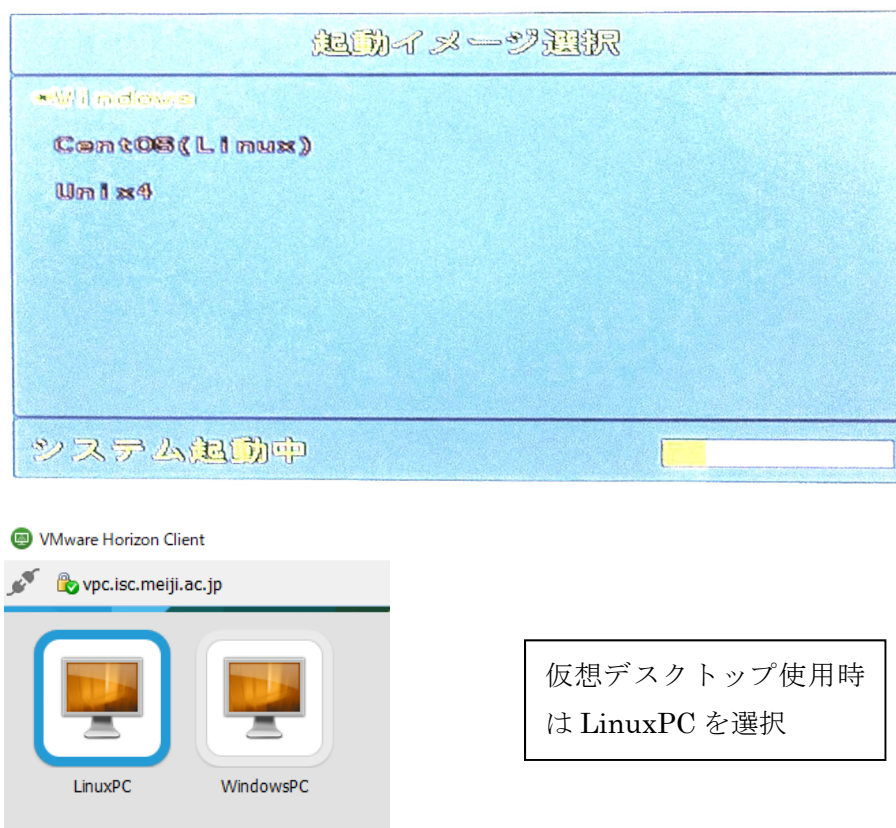


図 3-1 起動イメージ選択画面

OS の選択後しばらくするとログイン画面が表示されます。
CentOS にログインするためにユーザ名とパスワードを入力します。ユーザ名とパスワードは Windows にログインするための基盤アカウントのものと同一です。
はじめに、ユーザ名を入力します。



図 3-2 CentOS ユーザ名入力

次にパスワードを入力します。

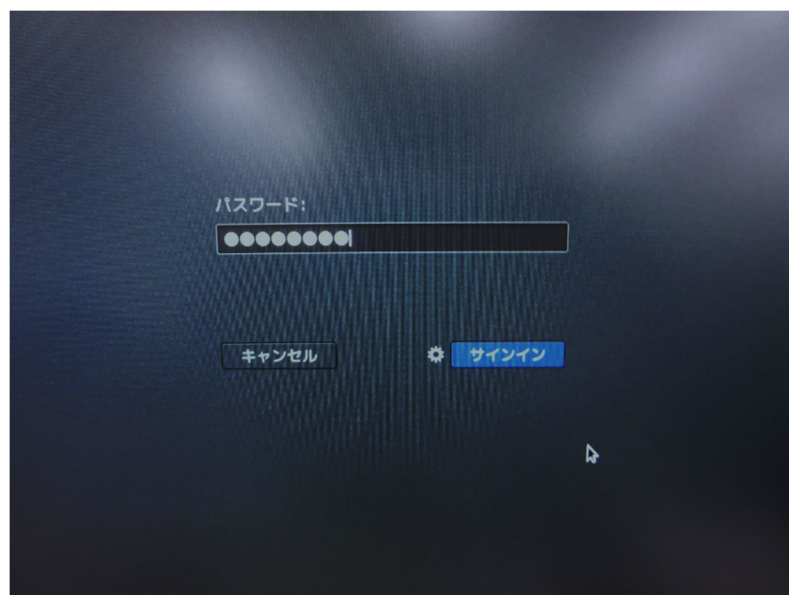


図 3-3 CentOS パスワード入力

ログイン認証が完了すると、デスクトップ画面が表示されます⁴。

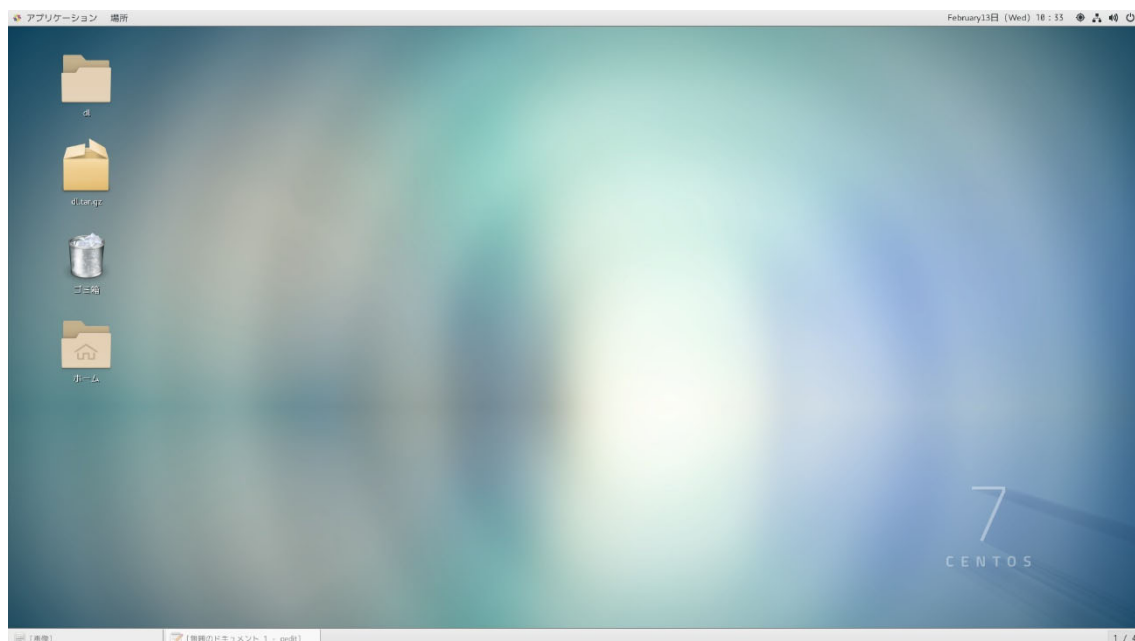


図 3-4 CentOS デスクトップ画面

3.2. 各画面の説明

各メニューやディレクトリの説明を以下で示します。

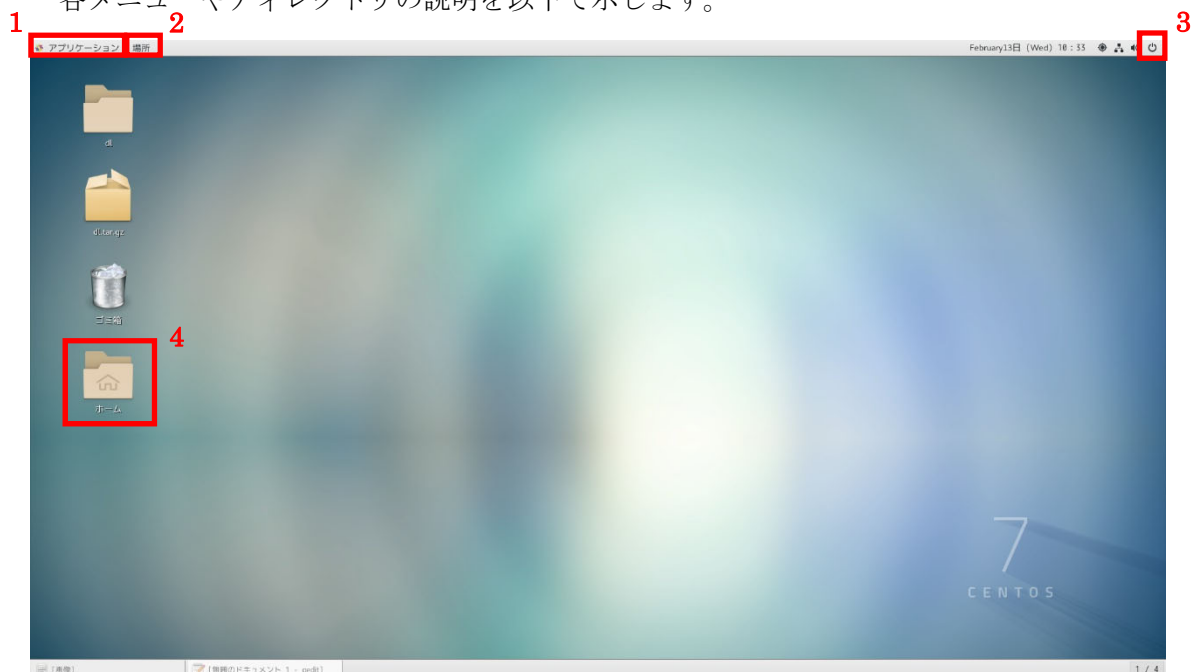


図 3-5 CentOS 各メニュー

⁴ 現在、表示されているデスクトップ環境は GNOME という。デスクトップ環境の種類はいくつかあり、ユーザが自由に採用することができる。

表 2 CentOS のメニュー一覧

項番	タイトル	内容
1	アプリケーション	CentOS にインストールされているアプリケーションが使える
2	場所	ユーザの各種ディレクトリを開く
3	システム	システム全般の設定と CentOS のログアウトやシャットダウンが行える
4	○のホーム(○はユーザ名)	ユーザのホームディレクトリを見ることができる ※現在、画面に表示されているディレクトリはデスクトップ

3.2.1. アプリケーションメニューの説明

アプリケーションメニューから様々なアプリケーションを起動することができます。

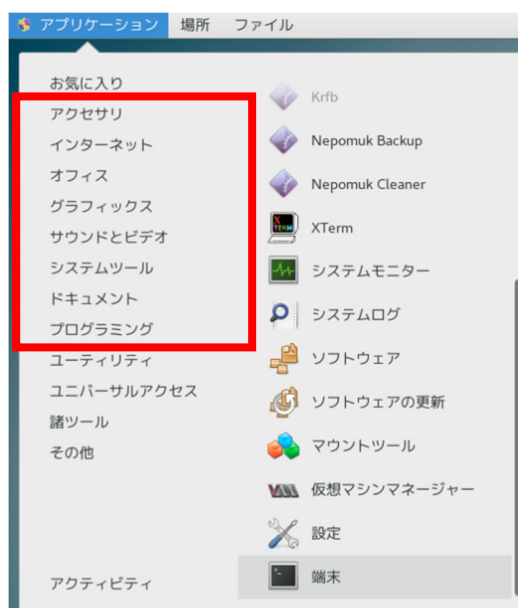


図 3-6 CentOS アプリケーション

表 3 CentOS のアプリケーション一覧

項番	タイトル	内容
1	アクセサリ	テキストエディターや電卓など基本的なソフトウェアが入っている
2	インターネット	ウェブブラウザやメールクライアントなど、ネットワークを経由するソフトウェアが入っている
3	オフィス	オフィススイートと呼ばれる文章作成ソフトや表計算ソフトが入っている
4	グラフィックス	ペイントソフトや画像ビューワソフトが入っている
5	サウンドとビデオ	動画や音声ファイルを再生するソフトウェアが入っている
6	システムツール	コマンドラインで操作する「端末」やシステム関連のソフトウェアが入っている
7	プログラミング	ソフトウェアの設計や作成に関連するソフトウェアが入っている

3.2.2. 場所メニューの説明

任意のディレクトリを開くことができます。また、ディレクトリやファイルに関連することもここで調べることができます。

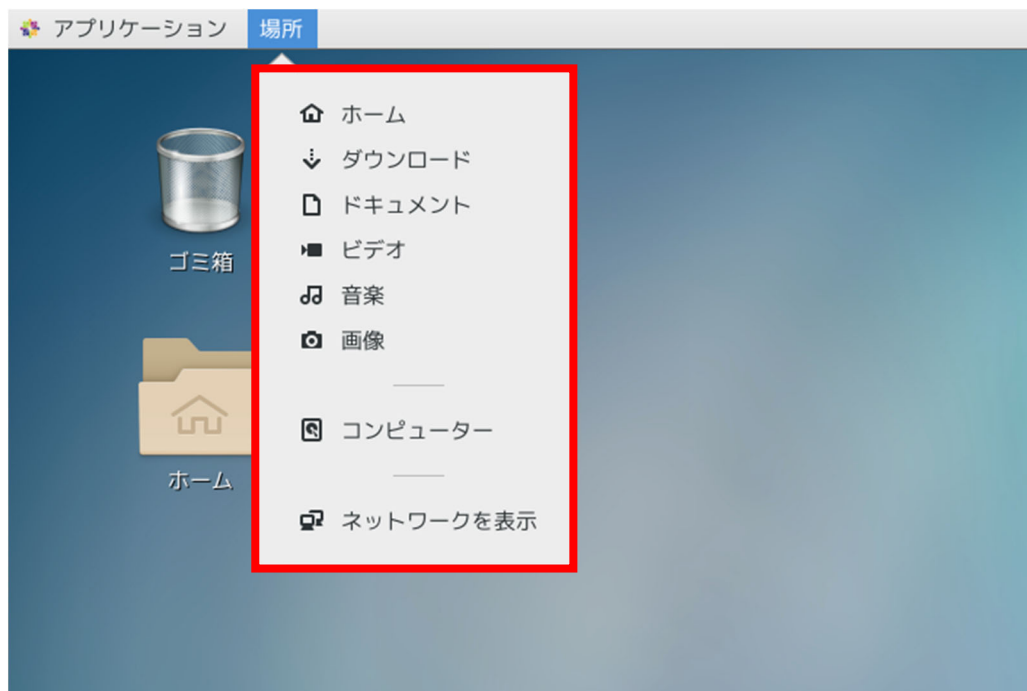


図 3-7 CentOS 場所一覧

3.2.3. システムメニューの説明

システム全般の設定、ソフトウェアのアップデート、ログアウト、シャットダウンができます。ただし、生田の環境でソフトウェアのアップデートなどシステムの変更に関わることは管理者権限を必要とするので、ここでは **CentOS** を終了する手順を覚える程度で十分です⁵。



図 3-8 CentOS システム一覧

⁵ シャットダウンの方法は p.27 「3.4 CentOS の終了」を参照すること。

3.3. CentOS で操作できるソフトウェア

3.3.1. インターネットブラウザを使う

「アプリケーション」メニューから「インターネット」、「Firefox Web ブラウザ」を選択します。



図 3-9 Firefox ウェブブラウザの起動

Firefox ウェブブラウザが起動します。

なお、Mozilla Firefox が起動しない場合は、付録の p.37 を参照してください。



図 3-10 Firefox の画面

Firefox でインターネットの検索をしてみましょう。例えば、気象庁のサイトを調べてみます。赤枠の検索ボックスに「気象庁」と入力します。次に[Enter]キーを押します。



図 3-11 Firefox でのインターネット検索方法

検索結果が表示されます。このように自分が知りたい情報を調べることができます。



図 3-12 Firefox でのインターネット検索結果

3.3.2. テキストエディターを使う

「アプリケーション」メニューから「アクセサリ」、「テキストエディター」を選択します。



図 3-13 gedit テキストエディターの起動

gedit が起動します。

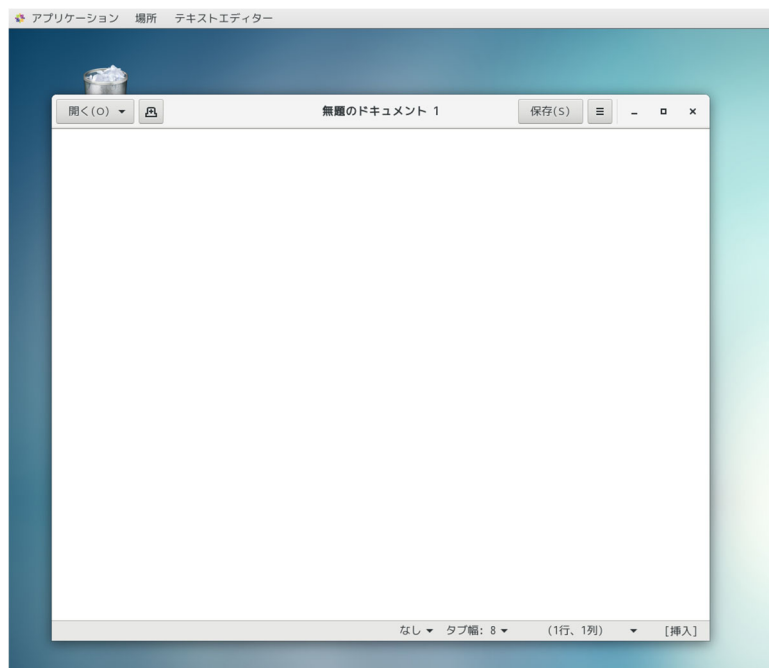


図 3-14 gedit の画面

gedit で文字を入力してみましょう。赤枠のテキストボックスに「これは編集中のテキストです。」と入力します。

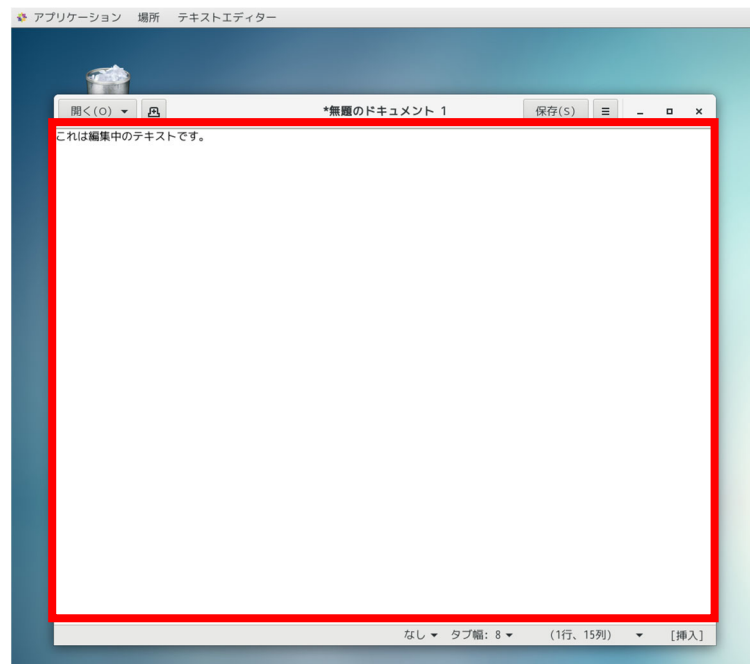


図 3-15 gedit の編集方法

テキストの内容を保存するには、「保存」を選択します。

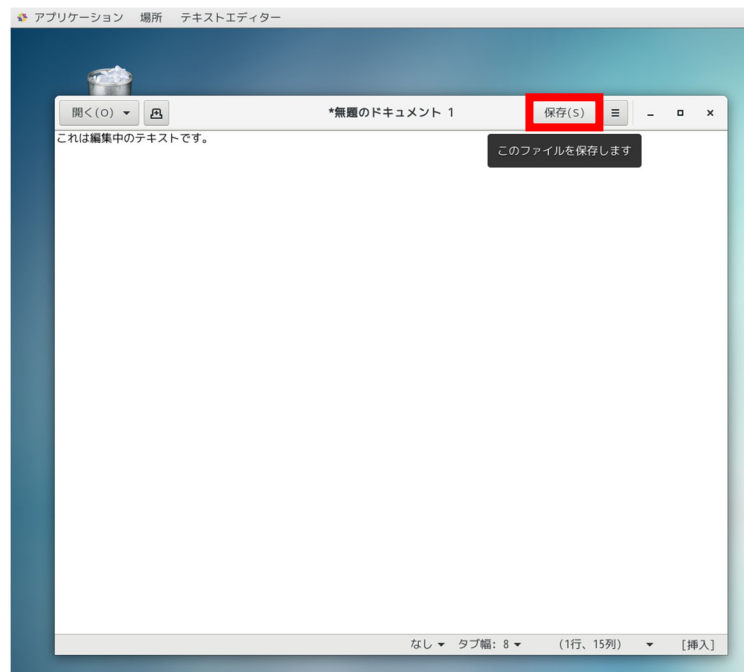


図 3-16 gedit でのファイルの保存方法

任意の名前を付けた後、保存場所を選択して、「保存」ボタンを押します。

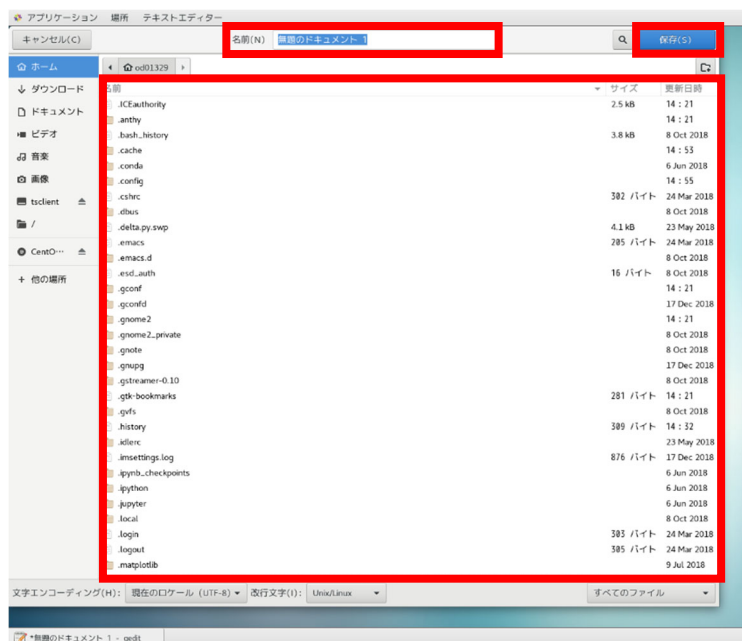


図 3-17 gedit でのファイルの保存場所の指定

テキストファイルが保存されました。

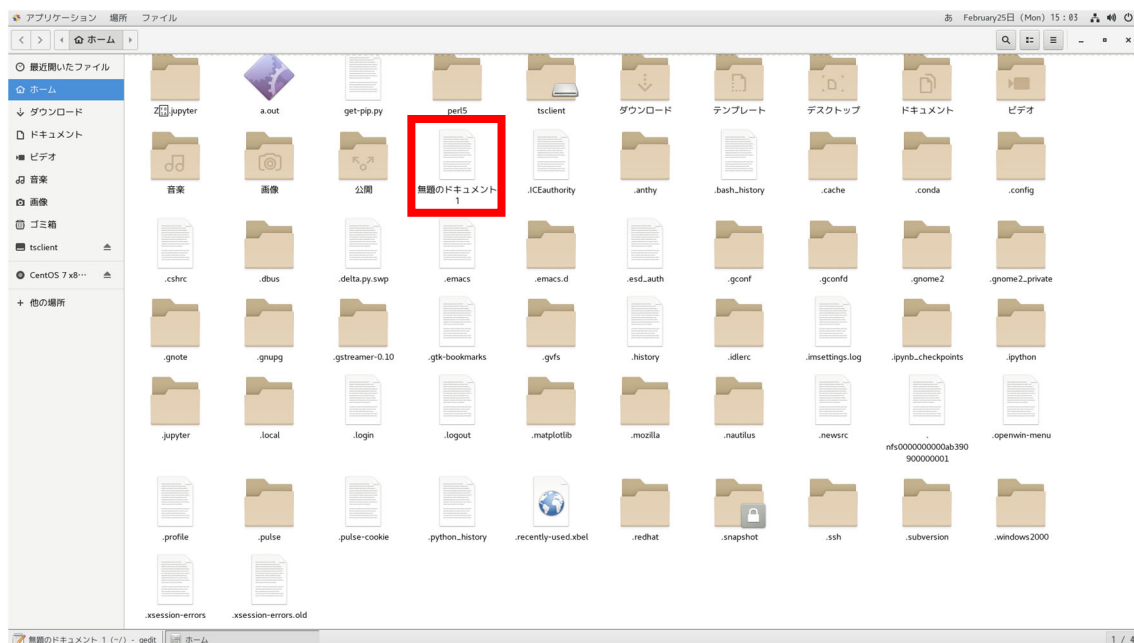


図 3-18 CentOS でのファイルの保存場所の確認方法

3.3.3. ソフトウェアの紹介

主なソフトウェアについて簡単に紹介します。CentOS で使用できるソフトウェアには、先述したもの以外にも以下のようなオフィス業務に必要なソフトウェアがあります。

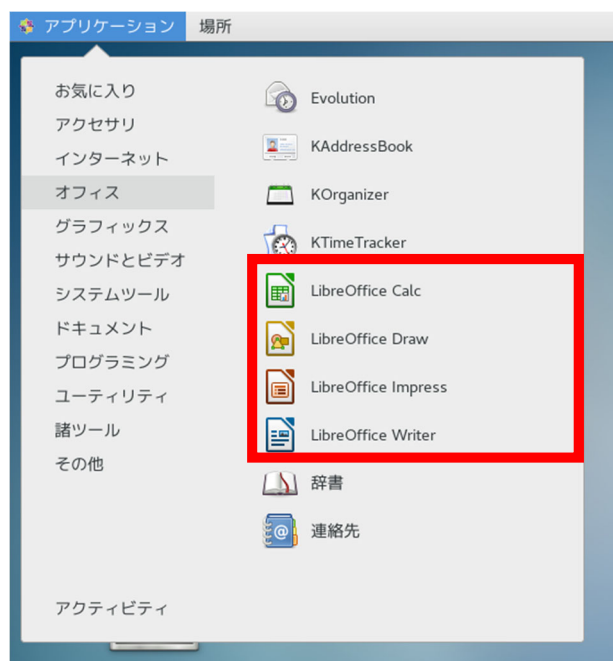


図 3-19 オフィス業務に必要なソフトウェア

(1) LibreOffice の文章作成ソフト Writer

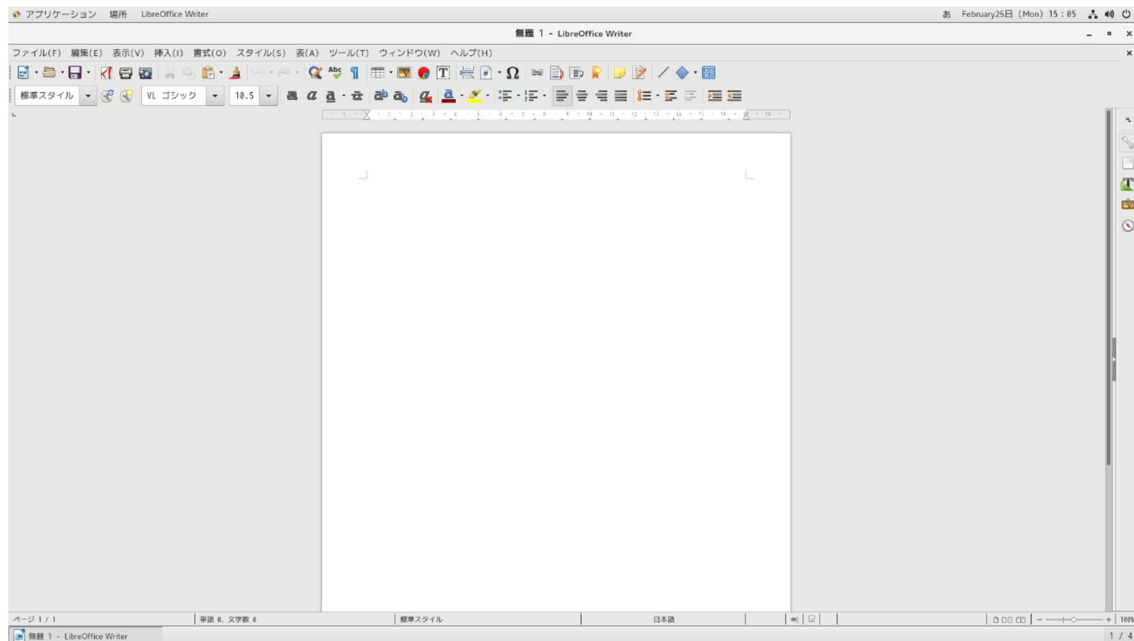


図 3-20 LibreOffice Writer

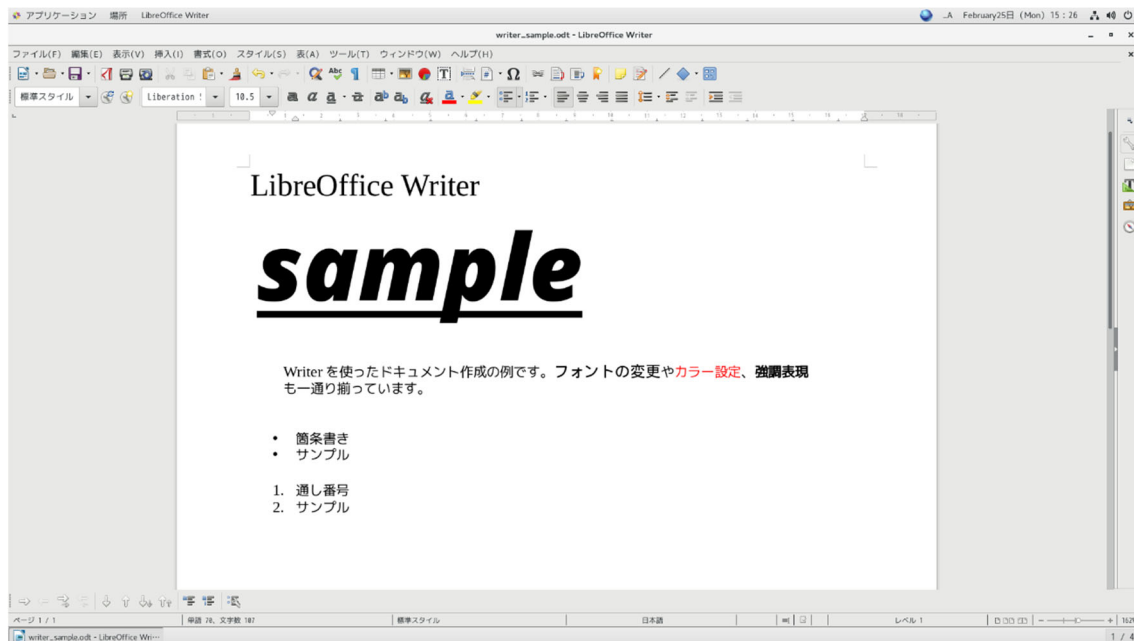


図 3-21 Writer のサンプル

(2) LibreOffice の表計算ソフト Calc

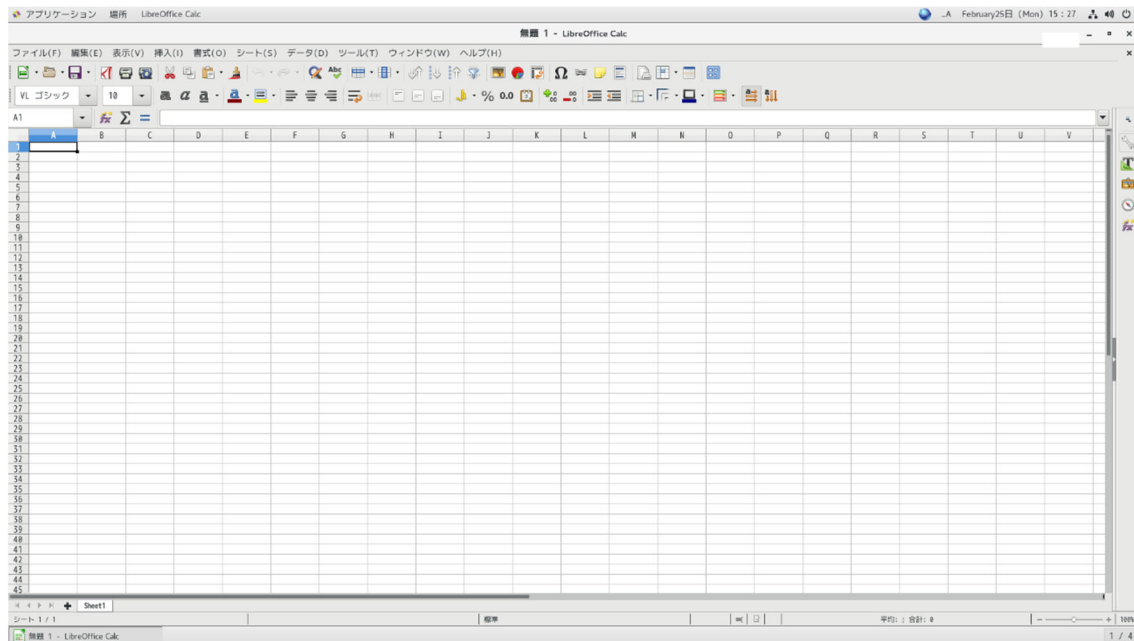


図 3-22 LibreOffice Calc

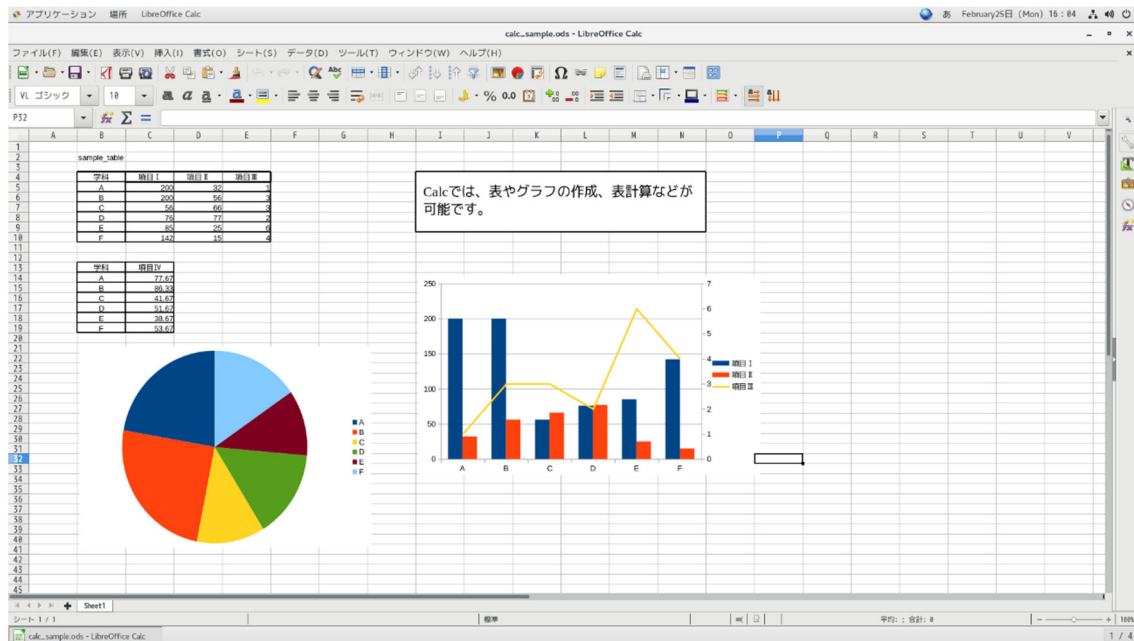


図 3-23 Calc のサンプル

(3) LibreOffice のプレゼンテーションソフト Impress

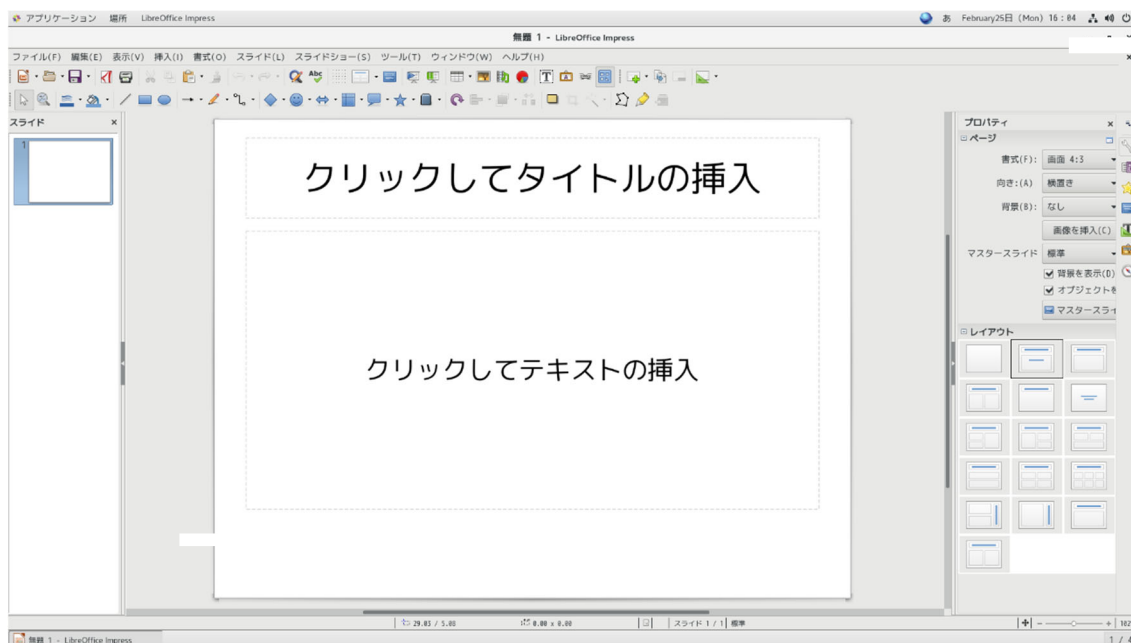


図 3-24 LibreOffice Impress

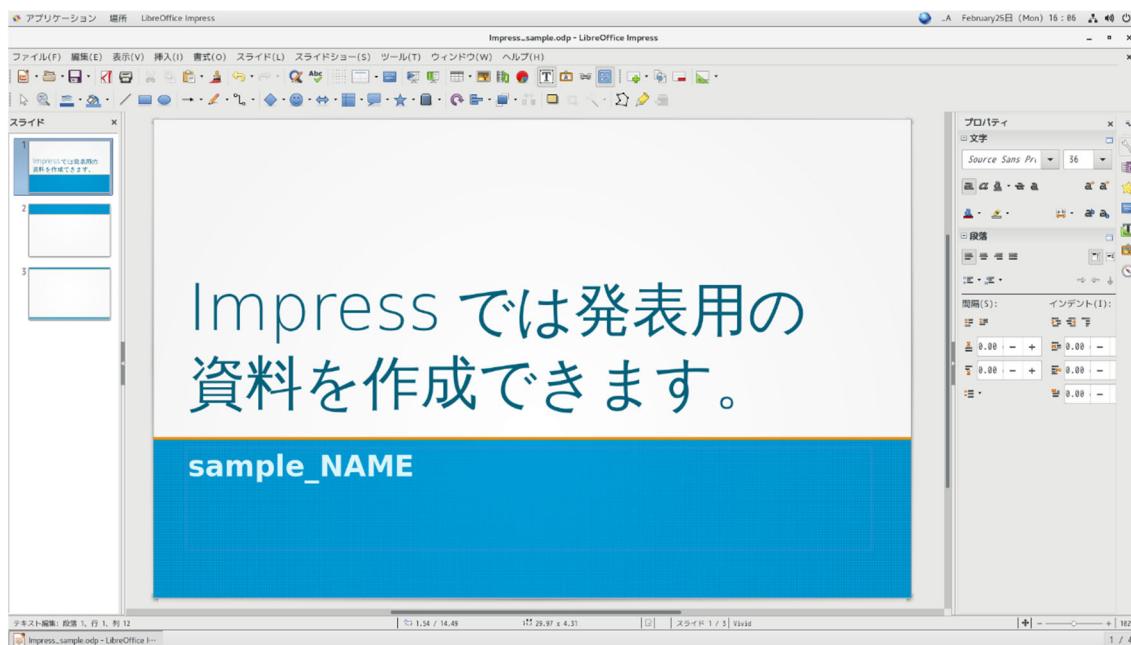


図 3-25 Impress のサンプル

(4) LibreOffice の図形描画ソフト Draw

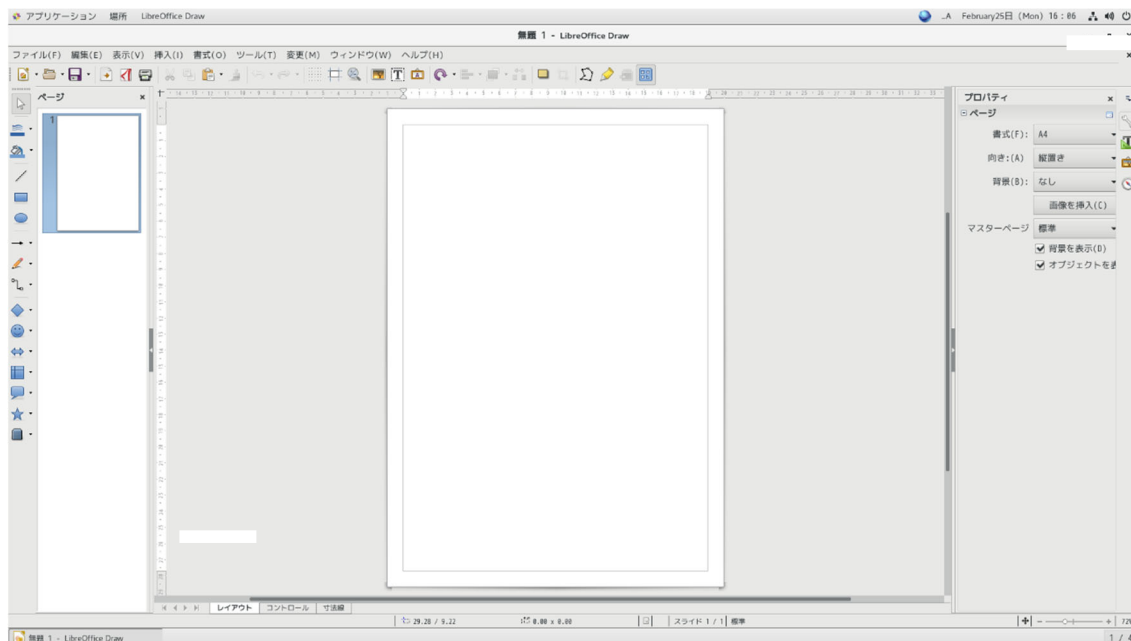


図 3-26 LibreOffice Draw

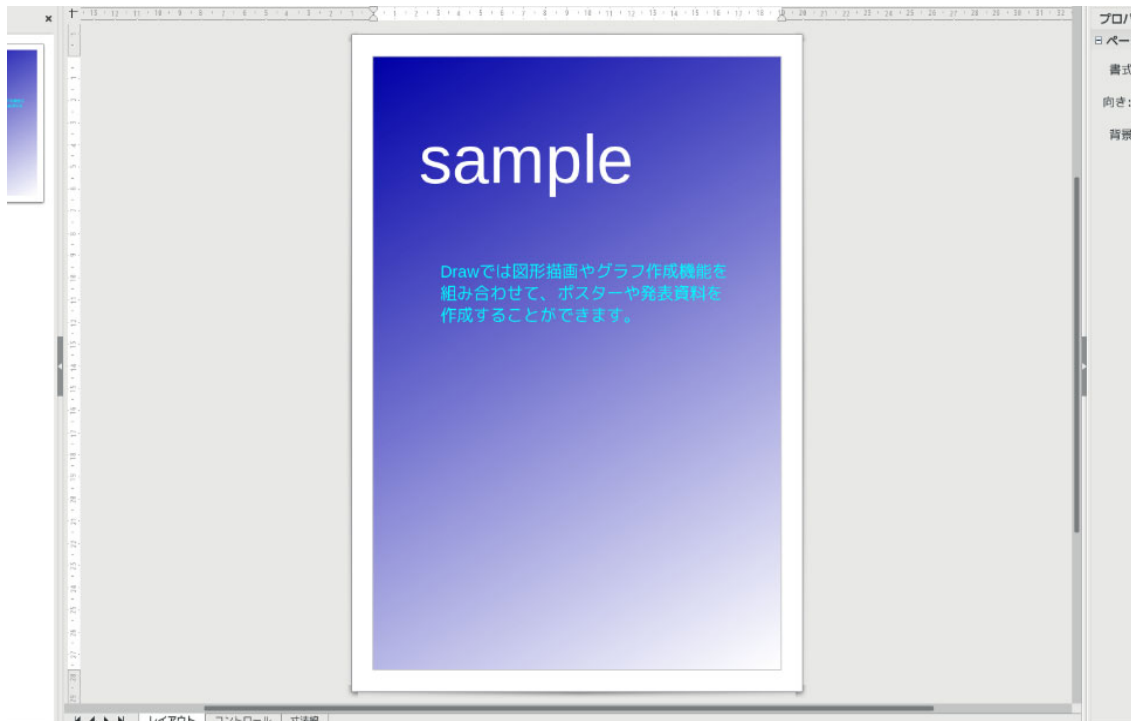


図 3-27 Draw のサンプル

3.4. CentOS の終了

CentOS の終了手順は以下の通りです。

右上のシステムメニューから「電源マーク」のアイコンを選択します。

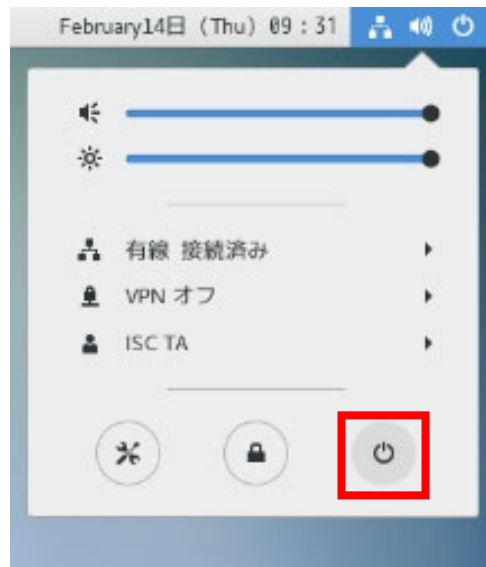


図 3-28 CentOS の終了

電源オフというメッセージダイアログが出るので、「電源オフ」ボタンを押します。

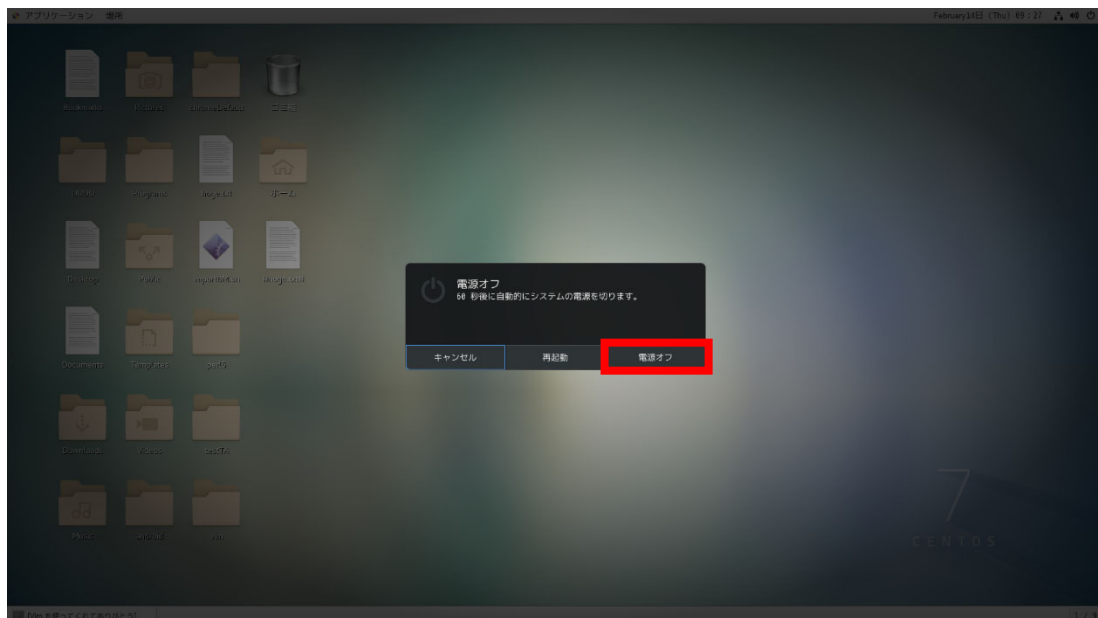


図 3-29 システムのシャットダウン

しばらく待つとシステムが終了します。

4. コマンド操作とマウス操作の比較

UNIX/Linux ではコマンド操作とマウス操作のどちらかを使い操作することができますが、基本的に内容は同じです⁶。以下、その例を示します。

ディレクトリの選択について

コマンドを使ったディレクトリの移動は `cd` コマンドで行います。

なお、`cd` コマンドについては下記表を参照してください。「~」はホームディレクトリを表します。

表 4 `cd` コマンドの入力例

入力 ⁷	動作
<code>cd</code>	ホームディレクトリへ移動
<code>cd_..</code>	一つ上のディレクトリへ移動
<code>cd_dir1</code>	カレントディレクトリの下にある <code>dir1</code> ディレクトリへ移動
<code>cd_dir1/dir2/dir3</code>	カレントディレクトリの 3 階層下にある <code>dir3</code> へ移動
<code>cd_/_</code>	<code>/</code> (ルート)ディレクトリへ移動
<code>cd_~/dir1</code>	ホームディレクトリの下にある <code>dir1</code> ディレクトリへ移動
<code>cd_~/dir1/dir2/dir3</code>	ホームディレクトリの 3 階層下にある <code>dir3</code> へ移動

マウスを使ったディレクトリの移動はディレクトリに対してダブルクリックします。

⁶ 昔のコンピュータは全てコマンドで操作していた。現在はグラフィカルに操作できるため、どちらを選択しても良い。

⁷ 本テキストでは半角スペースを `_` で表す。

4.1. ホームディレクトリに移動

コマンドを使う場合は、「端末」を選択します。

「アプリケーション」メニューから「システムツール」、「端末」を選択します。

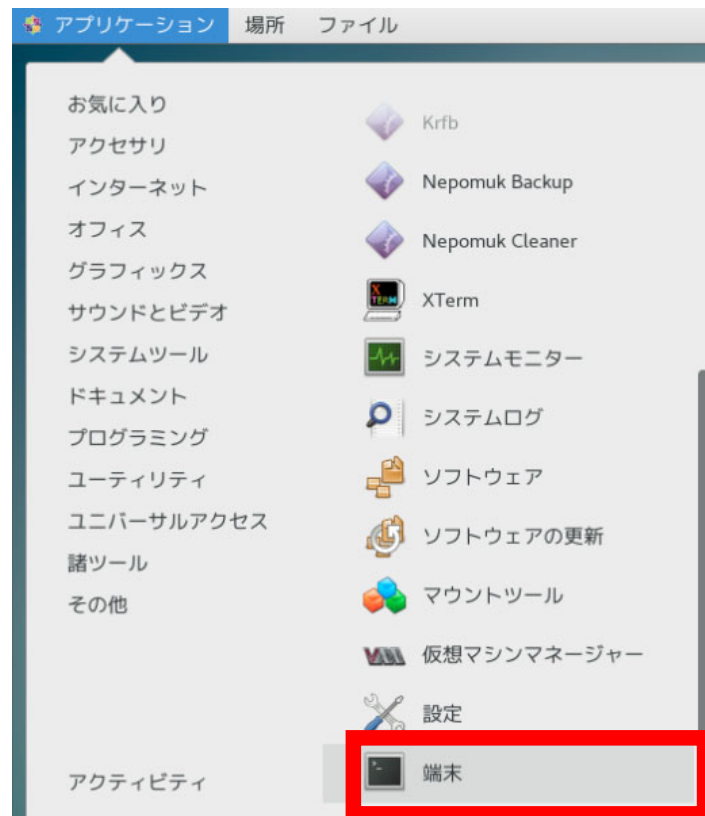


図 4-1 「端末」の起動

ディレクトリの移動には `cd` コマンドを使用します。⁸

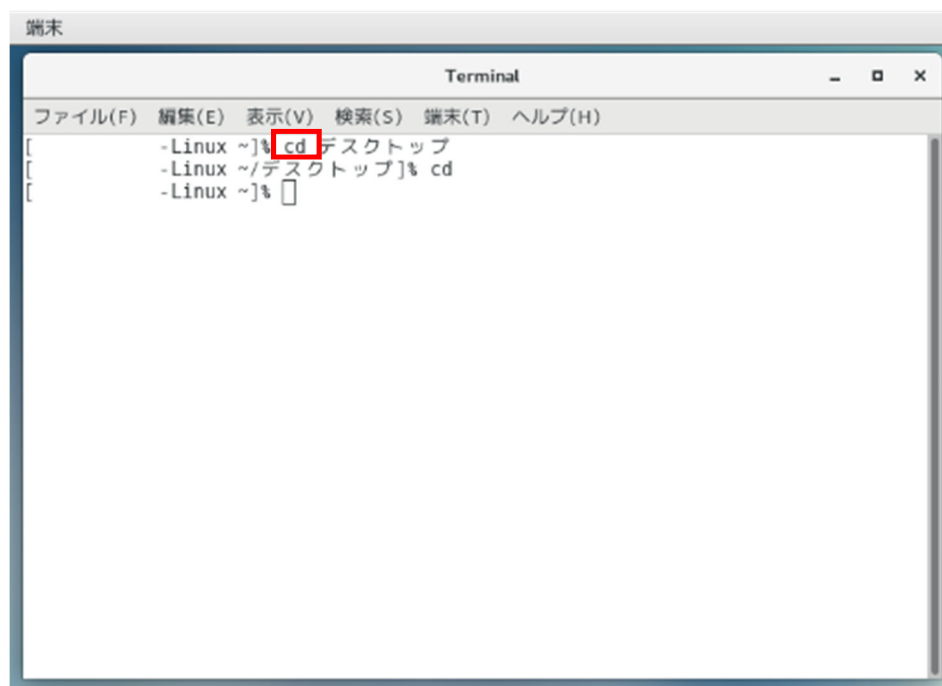


図 4-2 CentOS の「端末」画面(`cd` コマンドの使用)

マウスを使う場合は、ホームディレクトリ(赤枠のディレクトリ)をダブルクリックします。

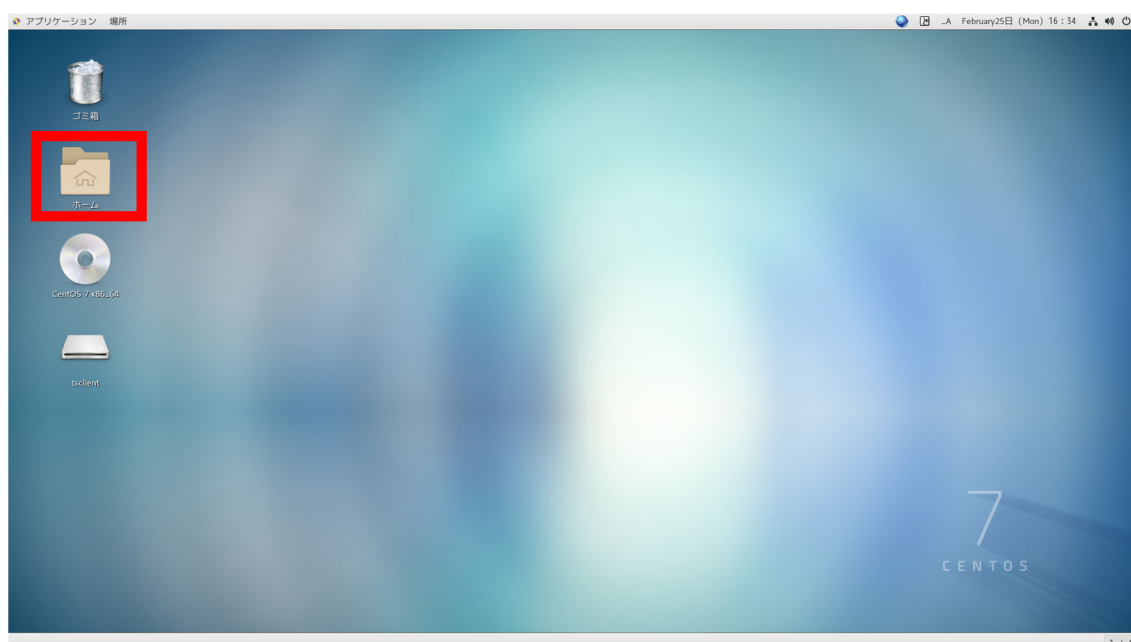


図 4-3 マウスを使ったホームディレクトリの参照方法

⁸ `ls` コマンドの実行後に表示される「デスクトップ」をコピーし、`cd` コマンドの後ろに貼り付けることで「`cd デスクトップ`」を入力できる。

4.2. ディレクトリにあるファイルやディレクトリを表示

コマンドを使う場合は、ls コマンドを入力します。

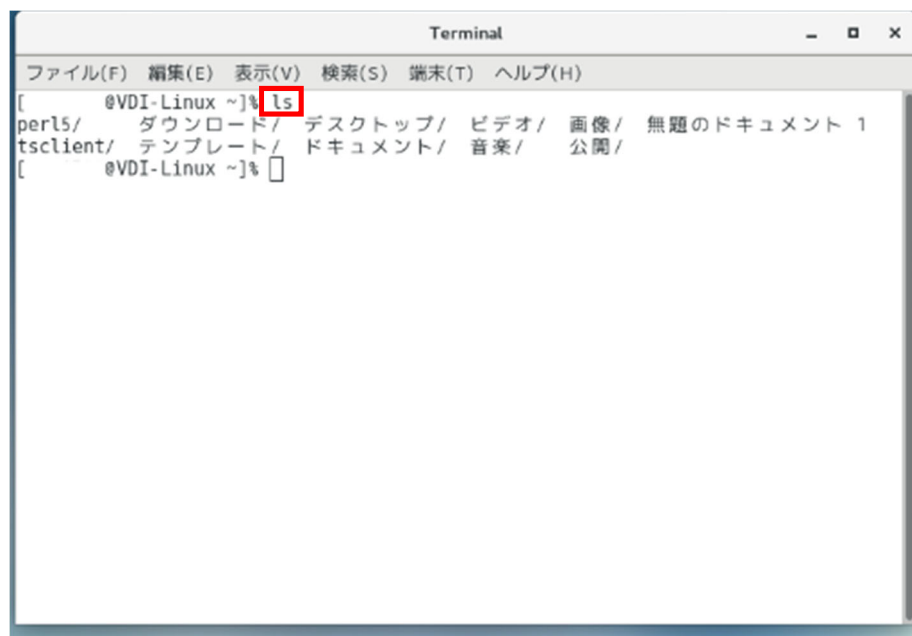


図 4-4 ls コマンドの使用例

マウスを使う場合は、ディレクトリをダブルクリックすればファイルやディレクトリが表示されます。

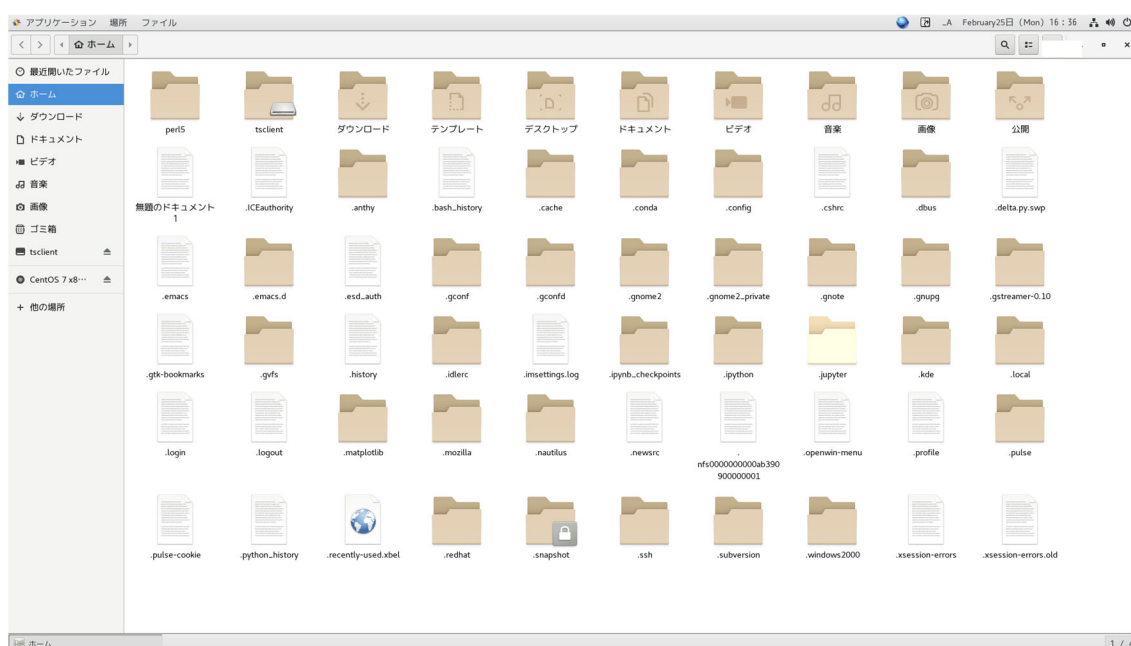


図 4-5 マウスを使ったディレクトリの表示

5. シェルの機能

本章では、シェルの持つ機能について説明します。シェルとはユーザからの命令を受けつけて OS に伝える役割を持ったプログラムです。主な機能は以下の通りです。

表 5 シェルの機能一覧

シェルの主な機能	説明
コマンドラインインタプリタ	コマンドの解釈や実行
リダイレクト	入出力の方向を変更する
パイプライン(パイプ)	コマンドの出力を直接別のコマンドに結び付ける機能
シェルスクリプト機能	コマンドを書き込んだファイル(シェルスクリプト)を読み込み、実行する機能

5.1. コマンドシェル

コマンドシェル(Command Shell)はユーザが入力したコマンド行を解釈し、その実行を UNIX/Linux に依頼するという役割を持っています。そのため、コマンドインタプリタ(Command Interpreter)とも呼ばれます。

情報処理教室等の UNIX/Linux 環境で利用できるコマンドシェルには、Bourne シェル、C シェル、Bourne Again シェル⁹などがありますが¹⁰、生田システムが標準として採用しているのは C シェルです。UNIX/Linux を快適に操作するためには、この C シェルの機能をうまく活用する必要があります。

なお、bash は C シェルと比較すると機能が拡張されているので、bash の使用をお勧めします。UNIX/Linux②基礎編・③中級編でのシェルの使用は bash を想定しています。

⁹ bash(バッシュ)とも呼ばれる。

¹⁰ p.33 表 6 を参照。

表 6 シェルの種類

主なシェルの種類	略省	説明	分類 ¹¹
Bourne Shell	sh	UNIX/Linux の標準的なシェル	sh 系
C Shell	csch	文法が C 言語に似ていると言われているシェル	csch 系
Korn Shell	ksh	sh を基本に機能を追加したシェル。Korn Shell の後に Bourne-Again shell が作成されている	sh 系
TENEX C shell	tcsh	csch を基本に、機能を追加したシェル	csch 系
Bourne-Again shell	bash	sh を基本に、多数の機能を追加したシェル。UNIX/Linux で広く用いられるオープンソースであるシェルの一つ	sh 系
Z Shell	zsh	多機能シェル。sh や csch の機能を持っている	sh 系、csch 系

※ シェルの変更について

「端末」(ターミナル)から「bash」、「tcsh」などを入力すれば、その「端末」内のシェルは変更されます。変更されたシェルを戻すには、同じ「端末」で「exit」と入力します。

5.2. 入出力の切り替え

一般的に、入力はキーボード、出力はディスプレイとなっていますが、UNIX/Linux では入出力を切り替えることができます。その機能を実現するのが、シェルにあるリダイレクト機能とパイプ機能です。

5.2.1. リダイレクト機能¹²

リダイレクト機能には、(1)上書きファイル出力「>」、(2)追加ファイル出力「>>」、(3)ファイル入力「<」、(4)エラー出力「>&」の 4 つのパターンがあります。

¹¹ シェルには様々なものがあるが、歴史(派生関係)から sh 系と csch 系に大別されることがある。同システムのシェルは機能や特徴が似ているとされている。

¹² シェルの種類によって使用方法が異なるため、注意が必要。

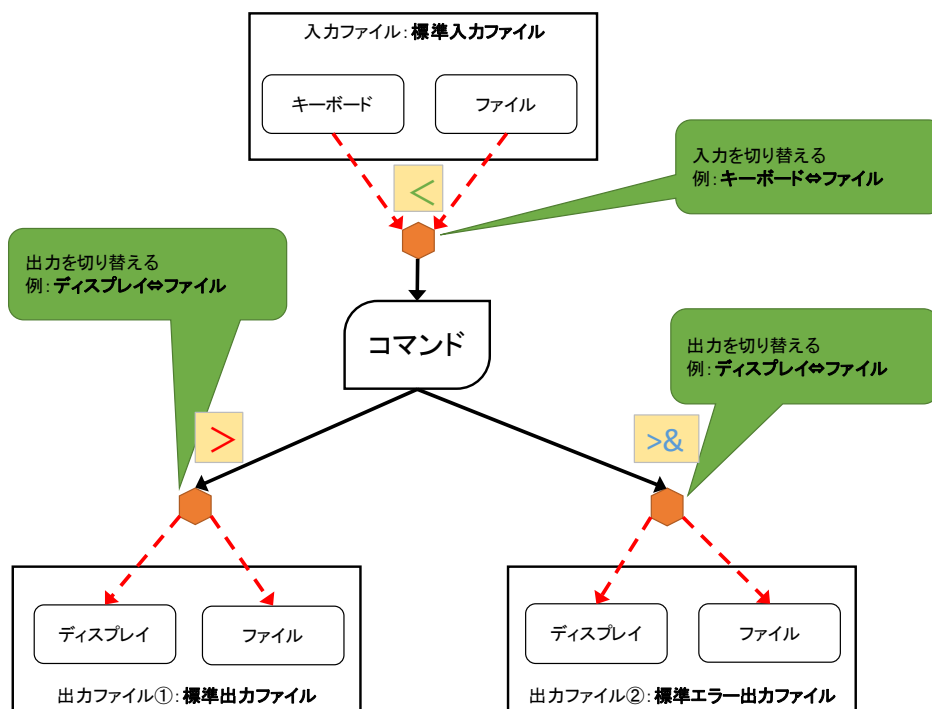


図 5-1 リダイレクト機能のイメージ

(1) 上書きファイル出力

プログラムやコマンドの実行結果を画面ではなくファイルに書きだすには、次のように「>」を使います。

\$ コマンド名 > 出力先ファイル名

例えば、ディレクトリの中に含まれるファイルの一覧を表示するコマンド `ls` の出力を `out` というファイルに書きだす処理は次のようになります。

```
$ ls > out
```

(2) 追加ファイル出力

「>」で指定された出力先のファイルが存在していなかった時には、新しくファイルが作成されて、そこに書き込まれます。しかし、既に存在しているファイルを指定した場合には、上書きされてしまい、元の内容は消えてしまいます。元の内容に上書きせず、追加して書きだしたい場合には、次のように「>>」を使います。

\$ コマンド名 >> 出力先ファイル名

(3) ファイル入力

リダイレクト機能を使ってファイルから入力を行うには、次のように < を使います。

\$ コマンド名 _<_入力元ファイル名

(4) エラー出力

リダイレクト機能を使って、エラー出力を行うには、次のように >& を使います。

\$ コマンド名 _>&_エラー出力先ファイル名

5.2.2. パイプ機能

パイプ機能はプログラム同士を結びつける機能です。つまり、あるプログラムの標準出力を、そのまま別のプログラムの標準入力に結びつけます。パイプ機能については、UNIX/Linux②基礎編で紹介します¹³。

¹³ UNIX/Linux②基礎編・③中級編 テキスト「2.2.4.パイプによりコマンドを組み合わせる」

6. UNIX/Linux②基礎編に向けて

本テキストで扱ったディレクトリ、シェル機能などの知識をベースにして、UNIX/Linux 初心者・中級編では、UNIX/Linux を使用する際に広く活用されているコマンドライン操作、テキストエディターの操作について学びます。さらにやや発展的なトピックとしてシェルスクリプトを扱います。本テキストで学んだことに加え、UNIX/Linux②基礎編・③中級編テキストで UNIX/Linux に関して発展的な事例を学び、UNIX/Linux④管理者編テキストでの UNIX/Linux の管理者としての基本的な内容の学習に進みます。

付録

Mozilla Firefox が起動しない場合

ホームディレクトリから Firefox の設定ファイル"parentlock"を削除すると解決する場合があります。

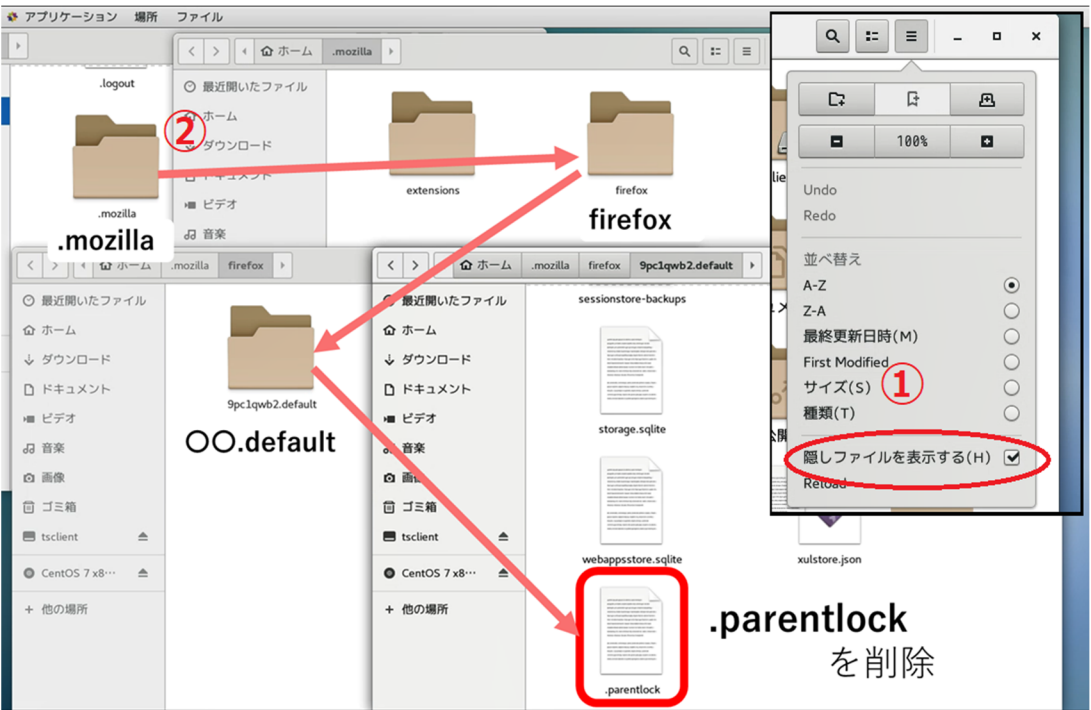


図 7-1 Firefox の設定ファイル

プロセス、ジョブ、バックグラウンド処理について

UNIX/Linux を使うにあたり頻出する用語について、下記表で説明をします。

表 7 ジョブとプロセス

用語	分類	内容
ジョブ		コンピュータに処理をさせる作業の単位
	フォアグラウンドジョブ	ユーザがある時点で操作している作業のこと
	バックグラウンドジョブ	ユーザがある時点では操作していないが裏で起動して控えている作業のこと
プロセス		プログラムの実行単位
	親プロセス	最初に実行されたプログラムのこと
	子プロセス	親プロセスが実行したプログラムのこと

ファイル、ディレクトリ操作について

UNIX/Linux を使うにあたり頻出するコマンドについて、下記表で説明をします。

表 8 ファイル操作に関するコマンド

コマンド	内容	『UNIX 利用の手引き』 対応ページ
cat	ファイルの内容を表示する ファイルを連結する	p.74
cp	ファイルをコピーする	p. 79
mv	ファイル名またはディレクトリ名を変更する ファイルまたはディレクトリを別の場所に移動する	p. 106
rm	ファイルを削除する	p. 111

表 9 ディレクトリ操作に関するコマンド

コマンド	内容	『UNIX 利用の手引き』 対応ページ
pwd	カレントディレクトリを表示する	p. 110
cd	カレントディレクトリを変更する	p. 75
ls	ディレクトリの内容を一覧表示する	p. 100
mkdir	ディレクトリを作成する	p. 103
mv	ファイル名またはディレクトリ名を変更する ファイルまたはディレクトリを別の場所に移動する	p. 106
rmdir	ディレクトリを削除する	p. 113

ファイルやディレクトリ操作の基本

- 1) ホームディレクトリに移動する

```
$ cd
```

- 2) test と書かれたテキストファイル test221 を作成する

```
$ gedit test221
```

gedit が起動するので、中身に test と書き、保存して終了する

- 3) ファイルの内容を見る

```
$ cat test221
```

- 4) ディレクトリを作成する

```
$ mkdir testdir
```

- 5) testdir ディレクトリが作成されたかどうか確認する

```
$ ls
```

- 6) ファイルを作成したディレクトリ内へ移動させる

```
$ mv test221 testdir/
```

- 7) ディレクトリを移動する

```
$ cd testdir
```

- 8) ファイルが移動しているか確認する

```
$ ls
```

- 9) ファイルを削除する

```
$ rm test221
```

- 10) ファイルが削除されたことを確認する

```
$ ls
```

- 11) ホームディレクトリに移動する

```
$ cd
```

- 12) 現在のディレクトリの場所を確認

```
$ pwd
```

- 13) 作成したディレクトリを削除する¹⁴

```
$ rm -r testdir
```

- 14) ディレクトリが削除されたことを確認する

```
$ ls
```

¹⁴ rm コマンドの後ろに入力した「-r」はオプションと呼び、オプションを入力することでコマンドに対して細かい動作を指定することができる。rm コマンドは「-r」オプションを指定することでファイルだけでなくディレクトリも削除することができる。(指定したディレクトリ内のファイルもすべて削除するので注意が必要)

参考文献

小林準(2011)『独習 Linux 第 2 版』翔泳社

中島能和、飛田伸一郎(2012)『CentOS 徹底入門第 3 版』翔泳社

濱野賢一郎監修、中島能和著(2012)『Linux 標準資格教科書 LPIC レベル 1 対応』技術評論社