

## マルチメディア | Web基礎「WordPress」第28回 デザインとCSS

授業特設サイト☆

<https://icc.firstelement.co.jp/>

配布資料のPDFダウンロード、授業のフォローアップなど。  
わからないことはコメント欄から質問も歓迎！

「Local」を立ち上げて、icc-booksの左側のグレーの三角をクリックしてサイトを起動し、Localの右端にあるADMINボタンをクリックしてWP管理画面にログインしよう。

### デザインとCSS

「デザイン」の定義は幅広く、実際に境界も曖昧。  
“機能”のデザイン、“世界観”のデザイン、“体験”のデザイン...。  
サイトが何を目指すのかによって優先度・重要度は変わるが、どのようなサイトであっても【伝えたいことをどう表現したら伝わるか】という軸は変わらない。

1. Webサイトで何がしたいのか、伝えたいのか
  - a. 商品売りしたい
    - i. どのような商品？
      1. 食品、機械、サービス、嗜好品...
  - b. 事業内容を多くの人に知ってほしい
  - c. 人材を募集したい
    - i. 新卒、中途...
  - d. 社会貢献(CSR)をアピールしたい
    - i. 環境、地域との繋がり...
2. ターゲットは？
  - a. B to B、B to C、C to C
  - b. 年齢、性別、国籍、趣味/嗜好
    - i. エコ/自然志向、最先端、コスパ重視...
  - c. この過程で【ペルソナ】をおこなって、ターゲット顧客の具体的なイメージ像を描き出す作業を行うこともある。
    - × 30代の女性、独身男性
    - 都内在住(荻窪駅)、未婚、29歳、年収400万円、地方公務員...



木村 佳子さん

## スタイリッシュで快適な空間で 友人達と楽しく生きたい

職業 地方公務員  
年収 400万円  
年齢 29歳  
婚姻状況 未婚・一人暮らし  
居住地域 東京都、荻窪駅近ワンルームマンション

実家は宮城。18歳の時、大学入学を機に上京。その頃からずっと一人暮らしです。今は荻窪のワンルームマンションに住んでいて、将来は結婚もしたいけど、まだそのチャンスらしいものもなく、29歳になりました。

老後の心配はまだまだ先の話ですが、不安がないわけではありません。公務員という職業がら、今後も年収や仕事の内容に大きな変化はなさそう。今の仕事に特に不満もないので、転職は考えていません。

もし、おしゃれな物件があり、信頼できる不動産会社と出会えば、具体的に検討してもいいかもと思っています。

今の住まいは、Francfrancの家具で自分なりにおしゃれな部屋づくりをしています。料理が趣味ですが、平日は仕事で遅くなるのでできません。主に土日に楽しんでいます。ただ、キッチンが狭くコンロも一口しかないため、いろいろ作れないのが残念です。

将来は自分のマンションを持ち、友人達を呼んでホームパーティなど開けるといいなと思っています。

エピソード  
・広い部屋に住む  
・料理のできる設備がある部屋  
・ホームパーティを開く

エモショナルゴール  
快適に暮らしたい、生活を楽しみたい

ライフゴール  
友人達に囲まれた幸せな人生



よく読む雑誌「Premium」

ペルソナ参考例

出典: OFFICE ANTS <https://www.officeants.com/blog/persona.html>

- d. そのようなターゲット顧客に最適な世界観、インターフェイス、言葉遣い、機能を起こしていく。
  - i. 実際のところ、小規模なサイトではそこまでのターゲティングはできず、顧客(サイト運営者)に大まかにヒアリングした上で、相手の希望に合わせてつくることが多い。相手もどのようなデザインが良いか分からないので、制作者にお任せ。制作者のデザインセンスと知見、流行、運営者の好みの中で落とし込むことになる。「こんな風な雰囲気が良い」と他社サイトを提示されることも多い。

### 3. 必要な情報の洗い出し

#### a. 商品情報

- i. スペック、画像、動画、特長、利用シーン、よくある質問、お客様の声...

### 4. グループ分け

- a. グループごとに別ページにするのか、同一ページ内で別セクションにするのか
- b. この過程で、自ずと【サイトマップ】= ページ構成案のようなものに落とし込むことになる

### 5. グループ内での順序、優先順位、重要度

- a. この過程で、いわゆる【ワイヤーフレーム】= ページ毎のレイアウト案に落とし込むことになる
- b. コンテンツが固まってくれば、自ずとページのタイトル案も固まってくる

### 6. 表現方法の検討

- a. 文章、写真、動画、イラスト、アニメーション、事例...
  - i. かっちりとした堅い文章、やわらかい言葉遣い...
  - ii. 商品写真、商品がある風景の写真...
  - iii. 動作に特長があれば動画、仕組みに特長があればアニメーション...

## デザイン手法のTips

いきなり、色、形に頼らない。

白黒、文字だけでも必要な情報が伝わるように工夫する。  
いきなり色・形に頼ると、素人がつくるWord原稿のようになる(派手な吹き出しや囲みなど)。

分ける、離す。

デザインの基本。正しくグループ分けされていれば理解の助けになる。しっかりと余白があれば実線で区切る必要もない。太字にする必要も少ない。

並べる。

優先順に並べる。前後の並びで文脈が伝わるか。スムーズに頭に入ってくるか。

揃える。

文字の面を揃えるだけで”線”が現れる。本当に必要になるまで実線に頼らない。

## その上で初めて、大きさ・太さにメリハリを付ける。

グループ分けと分離が十分にできていれば、大サイズの文字や太字も最小限で済む。ここまでで過剰なデザインが少なければ、その分、背景や色・形など追加要素の自由度が高まる。

## 色。絶大な効果 or 絶望の結果。

この段階で初めて色を試してみよう。

- どの要素に色を付けるか
  - 背景に色
    - 全体の背景色
    - グループの背景色
    - 見出し(大サイズ・太字)の背景に色
  - 文字に色
    - 見出し(大サイズ・太字)に色
    - 本文に色
- 濃い色 or 薄い色
  - 背景色は薄く、文字色は濃く
  - 背景色が濃く、文字色は明るく
- 近似色で濃さを変える or 反対色でメリハリ

色は手強い。やりすぎてもとに戻したり、少しずつ攻めて煮詰まった挙げ句、好きな色の組み合わせを見つけたが、サイトの方向性からズレていたり…。そんな時は、同業他社のいろんなサイトを見てみよう。先人が苦勞してたどり着いた成果にはヒントが詰まっている。

「Pinterest」で「web design」で探してデザインサンプルを眺めてみるのもおすすめ。見すぎてお腹いっぱいにならない程度に。

[https://www.pinterest.jp/search/pins/?q=web%20design&rs=typed&term\\_meta\[\]=web%7Ctyped&term\\_meta\[\]=design%7Ctyped](https://www.pinterest.jp/search/pins/?q=web%20design&rs=typed&term_meta[]=web%7Ctyped&term_meta[]=design%7Ctyped)

## まとめりのある色にたどり着く1つの確実なTips。

1. まず、サイト(ページ)の扉画像になりうる素敵な写真を用意しよう。
2. グラフィックソフトのスポイトツールで写真の中で大きなウェイトを占める3～5色程度を抜き出す(スポイトの位置で結果が大きく異なることがあるので、見た目に近い色が採取できるよう何度か試そう)。

3. それぞれの色の明度・濃さなどを調整し、文字の可読性なども考慮しながら以下のようなカラーパレットをつくる。
- a. メインカラー(写真の中で一番強い色)
  - b. サブカラー(写真の中で二番目に強い色)
  - c. 差し色(写真の中に点在する色1～3色)
  - +
  - d. 白 / 黒 / グレー

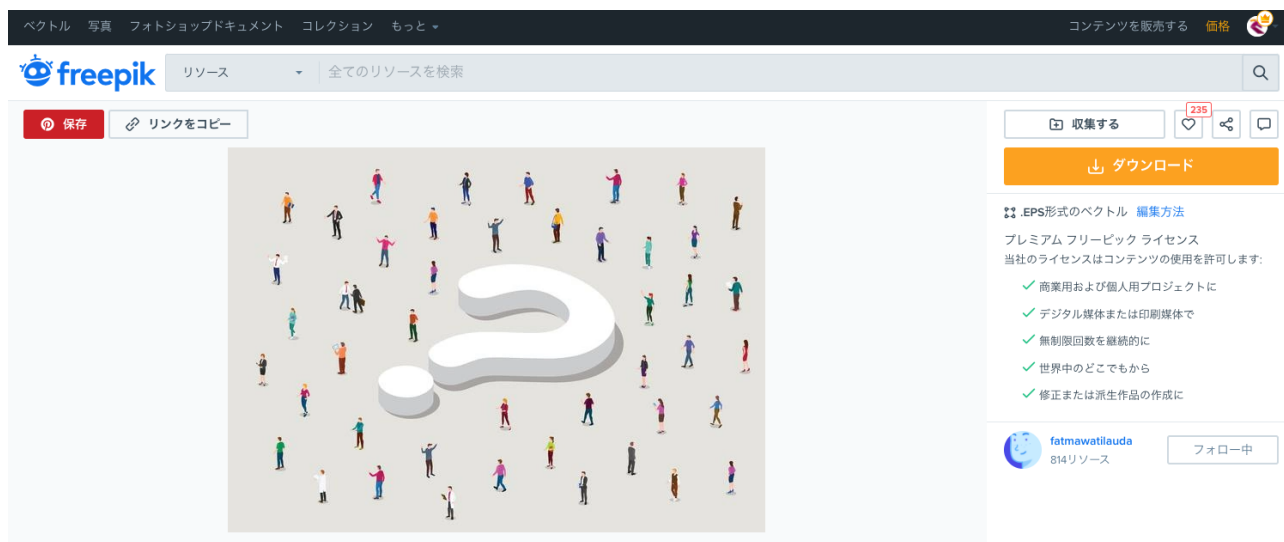
このようにメインの写真から色を取り出すことで、ページ内でまとまりのある色の組み合わせパターンが用意できる。ただし、メインになる写真のクオリティに依存する。また、写真に明確な色がないと難しい。色がない場合は逆にどのような色でも(商品イメージに合えば)使える。

## 素材集サイトを上手に使う。

「freepik」

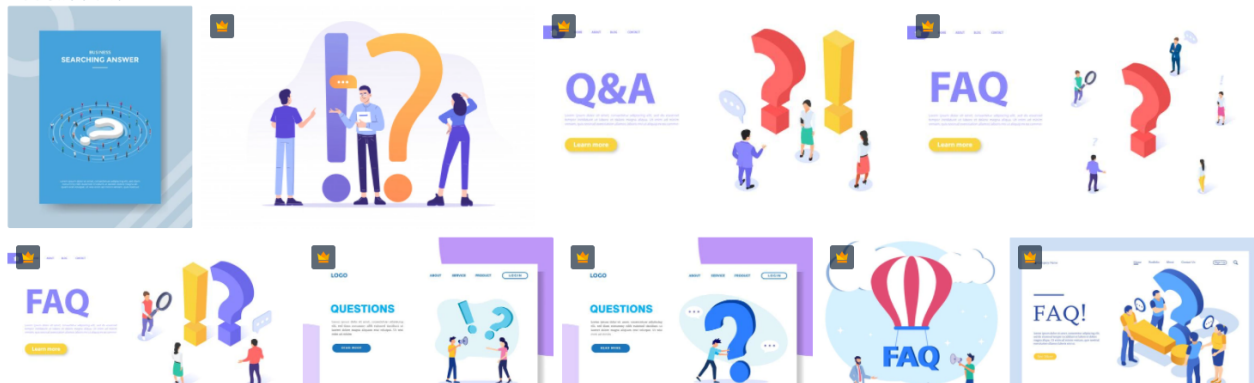
<https://www.freepik.com/>

アイコンやイラスト素材など。EPS/SVGなどグラフィックソフトで再編集可能なフォーマットも提供される。



大きな疑問符と人々がモダンなアイソメトリックスタイルのイラストで一周する答えの概念を探しています 無料ベクター

こちらもおすすめです



[https://jp.freepik.com/free-vector/searching-for-answer-concept-with-big-question-mark-and-people-circle-around-with-modern-isometric-style-illustration\\_12846681.htm](https://jp.freepik.com/free-vector/searching-for-answer-concept-with-big-question-mark-and-people-circle-around-with-modern-isometric-style-illustration_12846681.htm)

↑ この素材を使って...

↓ ヘッダー部分にアクセントとなる画像を追加(背景色 + 透過画像)



「oshiete! 掲示板 - 結いなびIIDA」<https://yuinavi.jp/oshiete/>

※グローバルメニューの画像も「freepik」の姉妹サイト「flaticon」から使用

## WordPressによるサイト制作におけるデザインの流れ

おもに以下の2つの流れが想定できる。

### 1) 使用するテーマを軸にデザインする

1. イメージに合う(機能を満たす)テーマを探す
2. カスタマイザーで大まかに作り込む
3. CSSで独自の調整(必要に応じてグラフィックソフトでパーツを作成)

国内で人気のテーマの例:

「Snow Monkey」(有償)

<https://snow-monkey.2inc.org/>

「Lightning」(有償)

<https://lightning.vektor-inc.co.jp/>

「Cocoon」(無償)

<https://wp-cocoon.com/>

「TCD」(有償)

<https://tcd-theme.com/>

## 2)オリジナルデザインを軸にテーマを選定する

1. グラフィックソフトでデザインを起こす
2. デザインを元にHTML/CSSを組み上げる(WordPress組み込み前)
3. HTML/CSSファイルを元に、オリジナルのWordPressテーマ/テンプレートを組み上げる
  - HTMLファイルの拡張子をphpに変更し、WordPressの関数を埋め込んでいく
  - header.php、footer.php、sidebar.phpなどに分割する
  - CSSファイルは名称をstyle.cssとし、テーマに必要な記述を追記し、テーマ側で読み込む
4. または、再現できそうなテーマにHTML/CSSを統合していく

## CSSとは

HTMLが文書の構造を指定するのに対して、そのHTMLにビジュアルスタイルを指定するのがCSS(カスケーディング・スタイル・シート)。**Web**ページのデザインはおもに**CSS**で行う。

## 3種類のCSSの指定方法

### 1. **CSS**ファイルを作成し**HTML**ファイル内で読み込む

拡張子に.css(HTMLは.htmlまたは.htm)が付いたファイルを作成しHTMLファイル内で読み込む。おもにhead内で以下のように指定する。

```
<link rel="stylesheet" href="style.css" type="text/css" />
```

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta name="viewport" content="width=device-width" />
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>WordPress &#8250; お読みください</title>
  <link rel="stylesheet" href="wp-admin/css/install.css?ver=20100228" type="text/css" />
</head>
<body>
<h1 id="logo">
  <a href="https://wordpress.org/">

```

body {
    background-color: silver;
    color: #000000;
}

```

</style>

```

<script type='text/javascript' defer='defer' src='https://test.yuinavi.jp/wp-content/themes/feas/js/main.js'></script>
<script type='text/javascript' defer='defer' src='https://test.yuinavi.jp/wp-content/themes/feas/js/main.js'></script>
<link rel="https://api.w.org/" href="https://test.yuinavi.jp/wp-json/wp/v2/" type="application/json" />
<link rel="alternate" type="text/xml+oembed" href="https://test.yuinavi.jp/oembed?url=https://test.yuinavi.jp/" />

<script type="text/javascript">var ajaxurl = 'https://test.yuinavi.jp/wp-admin-ajax.php';</script>

<style type="text/css">
#feas-searchform-2 {
    margin: 2rem 1rem;
}
.feas-row {
    display: flex;
    flex-direction: column;
}
.feas-midashi {
    /* width: 8rem; */
    font-weight: 600;
    color: gray;
}

```

HTMLファイルのhead内に直接CSSが記述されている様子

### 3. HTMLタグにインラインで記述

CSSを適用したいHTMLタグ内に以下のような書式でstyle属性を使って直接CSSを記述する。一回限りの使用で、手っ取り早くCSSを適用したい場合などに使用することが多い。

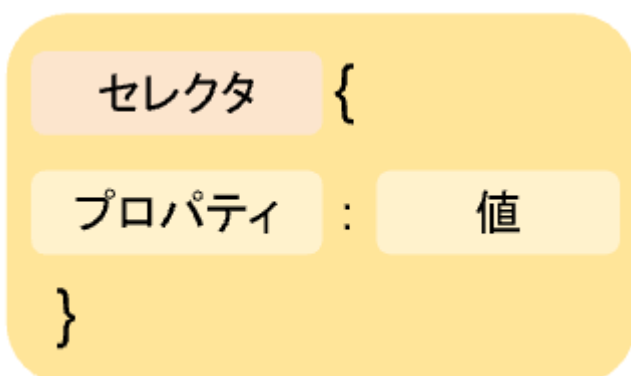
```
<dl style="font-size:18pt;color:red;">
  <dt>価格</dt>
  <dd>1,000円</dd>
</dl>
```

※HTMLタグに直接書くので、IDやClassなどの指定は不要

```
n__link" href="/register/">マイページをつくる<br><span style="font-size:0.7em;color:silver;">要メンバー登録</span></a></div>
```

HTMLタグにインラインでCSSが記述されている様子

## CSSの書式



出典:ITSakura <https://itsakura.com/css-selector-zokusei>

```
body {
  background-color: silver;
  color: #000000;
  font-size: 16px;
  font-family: "Yu Gothic", "Meiryo", "Hiragino Sans", "Helvetica", "Arial", sans-serif;
}
```

## おもなセレクトア

CSSを適用するターゲットを指定する。HTMLタグ/ID/Classの3種類のセレクトアがある。

**HTMLタグ**(body,header,h1など)

HTMLタグそのものに対して指定する。

そのタグが現れる際はすべて同じCSSが適用される。

そのタグのデフォルト値を指定するようなイメージ。実際にはIDやClassを振ることでデフォルト値を上書きするような使い方が多い。

**ID**(#mainなど)

HTMLタグの中で id="〇〇" と振られているものに対してCSSを適用する場合。

IDはCSS上では冒頭に # (シャープ) を付け表す。#main とは HTML上では id="main" を指す。

IDは **同一ページ内で一度しか使用できない** (ゆえに "ID")。

**Class**(.entry-titleなど)

HTMLタグの中で class="〇〇" と振られているものに対してCSSを適用する場合。

ClassはCSS上では冒頭に . (ドット) を付けて表す。.entry-title は HTML上では class="entry-title" を指す。

Classは **同一ページ内で複数回使用できる** (ゆえに "Class")。

同じフォーマットの一連のコンテンツをページ内で繰り返し表示する場合などに使用。

## セレクターの限定方法

#main .title

idがmainの以下のclassにtitleが付いた項目(#main以下の下階層にも適用)。

もっともよく使う記述方法。

#main > .title

idがmain直下のclassにtitleが付いた項目(下階層には適用されない)。

ドロップダウン形式のグローバルメニュー(<li>の入れ子構造)のトップ階層だけアイコンを表示したい、などの場合に便利。

#main.title

idがmainかつclassにtitleが付いた項目(同じ要素内にidとclassを持つ場合)

その他さまざまな指定方法がある。以下を参照のこと。

参考: CSSセレクター - MDM Web Docs

[https://developer.mozilla.org/ja/docs/Web/CSS/CSS\\_Selectors](https://developer.mozilla.org/ja/docs/Web/CSS/CSS_Selectors)

## CSSのプロパティ

指定したセレクターに対して適用するスタイルを指定する。文字色、文字サイズ、背景色、背景画像、枠線(ボーダー)、幅、高さなど多岐にわたる。

基本的ないくつかを紹介する。

font-size 文字サイズ

font-weight 文字の太さ

font-family 書体

color 文字色

background-color 背景色

background-image 背景画像

border 枠線、ボーダー

border-radius 角丸の指定  
margin オブジェクト周囲の余白  
padding オブジェクト内部の余白  
display 表示形式...インライン、ブロックなど

## CSSの値

指定したプロパティの値を指定する。プロパティによって受け付ける値・書式は多岐にわたる。

基本的ないくつかを紹介する。

**font-size: 16px;** // 文字サイズは16ピクセル (他に 16pt, 90%, 1em, 1rem など)  
**color: red;** // 文字色は赤 (他に #ff0000, rgb(255, 0, 0), hsl(360, 100%, 50%) )  
**border: 1px solid #ff0000;** // 1ピクセルの実線 (点線)、色は赤  
**margin: 1em auto;** // 上下が1文字分の余白、左右が自動 (均等になりセンタリングされる)  
**padding: 1em 2em 3em 10em;** // 上 1文字分、右 2文字分、下 3文字分、左 10文字分の余白

参考: CSSプロパティ - MDM Web Docs

<https://developer.mozilla.org/ja/docs/Glossary/property/CSS>

## CSSの基本的で大事なルール

### 後から指定されたものが優先される

同一ファイル内に同一セレクトが複数回指定された場合は、後に指定されたものによってプロパティ単位で上書きされる。複数ファイルの読み込み、同一ファイル内、インライン、いずれの場合も最終的に読み込まれたものが優先。ただし、**!important** 属性が付いたものは例外。

```
.title {  
    color: #000000 !important;  
    font-size: 16px;  
    font-family: "Yu Gothic", "Meiryo", "Hiragino Sans", "Helvetica", "Arial", sans-serif;  
}  
#main .title {  
    color: red;           ... 上の設定で !important が付いているので上書きされない。  
    font-size: 24pt;      ... font-sizeを24ptに上書き。  
    font-family:          ... font-familyは指定がないので上の設定が引き継がれる。  
}
```

## WordPressにおけるCSSの設定方法

おもに4つの方法が考えられる。

### 1. テーマに同梱されるstyle.cssに追記

WordPressのテーマには必ずstyle.cssが含まれる。style.cssはテーマのスタイルの指定だけでなく、テーマの名称・バージョン・作者など、WordPressがテーマを認識するのに必要な記述が含まれるため (style.css冒頭のそれらの記述は削除してはいけない)。

2. 独自のCSSファイルを作成し、テーマに読み込む(head内にlink要素にて指定)
3. テーマのカスタマイザーにある「追記CSS」欄に書く(テーマがhead内に出力してくれる)
4. 各プラグインごとに提供されるCSS欄に書く(テーマがhead内に出力してくれる)