

WEB基礎演習

第5回

2019年10月25日(金)

- ・前回と同じところに座って下さい(毎回同じ席)
- ・第5回のノートをダウンロード→開いておく
- ・181教室でのログイン
 - 2限のユーザ名: webBasic
 - 5限のユーザ名: Kudo Seminar Pro4
 - パスワード: kudo

Today's Agenda

本日の内容

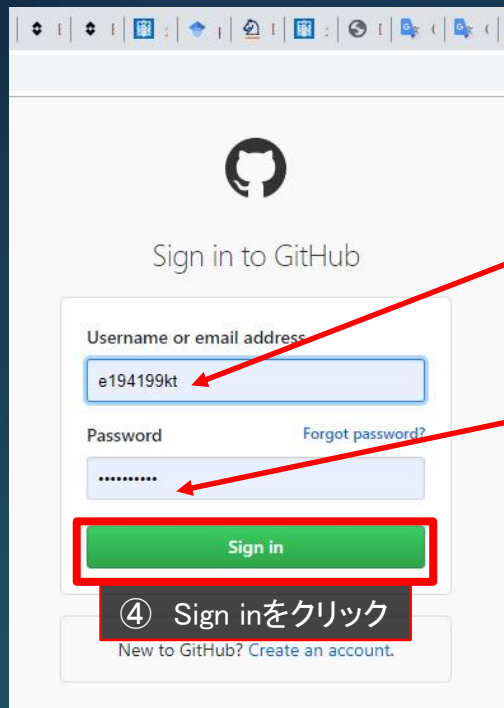
- 第2回小テスト: WEB技術の基本用語2
 - 課題2のアップロード
 - CSSの基本 (Webページを装飾する)
 - CSSとは
 - CSSを適用する3つの形式
 - 形式1: style属性でその要素にだけ適用
 - 形式2: <style>タグでそのHTML文書全体に適用 / CSSの基本書式
 - 形式3: スタイルシートを書いた別ファイル(.css)を作る
-

第2回小テスト
-WEB技術の基本用語 2-

課題 2 のアップロード

1) GitHubでログイン⇒リポジトリ選択

a) まずはgithubのページに行って[学籍番号の文字列]でログイン



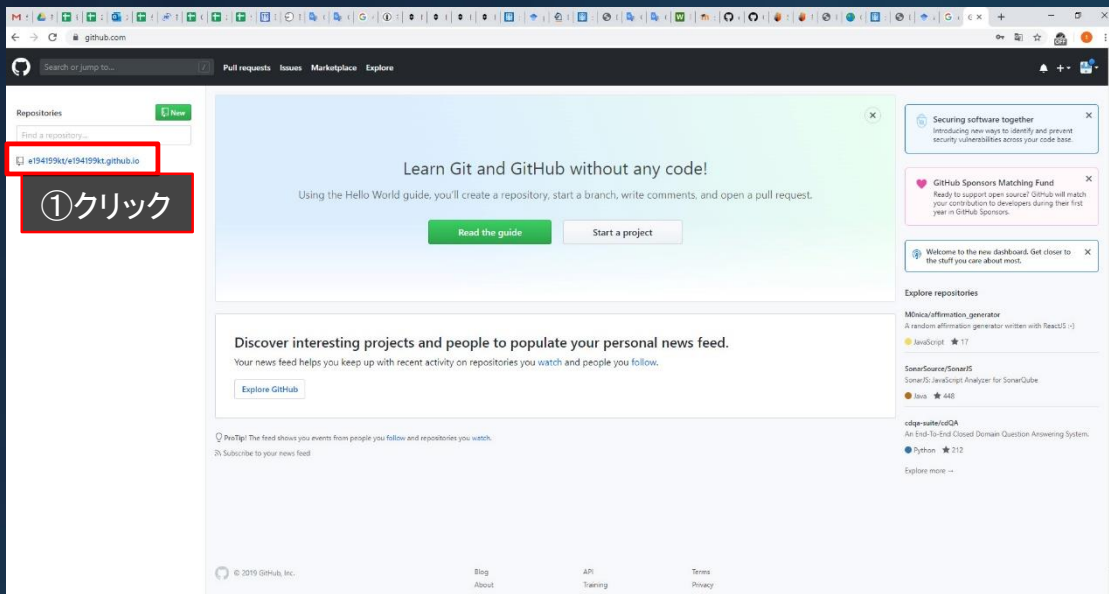
② Usernameに[学籍番号の文字列]
(メールアドレスの@の前の文字)を入れる

③ Passwordはwebbasic2019

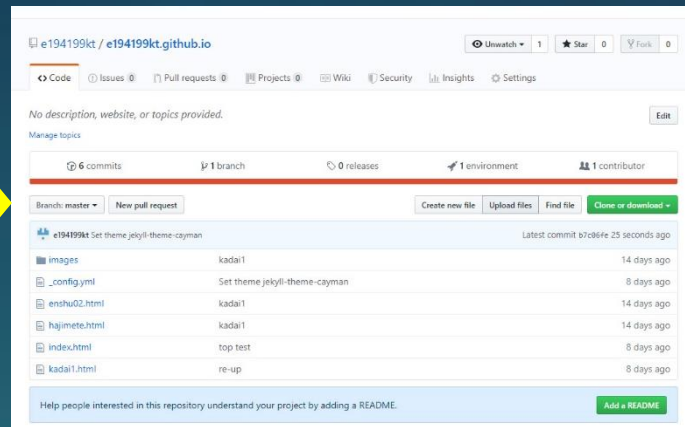
1) GitHubでログイン⇒リポジトリ選択

b) [学籍番号文字列].github.io リポジトリ (Github Pages用) を選択

ユーザトップページ

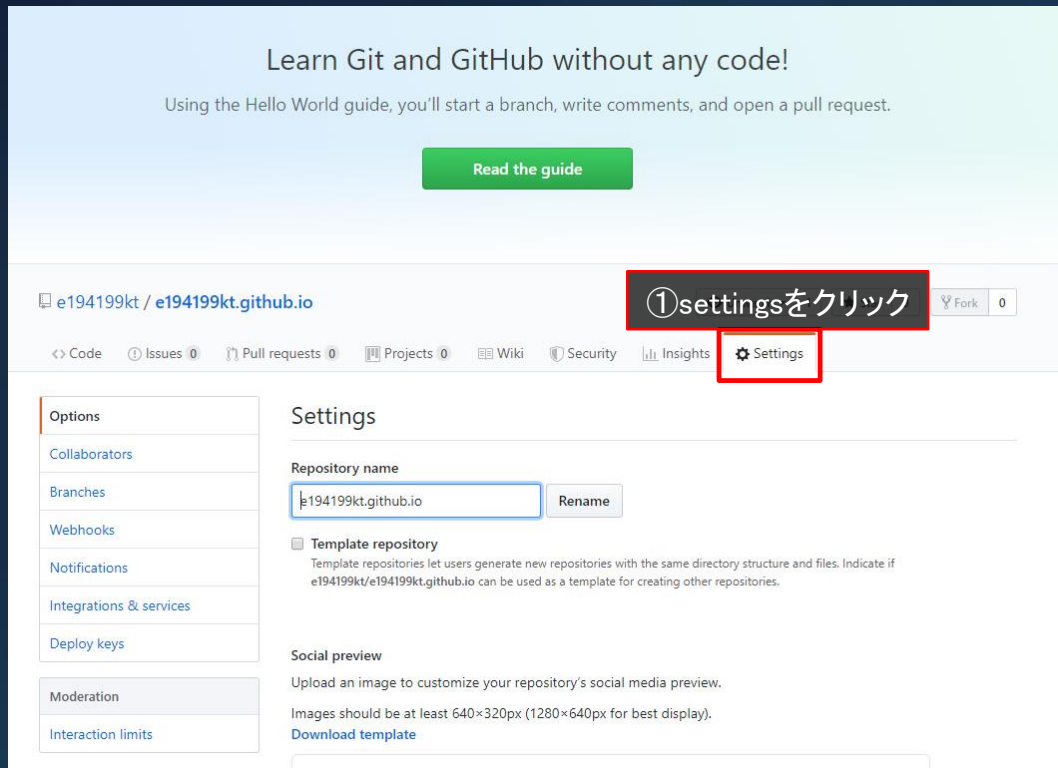


② リポジトリのファイル閲覧ページへ

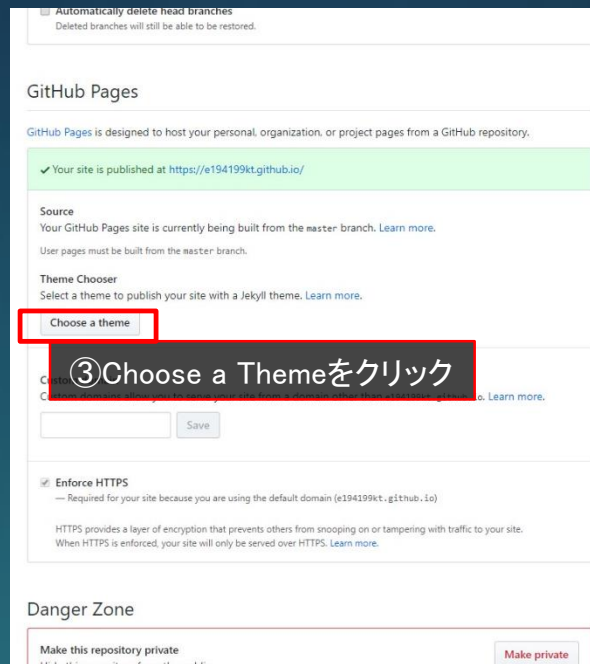


2) GitHub Pagesのテーマを選択

a) **Setting**ボタンを押し、テーマ選択画面へ

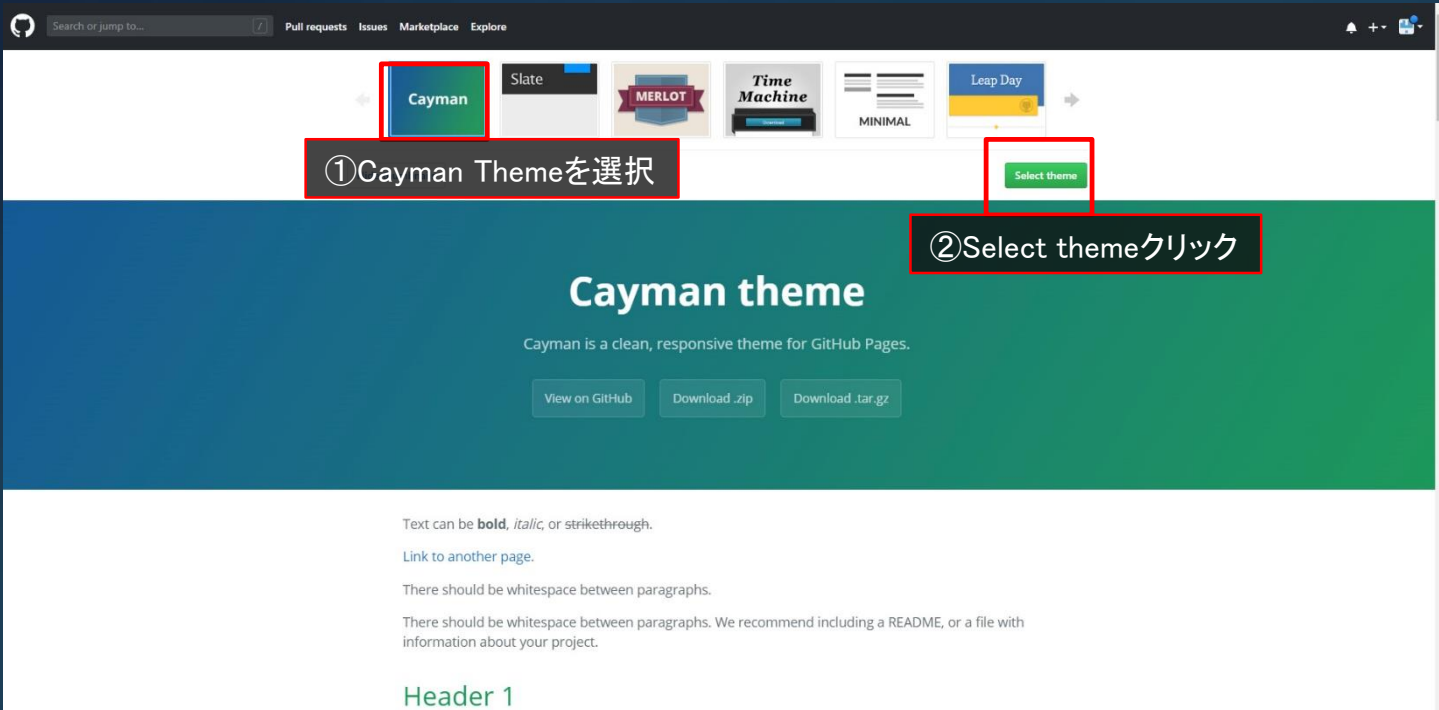


② 下へスクロールしていくとGitHub Pagesの項目がある



2) GitHub Pagesのテーマを選択

b) どれでも良いのでテーマを選択（この作業は課題2用、一度やればよい）



The screenshot shows the GitHub Themes selection page. The 'Cayman' theme is highlighted with a red box. Below it, a red box contains the text '①Cayman Themeを選択'. To the right, a green 'Select theme' button is also highlighted with a red box, with a red box containing '②Select themeクリック' below it. The main content area displays the 'Cayman theme' preview with the text 'Cayman is a clean, responsive theme for GitHub Pages.' and buttons for 'View on GitHub', 'Download .zip', and 'Download .tar.gz'. Below the preview, there is a text editor showing sample text and a 'Header 1' section.

①Cayman Themeを選択

②Select themeクリック

Cayman theme

Cayman is a clean, responsive theme for GitHub Pages.

View on GitHub Download .zip Download .tar.gz

Text can be **bold**, *italic*, or ~~strikethrough~~.

Link to another page.

There should be whitespace between paragraphs.

There should be whitespace between paragraphs. We recommend including a README, or a file with information about your project.

Header 1

3) ファイルのアップロード(単純な方法)

a) File Uploadボタンを押し、アップロード画面へ

Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

[Read the guide](#)

e194199kt / e194199kt.github.io

Unwatch 1 Star 0 Fork 0

①Codeをクリック

Code Issues Pull requests Projects Wiki Security Insights Settings

6 commits 1 branch 0 releases 1 environment contributor

Branch: master New pull request

Create new file **Upload files** Find file Clone or download

②Upload filesをクリックしてアップロード画面へ

File name	Author	Latest commit
images	kadai1	
_config.yml	Set theme jekyll-theme-cayman	
ensu02.html	kadai1	14 days ago
hajimete.html	kadai1	14 days ago
index.html	top test	8 days ago
kadai1.html	re-up	8 days ago

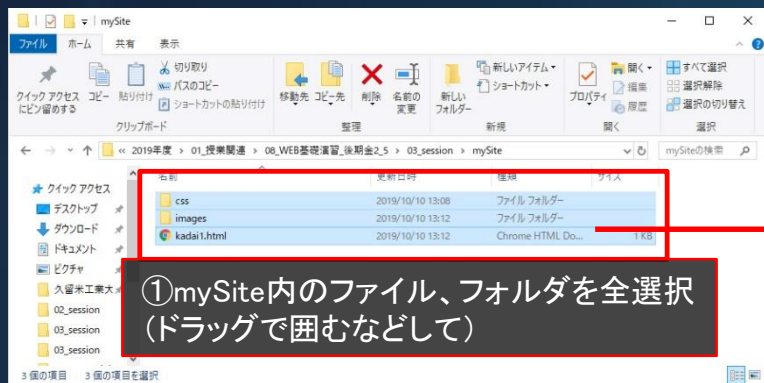
Help people interested in this repository understand your project by adding a README. [Add a README](#)

© 2019 GitHub, Inc. Terms Privacy Security Status Help

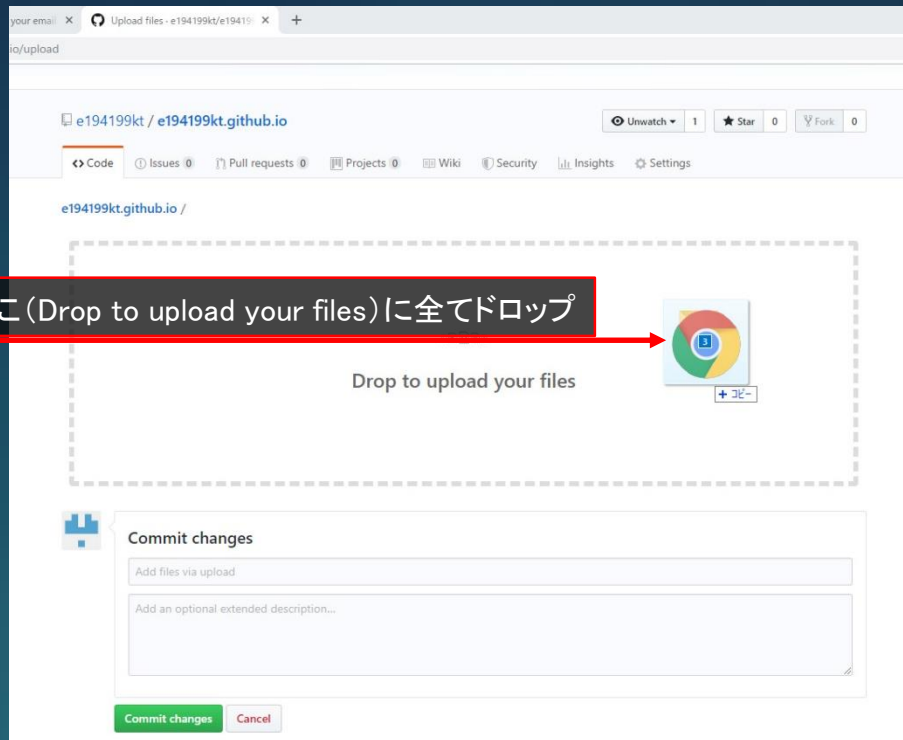
Contact GitHub Pricing API Training Blog About

3) ファイルのアップロード(単純な方法)

b) mySite内の全てのファイル・フォルダ選択⇒ドロップ



imagesやcssフォルダの中の
ファイルをばらばらに
アップロードしないように



3) ファイルのアップロード(単純な方法)

c) コメントを書いて **Commit changes** でアップロード完了

The screenshot shows the GitHub repository page for `e194199kt / e194199kt.github.io`. The repository is currently empty. A red box highlights the text "Drag additional files here to add them to your repository" and "Or choose your files". Below this, a list of files to be uploaded is shown: `kadai1.html` and `/images/01.jpg`. A red box highlights the "Commit changes" button. Another red box highlights the text "課題1のアップ" (Upload of Lesson 1) in the commit message field. A third red box highlights the text "①アップロード予定のファイルが表示" (Upload of files to be displayed). A fourth red box highlights the text "②コメント(任意)" (Optional comment). A fifth red box highlights the text "③クリック" (Click).

①アップロード予定のファイルが表示

②コメント(任意)

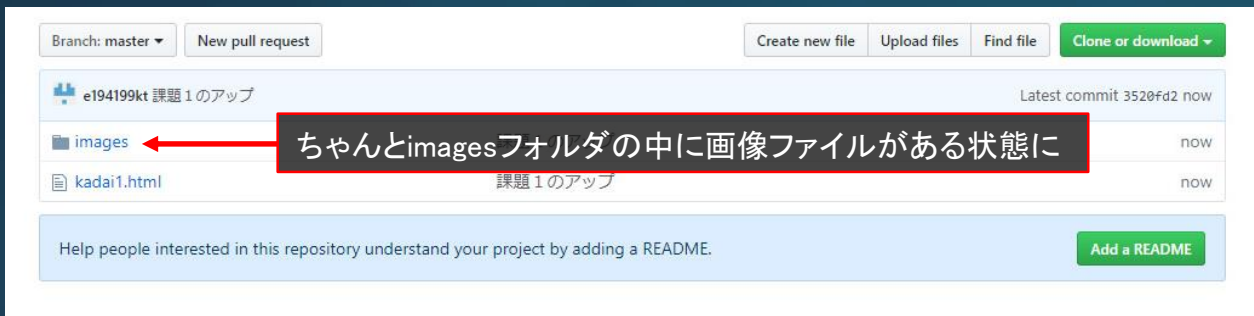
③クリック

The screenshot shows the GitHub repository page for `e194199kt / e194199kt.github.io`. The repository now contains a commit. A red box highlights the text "Learn Git and GitHub without any code!" and "Using the Hello World guide, you'll start a branch, write comments, and open a pull request." Below this, a green button "Read the guide" is visible. Another red box highlights the text "e194199kt 課題1のアップ" (Upload of Lesson 1) in the commit message field. A third red box highlights the text "④アップロード完" (Upload complete).

④アップロード完

3補足) 必要なものだけアップロードもできる

- 課題に必要なファイルだけアップロードすることもできる
課題2の場合: `index.html` と 画像ファイル
- ただし、必ず `mySite` と同じ階層構造になるようにアップしよう
例: 画像ファイルは、GitHub上でも `images` フォルダの中に入ってから
Upload Filesをクリックしてアップロードする



4) 表示の確認 ([https://\[学籍番号文字列\].github.io/](https://[学籍番号文字列].github.io/))

- ・ 表示の確認は、必ずしておこう

課題2のURL

<https://e○○○○.github.io/>



自分のGitHubアカウント名
(学籍番号+2文字)

今回トップページなので
ファイル名が必要ない



- 1) ちゃんとページが出るか
- 2) リンクは正しく動いているか

確認

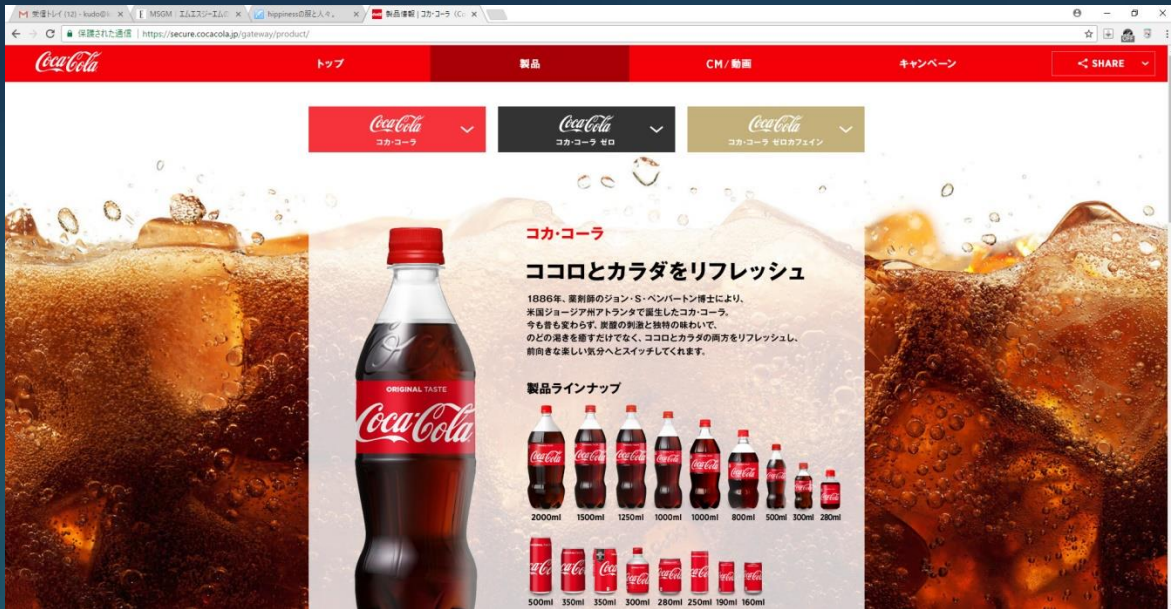
CSSとは

HTMLだけでは何かが足りない

皆が作れるWebページ



コカ・コーラのWebページ



装飾が足りない！



文字色、背景色、背景画像、
要素の配置などをもっと作り込みたい

CSS

- Cascading Style Sheetsの略
- Webページのスタイルを指定するための言語
(色や大きさ、配置などの装飾のこと)

or この言語で書かれた文書ファイル自身

- 単純にスタイルシートと呼ばれることが多い
- これを用いればHTML文書はそのまま、様々な見た目に変えられる

スタイルシート

```
セクタ  
p {  
  color: #F80206;  
}  
プロパティ 値
```


例えばCSSで何ができる？

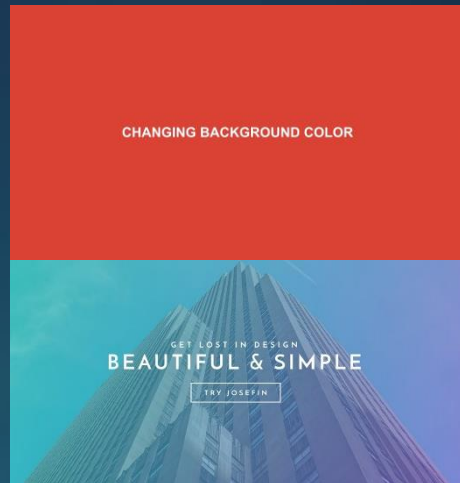
<文字の装飾>

- ・サイズ
- ・フォント
- ・色

フォントを変更する
フォントを変更する
フォントを変更する
フォントを変更する
フォントを変更する
フォントを変更する

<背景の装飾>

- ・色
- ・背景画像



<配置のコントロール>

- ・要素(例えばtableによる表)を
ページ内でどう配置するか



...etc

CSSを適用する 3 つの方法

CSSの適用法は3種類

1) タグにstyle属性を追加する
→ その要素だけに直接適用

2) HTML文書のhead内にstyleタグを書く
→ そのHTMLファイルに適用

3) 手順1. CSSだけの別ファイル(.css) を作る
手順2. head内でlinkタグを使って読み込む
→ どのHTMLファイルにも読み込める(装飾を使いまわせる)

↑ この3番目の形式が一番使われる

さっそく1つずつ体験していこう

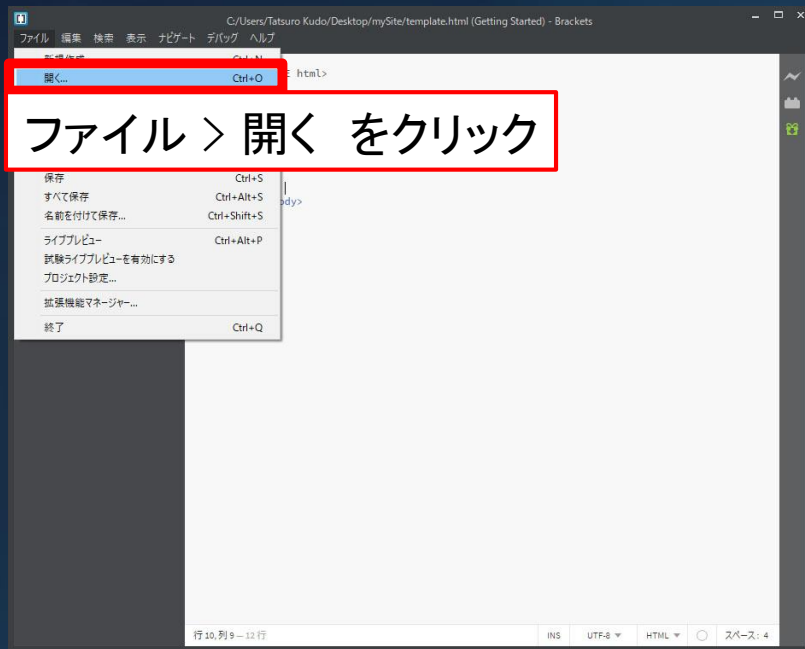
Brackets



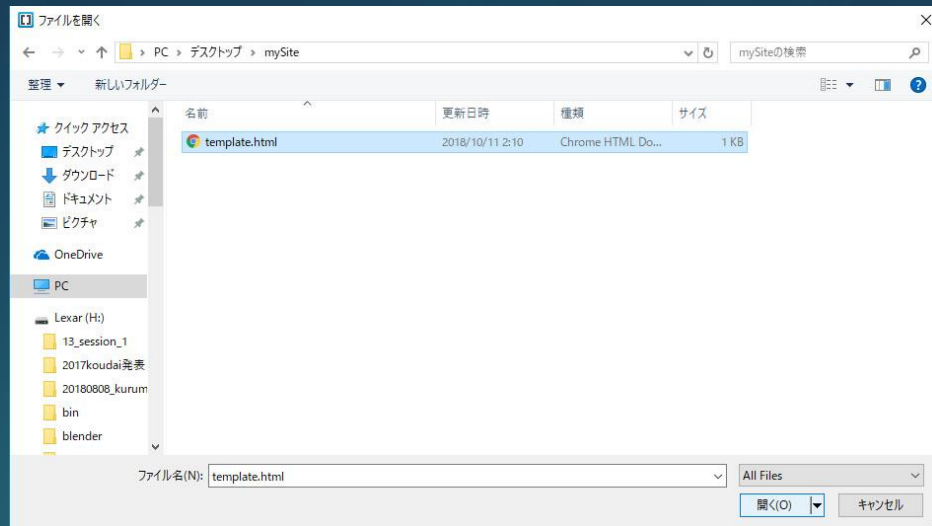
左下のスタートメニュー(あればショートカット)から起動しよう

演習用ファイルを3つ作ろう

1. template.htmlを開く



デスクトップ > mySite > 「**template.html**」を選択 > 開く



演習用ファイルを3つ作ろう

2. ベースとなるhtml文書の作成(body部分に書き加えよう)

```
<body>
  <h1>見出し1</h1>
  <p>
    あいうえお
  </p>

  <h1>見出し2</h1>
  <p>
    かきくけこ
  </p>
</body>
```

3. 別の名前の3つのhtmlファイルとして保存

「enshu03_direct.html」「enshu03_head.html」「enshu03_load.html」という3つの名前でデスクトップ > mySite 内に保存(名前をつけて保存を3回)

1. あるタグ要素に直接適用する

編集ファイル

enshu03_direct.html

1. タグ要素に直接適用する

- 開始タグにstyle属性を追加して、その要素にだけ適用する

書式 : <タグ名 style=" プロパティ : 値 ; " > 内容 </タグ名>

例1 (pタグに文字色を赤にする装飾を施す) :

```
<p style="color:red;"> あいうえお </p>
```

例2 (pタグに複数のプロパティを設定する. 太字の追加) :

```
<p style="color:red;font-weight:bold;">
```

「プロパティ:値;」を並べて書く

2. あるHTMLの全体に適用する

編集ファイル

enshu03_head.html

2. そのHTML全体に適用する

- head内にstyleタグを書き、HTML文書単位で適用する

書式 (*headタグの内部に書く*) :

```
<style>  
  セレクタ {  
    プロパティ: 値;  
  }  
</style>
```

2. そのHTML全体に適用する

- head内にstyleタグを書き、HTML文書単位で適用する

書式 (*headタグの内部に書く*) :

<style>

```
セレクト {  
    プロパティ: 値;  
}
```

</style>

これがCSS言語の基本形式
(しっかり押さえよう)

CSS言語の基本形式

スタイルの適用先

セレクタ {

プロパティ : **値** ;

設定する項目名

設定値

}

例:

p {

color : **red** ;

}



pタグの**文字色**を**赤**に装飾する の意味

2. そのHTML全体に適用する

- head内にstyleタグを書き、HTML文書単位で適用する

例1 (pタグに赤文字スタイルを適用) :

```
<style>
    p {
        color: red;
    }
</style>
```

先ほどと違い、
全てのpタグ要素に
適用される

2. そのHTML全体に適用する

- head内に**styleタグ**を書き、**HTML文書単位で適用**する

例2 (pタグに**複数のスタイルを適用**。太字追加) :

```
<style>
```

```
  p {
```

```
    color:red;
```

```
    font-weight : bold;
```

```
  }
```

```
</style>
```

「プロパティ:値;」を並べて書く

1)の””の中身は、
この部分を書いて
いた

3. CSSだけを書いた別ファイルを読み込む

編集ファイル

enshu03_load.html

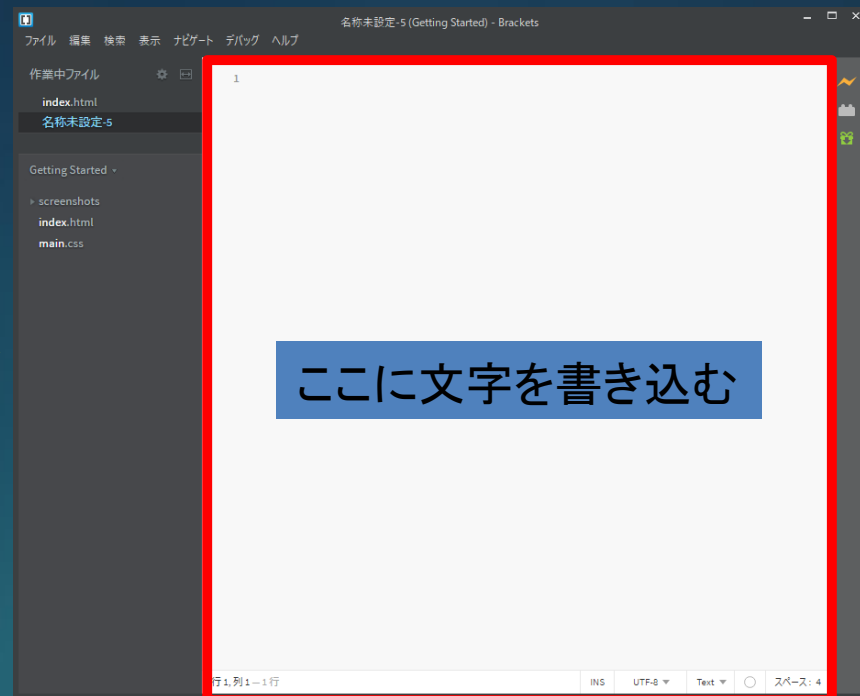
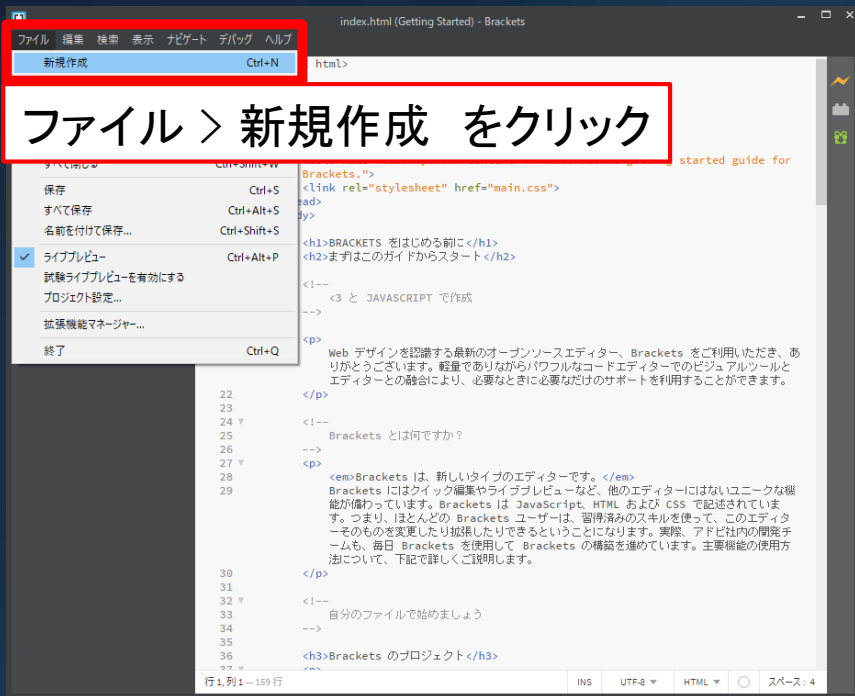
3. CSSだけを別ファイルで作成

- CSSだけを書いた別ファイル(.css)を作り、読み込む

まずは `brackets` でCSS形式のファイル
「enshu03.css」を作成しよう

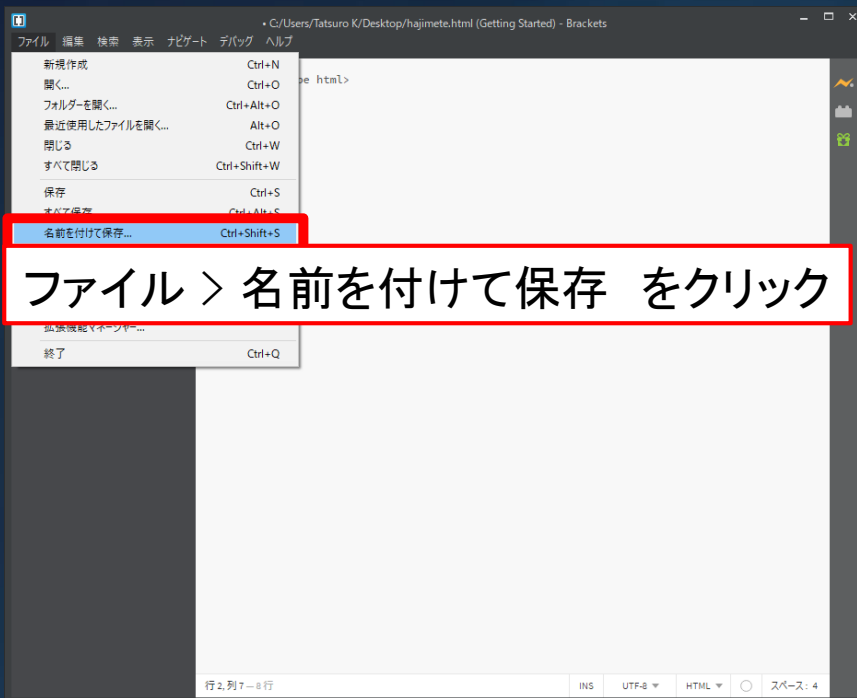
CSSファイルを作ろう

1. 編集ファイルの新規作成

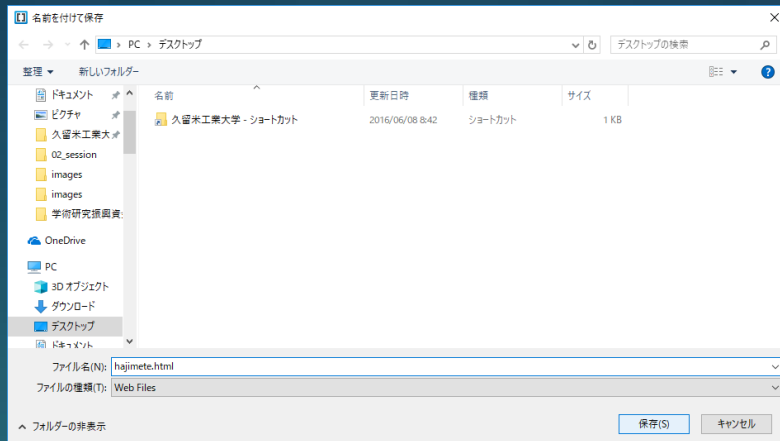


CSSファイルを作ろう

2. CSSファイルとして一度保存(これで候補表示などが働くように)



今回はデスクトップ > mySite > cssに
「enshu03.css」という名前で保存しよう



3. CSSだけを別ファイルで作成

- CSSだけを書いた別ファイル(.css)を作り、読み込む

1).cssファイルへの記述

.cssファイルには、CSS言語の書式で記述(2.のstyleタグの中と同じ形式)

```
p {  
    color:red;  
    font-weight : bold;  
}
```

3. CSSだけを別ファイルで作成

- CSSだけを書いた別ファイル(.css)を作り、読み込む

2).htmlファイルへの読み込みと適用

書式 (*headタグの内部に書く*) :

```
<link rel="stylesheet" type="text/css" href="cssファイルの場所と名前">
```

例 (*先ほどのenshu03.cssを読み込む*) :

```
<link rel="stylesheet" type="text/css" href="css/enshu03.css">
```

この書き方は後の授業で詳しくやるが、これはcssフォルダの中のファイルを指定している

複数のタグ要素にスタイルを適用するCSS

- 「セレクトア{プロパティ、値}」のかたまりを縦に並べて書く

例 (pタグだけでなくh1タグも装飾) :

```
p {  
    color:red;  
    font-weight:bold;  
}  
  
h1 {  
    color:blue;  
    font-size:35px;  
    font-weight : bold;  
}
```