

2019年6月28日
株式会社インプレスR&D
<https://nextpublishing.jp/>

プログラミング本の第一人者 田中賢一郎氏の好評を博したセミナーをテキスト化！
『Web 技術速習テキスト』発行
最新の Web 技術をフロントエンドからサーバーサイド、データベースまで手際よく解説

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『Web 技術速習テキスト』（著者：田中賢一郎）を発行いたしました。

『Web技術速習テキスト』

<https://nextpublishing.jp/isbn/9784844398769>



著者: 田中 賢一郎

小売希望価格: 電子書籍版 2100 円(税別) / 印刷書籍版 2600 円(税別)

電子書籍版フォーマット: EPUB3 / Kindle Format8

印刷書籍版仕様: B5 判 / モノクロ / 本文 330 ページ

ISBN: 978-4-8443-9876-9

発行: インプレス R&D

<<発行主旨・内容紹介>>

本書は、プログラミング本の第一人者である田中賢一氏が、好評を博したセミナーのテキストをもとに、大幅に手を加えたものです。

最新の Web 技術をフロントエンドからサーバーサイド、データベースまで分かりやすく解説いたしました。Web アプリ関連技術の進化は驚くほど速く、カバーすべき範囲も増える一方で、キャッチアップするのは容易ではありません。ブラウザ上で実行されるフロントエンドだけでなく、通信プロトコル、サーバーサイドでの処理、データベースと関連技術の範囲は多岐にわたります。最近ではクラウド上のサービスを活用するケースも増えています。どこからどう着手すればよいのかわからず、茫然としてしまう人も多いでしょう。

そこで本書では、以下のような点に留意し解説しています。

- 例や図を用いてわかりやすく説明すること。

- ・技術の歴史的な背景や、その流れに言及すること。
- ・手を動かしながら理解を深められること。

対象読者としては、プログラミングの勉強を始めた新社会人や学生の方を想定しています。

サンプルはJavaScript とPython、HTML/CSSで記述しています。JavaScript やPython などの若干の知識・経験があると、より内容を深く理解できる必携の1冊です。

(本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。)

第1章 インターネットプロトコルの基礎

1.1 IPアドレス

IPアドレスはコンピュータの電話番号のようなもので、コンピュータが通信をするときの土台となります。まずはIPアドレスがどのようなものか見ていきましょう。

1.1.1 IPアドレスとは

内々前のごとく、郵便を届けるには住所が必要で、電話をかけるには相手の電話番号が必要で、同じようにコンピュータを使って通信するには通信相手のIPアドレスが必要になります。

郵便番号は5桁です。電話番号は10桁もしくは11桁が一般的です。IPアドレスは「.」が区切りな4桁の数字(2進数)です。

128.188.1.12

コンピュータにはこのような数値を処理することは何の問題もありますが、人間にとっては見られる量ではありません。そこで、以下のように変換することになりました。

- 1) 8ビットごとに分ける
- 2) 2進数を10進数に変換する
- 3) 4つの10進数をドットで連結する

128.188.1.12

これが一般的にIPアドレスと呼ばれるものです。このように4つの数字を区切って表露することで読みやすくなります。インターネットにつながるPCにはすべてこのようなIPアドレスが割り当てられています。

では、今使っている端末にどのようなIPアドレスが割り当てられているか確認してみましょう。

【Windowsの場合】

「コマンドプロンプト」を立ち上げて、ipconfigと入力して実行してください。

「IPアドレス」の欄に書かれている値が、一般に「IPアドレス」と呼ばれるものです。

【MacOSの場合】

「ターミナル」を開き、ifconfigと入力して実行してください。

「inet」の後ろに書かれている値が、一般に「IPアドレス」と呼ばれるものです。

一般的にIPアドレスを調べてみると多くの人が、192.168.x.yもしくは10.x.y.zといったIPアドレスだったと思います。これは、RFC1918という文書で、以下の範囲をLAN(ローカルエリアネットワーク)用に予約すると記述してあるためです。

10 | 第1章 インターネットプロトコルの基礎

第2章 HTTPの基礎

2.6 URL

今まであまり詳しく説明をせずにURLを使用してきました。HTTPのリクエストラインにも関係してくるのでここで整理しておきましょう。

サーバのアドレスを表露するためにURLという用語が使われています。

- ・URL: インターネット上の場所(どのサーバの、どのリソース)を示すもの
- ・URN: インターネット上の名前を示すもの。書籍に割り当てられるISBN番号(urn:isbn:0451458523)のようなものと思ってください。
- ・URI: URLとURNの総称です。これらを厳密に使い分けるケースもありますが、URLとURIはほぼ同じ意味で使用されることが多いようです。本書ではURLという呼称を使用します。

2.6.1 URLのフォーマット

一般的なURLという以下のようなものをイメージするからしれません。

https://www.google.com

https://www.yahoo.co.jp

実はURLにはいろいろな情報を含めることができます。厳密には以下のようなフォーマットに従います。

プロトコル ホスト パス パンク

http://user:pass@test.blur.com:8080/dir/path/file?query=string&hash=xxxx

認証情報 ポート クエリストリング

- ・プロトコル: http, httpsなど

「認証情報」: アカウント・パスワードを指定する必要がある場合に指定します

・ホスト: 接続先のホスト

・ポート: ポート番号(省略時はhttpなら80, httpsなら443)

・パス: リソースへのパス

・クエリストリング「[名前=値]」の形式でのパラメータ情報。複数ある場合は?key1=value1&key2=value2のように「&」で結合します。クエリパラメータと呼ばれることもあります。

・ハッシュ: フラグメントとも呼ばれ、もともとは文書内の場所を示すものです

これらの情報をすべて暗記することはまれです。必要な情報は省略することができます。

2.6.2 URLエンコーディング

クエリストリングを扱うとURLの一部に何らかのデータを埋め込むことができました。URLによるデータ送信はまさにクエリストリングを活用していたことになります。

ただし、日本語や特殊文字などはそのままURLに埋め込むことができません。そのような文字をURLに埋め込むことをURLエンコード、逆に戻すことをURLデコードと呼びます。各文字を%xx(xxは16進数)に置き換えるのでパーセントエンコードと呼ばれることもあります。

エンコード・デコードするページを作成してみました。

```

<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8">
<title>URL Encode & Decode</title>
<script>
function urlEncode(){
var x = document.getElementById("decoded").value;

```

26 | 第2章 HTTPの基礎

第4章 jQuery の基礎

4.1 JavaScriptでのDOM操作

HTMLが登場したころはWebページに動きは全くありませんでした。今となってはいろいろな動きのあるページも珍しくありませんが、これはJavaScriptのおかげです。

jQueryはJavaScriptで最も広く使われているライブラリの1つです。ホームページを作成するときには様々な機能（画面の状態やネットワークアクセスなど）が含まれています。jQueryが登場する前は、ホームページの状態を監視するためにJavaScriptを使って画面DOMを操作していました。DOM (Document Object Model) とは「HTMLのページを構成する個々の要素をオブジェクトとして扱うようにするモデル」です。ちょっとわかりづらい説明ですね。簡単なHTMLページを作成してみましょう。

```
● <code>index.html</code>
<DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>jQuery Sampler</title>
</head>
<body>
  <input id="name">
  <button id="hit">Hit</button>
  <span id="greet"></span>
</body>
</html>
```

上記ページの構造を図にすると以下のようになります。



それぞれの要素が「ノード」に該当します。ルート（最上位の階層）はhead要素があり、その子供としてhead要素とbody要素があり——という具合です。個々の要素をオブジェクト（モノ、目的という意味）として扱うためのモデルがDOMです。つまり、DOMでは個々の要素を「ページを構成するモノ」のように扱います。例えばinput要素の場合を考えてみます。HTMLではidやvalueなどの属性をもっている要素として記述します。

```
input要素
<input id="name" value="hello">
属性名 属性値1 属性値2 属性値3
DOM
inputオブジェクト
{
  id: "name",
  value: "hello"
}
```

ブラウザは、このinput要素をinputオブジェクトに変換します。すると、このinput要素はidやvalueといったプロパティを持つオブジェクト（モノ）としてJavaScriptから操作できるようになります。

JavaScriptから見えるオブジェクトとはどのようなものでしょうか。開発者ツールを使って確認してみましょう。ブラウザで上記ページを表示した状態で、開発者ツールを起動してください。



Sourcesタブを選び、Watchウィンドウを開きます。ブラウザのウィンドウサイズや個人設定によりウィンドウの配置が異なるのでご注意ください。そのWatchウィンドウの+をクリックして表示される入力の階層に「document.getElementById("name")」と入力します。するとinputオブジェクトが表示されます。Watchウィンドウの右には編集ボタンが表示され、クリックすると、accept, acceptKey, altKeyから始まる、膨大な数のプロパティがあることがわかります。

第6章 Bootstrap の基礎

6.1 メディアクエリ

PC、スマホ、タブレットなどさまざまなデバイスが登場してきました。メディアクエリとはデバイスに応じてCSSを切り替える仕組みです。最近ではメディアクエリも自動記述する機会も増えていますが、その理解がどのようになっているか理解することは大切です。

以前はPC用のHTML。スマホ用のHTMLと別々に実装することも珍しくありませんでした。それぞれに別のページを作るのは面倒でコストがかかります。1つのページをさまざまなデバイスで使いまわすようになれば便利です。そのような背景を受けて、CSSにメディアクエリという仕組みが追加されました。画面の幅に応じて適用するスタイル（フォントサイズや表示する内容）を変化させることができます。以下のサンプルをご覧ください。

```
<code>index.html</code>
<DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>CSS Media Query</title>
  <style>
    hl { color: red; }
    p.overview { font-size: 12px; }
    p.detail { visibility: hidden; }

    @media (min-width: 576px) {
      hl { color: orange; }
      p.overview { font-size: 16px; }
      p.detail { visibility: hidden; }
    }

    /* Medium devices (tablets, 768px and up) */
    @media (min-width: 768px) {
      hl { color: purple; }
      p.overview { font-size: 18px; }
      p.detail { visibility: visible; }
    }
  </style>
</head>
```

```
/* Large devices (desktops, 992px and up) */
@media (min-width: 992px) {
  hl { color: green; }
  p.overview { font-size: 24px; }
  p.detail { visibility: visible; }
}

/* Extra large devices (large desktops, 1200px and up) */
@media (min-width: 1200px) {
  hl { color: blue; }
  p.overview { font-size: 30px; }
  p.detail { visibility: visible; }
}
</style>
</head>
<body>
  <div>メディアクエリ サンプル</div>
  <div class="overview">
    <p>メディアクエリは画面の解像度といった条件に対して、コンテンツの振舞いを変えるようにするCSS3のモジュールである。</p>
  </div>
  <div class="detail">
    <p>メディアクエリはHTML文書の拡張で、<code>@media</code>宣言の中で<code>media</code>属性を使って宣言できる。</p>
    <p>どのデバイスでリンクされた文章が表示されるかを<code>media</code>属性の値で指定される。</p>
    <p>またメディアクエリはCSS3標準仕様の<code>@import at-rule</code>や<code>@media at-rule</code>で定義することもできる。</p>
    <p>CSS3ではこれらがメディアクエリとして定義されている。</p>
  </div>
</body>
</html>
```

メディアクエリはスタイルの中で以下のように記述します。

```
@media (min-width: 1200px) {
  スタイルの指定
}
```

これは、ウィンドウのサイズが1200pxよりも大きい時に内部のスタイルが適用されることを意味

第10章 MongoDB の基礎 レッスン

10.4 レッスン

●課題1

「楽天ウェブサービス」では以下のようにWeb APIを抽出するページを公開しています。
<https://web.archive.org/web/20160701000000/http://api.rakuten.co.jp/explorer/api/>
この中から楽天ブックス系API→楽天ブックス検索一覧APIを選び、keywordにpythonを指定してGETボタンをクリックしてください。結果となるJSONがページに表示されます。その内容をテキストファイルにコピーして、MongoDBに取り込んでください。



結果をbook-data.jsonと保存した場合、以下のようなプログラムでファイルを読み込み、jsonデータをロードすることができます。

```
import json
books = []
with open("book-data.json", "r", encoding="utf-8") as f:
    data = json.load(f)
    for book in data["items"]:
        books.append({
            "itemName": book["item"]["itemName"],
            "itemCode": book["item"]["itemCode"],
            "itemPrice": book["item"]["itemPrice"],
            "reviewCount": book["item"]["reviewCount"],
        })
print(books)
```

このサンプルを参考に、データベース名を「setapp」、コレクション名を「books」として書籍のデータを格納してください。



●課題2

課題1で検索した書籍から価格2000円より高い書籍を抽出し、そのタイトルと価格を出力してください。

●課題3

booksコレクションの中からreviewCountが4のものを選択してください。

●課題4

booksコレクションの中から価格2000円より高い書籍を削除し、課題2で作成したプログラムで検索結果が4になることを確認してください。

<<目次>>

第1章 インターネットプロトコルの基礎

第2章 HTTPの基礎

第3章 HTMLとCSSの基礎

第4章 jQueryの基礎

第5章 Web-APIの基礎

第6章 Bootstrapの基礎

第7章 Flaskの基礎

第8章 Cookieとセッション・REST

第9章 Ajax(Asynchronous JavaScript + XML)

第10章 MongoDBの基礎

第11章 フレームワークの基礎

第12章 サンプルアプリ作成で確かめるWeb技術の変遷

<<著者紹介>>

田中 賢一郎

慶應義塾大学理工学部修了。キャノン株式会社に入社し、デジタル放送局の起ち上げに従事。その間に単独でデータ放送ブラウザを実装し、マイクロソフト(U.S.)へソースライセンスし、Media Center TVチームの開発者としてマイクロソフトへ。MSではWindows、Xbox、Office 365などの開発・マネージ・サポートに携わる。2016年に中小企業診断士登録後、セカンドキャリアはIT教育に携わると決め、IT系の専門学校で1年間現場経験を積んだ後、2017年春にFuture Codersを設立。「プログラミング教育を通して一人ひとりの可能性をひろげる」という理念のもと、英語と数学に重点をおいた実践的なプログラミングスクールの運営を目指す。趣味はジャズピアノ演奏。

<<販売ストア>>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple Books、紀伊國屋書店 Kinoppy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

※ 各ストアでの販売は準備が整いしだい開始されます。

※ 全国の一般書店からもご注文いただけます。

【インプレス R&D】 <https://nextpublishing.jp/>

株式会社インプレス R&D（本社：東京都千代田区、代表取締役社長：井芹昌信）は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishing を使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知識の流通を目指しています。

【インプレスグループ】 <https://www.impressholdings.com/>

株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:唐島夏生、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「旅・鉄道」「学術・理工学」を主要テーマに専門性の高いメディア&サービスおよびソリューション事業を展開しています。さらに、コンテンツビジネスのプラットフォーム開発・運営も手がけています。

【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

TEL 03-6837-4820

電子メール: np-info@impress.co.jp