

Python 教材の応用について  
——JavaScript への書き直し——

杉田 一樹

## 目 次

<b>1</b>	<b>はじめに</b>	<b>1</b>
1	テーマ選択の理由 . . . . .	1
2	JavaScript . . . . .	2
3	対象者 . . . . .	3
4	本教材の利用価値 . . . . .	4
5	過去の教材との比較 . . . . .	4
<b>2</b>	<b>本教材について</b>	<b>6</b>
1	制作する際の注意点 . . . . .	6
2	本教材の構造 . . . . .	7
3	必要な技術 . . . . .	8
4	活用した資料 . . . . .	9
5	制作の過程と見通し . . . . .	10
<b>3</b>	<b>教材の詳細</b>	<b>11</b>
1	ファイル構成 . . . . .	11
2	本教材のデザインについて . . . . .	12
3	マニュアルの構成 . . . . .	12
4	工夫した点 . . . . .	15
<b>4</b>	<b>評価</b>	<b>16</b>
1	アンケート . . . . .	16
2	自己評価 . . . . .	20
<b>5</b>	<b>まとめ</b>	<b>21</b>
1	今後の課題 . . . . .	21
2	終わりに . . . . .	22

## 1 はじめに

本教材は、福田ゼミ必修授業である「プログラミング演習」で使用する、Python 教材を JavaScript 用教材に書き換え、オンラインマニュアル化したものである。プログラミング演習で Python<sup>(1)</sup>を用い実行したことを JavaScript でも実行できるように説明している。Python 教材と同様の問題も用意しており、問題の答えは別ファイルに記載してある。福田ゼミ生には是非、プログラミング演習受講をし、休暇期間に復習として本教材に取り組んでほしいと思う。

### (1) テーマ選択の理由

このテーマを選択した理由は、福田ゼミの基本テーマである「人の役に立つものを作る」テーマに従い、「プログラミング演習」という授業で使用される Python 教材の JavaScript 版があれば便利なのではないかという考えから、このテーマへの取り組みを始めた。

福田ゼミでは Python というプログラミング言語を中心に学ぶが、JavaScript というプログラミング言語についてはあまり触れられていない。JavaScript はとてもポピュラーなプログラミング言語であり、使用頻度が多く、HTML<sup>(2)</sup>にも多く組み込まれている。福田ゼミ生はプログラミング演習で Python を学ぶ。休暇期間等に本教材を使い、Python 教材の内容を JavaScript でも実現できるということを知ってほしい。そうすることで、プログラミング演習の復習にもなると考える。また、福田ゼミの基本テーマである「人の役に立つものを作る」というのは決しての大勢の人を対象に作るのに意味があるわけではない。大勢の人を対象にする教材は需要があるので、誰かが作ってくれる。しかしターゲットを絞った教材は誰も作ってはくれないので、誰かが制作

する必要があるのだ。本教材を利用することによって、福田ゼミ生はプログラミングの知識をさらに蓄えていってもらいたいと考えている。

本教材はプログラミング演習という授業で使われている Python 教材 (プリント) を JavaScript 教材 (オンラインマニュアル) に書き直したものである。よって、教材内でプログラムを実行していることは同じである。これまで JavaScript のマニュアルとして、ゼミの先輩が作られた教材があるが、本マニュアルでは、全く別のことを説明している。

## (2) JavaScript

### (i) JavaScript とは

JavaScript とは静的な表現しかできなかった Web ページに、動的な表現を可能にする目的で開発されたプログラミング言語である。

Web ページを作成する際に HTML という言語が使われる。HTML は文章や絵を効果的にシンプルに配置することが出来る。しかし HTML のみでは動的な表現をするのができないため、何の動きもない Web ページになってしまう。そこで、HTML 文書に JavaScript を組み込むことにより、動的なページの表現が可能となり、Web ページに様々な機能を追加することが出来る。

JavaScript 普及時には、Web ブラウザ<sup>(3)</sup>との互換性が乏しく問題になったが、現在ではほとんどの Web ブラウザに標準搭載されているため、新たなソフトを求める必要はない。我々にとって最も身近なプログラミング言語のひとつといえる。また、JavaScript と Java は名前が似ていて混同されることが多いが、全く別のプログラミング言語であるので注意していただきたい。

## (ii) JavaScript の問題点

前項で述べたとおり、数年前までは互換性の問題があった。その問題は近年解消されたが、セキュリティの問題は JavaScript 普及時から問題になっている。

JavaScript は悪質なプログラムも比較的、簡単に作れてしまう。Web ページを開いたら無尽蔵にウィンドウが増えていき、パソコンの動作が不安定になった経験はないだろうか。また、ActiveX コントロール<sup>(4)</sup> を悪用し Windows システムを書き換えたりする悪質なサイトが急増した。HTML メールを返信すると第三者に盗み見られるという脆弱性も発見されている。

JavaScript が標準搭載されている Web ブラウザでは、JavaScript を有効にさせるか無効にさせるか設定できるものが多い。「見知らぬサイトを訪れる際は、JavaScript を無効にするのが常識」と一部では言われている。JavaScript はセキュリティの面でまだまだ課題の残されている言語だといえる。

## (3) 対象者

福田ゼミ生のみを対象者とする。パソコンを触ったことがない人や、プログラミングについて全く知らない人には向かない教材である。プログラミング初心者には説明することが多すぎて、本教材内でも説明しきれていない。プログラミングの事を一通り勉強している福田ゼミ生だからこそ有効に取り組んでいただける教材となっている。

#### (4) 本教材の利用価値

本教材内の「実行してみる」のリンクをクリックすることにより、JavaScript の実際の動きを見てもらうことができ、なぜこのような動きをしているかなど、基本的な動作の仕組みを理解していただける。また福田ゼミ生は、プログラミング演習で使用する、Python 教材の復習教材として夏休みなどに学習することにより、Python との特徴の違いなどを学習することができる。

プログラミングの復習として、メニューに演習 1 から演習 8 の他に「JavaScript の基本」というページを記載してある。このページで基本的なことを復習していただくことにより、演習 1 から演習 8 に取り組んでいただき、問題を解いてもらうことで技術取得の意欲が増す。そうして、プログラミング全般の知識が増していくと考える。

#### (5) 過去の教材との比較

JavaScript 教材として、過去のゼミ生が作成した JavaScript 教材がいくつか存在する。しかし、その教材にはいくつかの問題点が挙げられる。

- ・ 個人の趣味が全面に出すぎていて、デザインに問題がある。
- ・ 例文に画像が使用されていて、実行例が見れない上、コピー&ペーストが使用できないページがある。
- ・ 各ページに問題が用意されていないため、しっかりした学習ができない教材がある。
- ・ 誤字脱字が多い。
- ・ プログラムの詳しい解説がない。
- ・ 正常に動作しないプログラムがある。

以上のような問題点が挙げられる。先輩が作った JavaScript 教材との大きな違いは、プログラミング演習で使用する Python 教材の内容を、できる限り JavaScript でも実現できるように解説してある点である。これが本論文のサブタイトルの意味となっている。また、ウェブ上でも高機能な JavaScript マニュアルサイトは存在する。例として2つのサイトを挙げる。『JavaScript 例文辞典 (<http://www.openspc2.org/reibun/javascript/>)』は JavaScript のサンプルが詳しく掲載されているサイトである。しかし、プログラムの詳しい説明はなされていないし、各項には実行例は用意されているが、問題は用意されていない。また、文字化けが多く見られた。『JavaScript 入門 & サンプル集 (<http://plusone.jpn.org/javascript/>)』では、説明不足と感じられる点が所々にあった。基本事項である演算子の説明が十分になされていなかったり、コメントアウト方法の記述はあるが、それについて詳しく説明されていなかったりしている。こちらのサイトでも問題は用意されておらず、理解を深めることは厳しいだろう。本教材では、テキストボックス内にプログラムを記述し、テキストボックス内でもプログラムの解説を加えている箇所がある。また、問題も用意しており、理解が深められる教材となっている。

本教材のデザインについては前述している。例文、例題にはテキストボックスを使用しコピー&ペーストが可能である。コピー&ペーストを利用し、自分なりにプログラムを書き換えることも可能だし、例題の実行例は必ず用意してある。各ページの最後には理解度のチェックも含めて、問題を用意してある。問題の解答は別ファイルに保存してあるので、一般利用者は閲覧できない。誤字脱字や、プログラムの動作は何度もチェックを繰り返し、正確に動作している。プログラムの詳しい解説

は、テキストボックス内で行ごとに解説をしているページがある。詳しい解説を導入することにより、ここではどういった指示をしているのかなどが明確にわかり、プログラミング知識の向上につながる。

## 2 本教材について

### (1) 制作する際の注意点

教材を制作するに際して、いくつかの注意しながら制作をした。まず、教材のデザインはシンプルで見やすいものにしようと考えた。なぜならば、デザインというのは人の意欲を駆り立てると考える。複雑で見にくいデザインであれば、それだけで教材を使うのが嫌になり、意欲が低いまま勉強をしてしまうということになる。最悪の場合、教材を使わないということにもなりかねない。

そこで、教材内では比較的薄めの色を使い、同系色でページを構成している。背景色も説明文などの邪魔にならないように設定した。また、重要なポイントには赤で色分けをしている。実行例の中も、同系色を使用して、文字サイズも比較的大きめに設定し、見やすいデザインというのを心がけた。

もう一点は、使用者に教材内容を理解してもらえようようにすることである。“教材”として制作しているのだから、教材としての機能を果たすことは最重要項目である。そのために、まず説明文を徹底的に見直した。自分で文章を組み立て、誤字脱字がないかを確認し、理解できる文章か何度も読み直した。こうすることにより、文章を簡略化しすぎることなく、対象者の方には十分に理解していただける説明ができると考えている。

## (2) 本教材の構造

本教材はオンラインマニュアルとして制作した。ページ構成は、演習 1 から演習 8 のページと、JavaScript の基本、トップページの 10 ページとなっている。演習 1 から演習 8 のページは、Python 教材に沿った内容が記載されている本教材のメインページである。JavaScript の基本のページは、演習 1 から演習 8 を取り組むにあたり基本的なことを記載してある。

各ページでは解説の実践例をテキストボックス内で表示して、説明文との差別化を図っている。また、演習 1 から演習 8 のページではテキストボックスの下部に「実行してみる」のリンクを貼り、テキストボックス内のプログラムを実際に動作させ、利用者に理解してもらうようにしている。そうして、問題に取り組んでもらい、問題が解けた時点で次の演習ページに行ってもらおうようにしている。問題に取り組む際、ユーザーにはテキストエディタ<sup>(5)</sup>を用意していただく。

各演習ページの問題の答えだが、福田ゼミの 3 回生に実際に本教材を使用してもらいアンケートを取り、「問題の答えが欲しい」という意見が多数あった。しかし、簡単に答えが見れてしまう教材を作ったら、理解しないまま答えを見てしまう可能性が高いと考えた。はじめは、HTML ファイル内の最下部にコメントアウトで記載して HTML ソースで見ってもらうという、わかりづらい載せ方をしようと考えたが、解答を載せたら、知識が身に付かないし絶対に解答を見てしまうと見え、教材内には解答は記載していない。今回は別ファイルで保存してある。こうすることにより、各演習の理解度を深めてもらう。

ページ左のメニューだが、どのページでも左にメニューを表示していることにより、ページ間の移動の手間を省いている。演習 1 のページを

取り組んでいて、基本のページに戻りたければすぐに戻ることも可能である。しかし、ページをスクロールしてもメニューは一緒にスクロールされるわけではないので、ページの一番上までスクロールしてメニューに行くしかない。ここは使い勝手が悪いので、改善する必要があると考えている。

### (3) 必要な技術

本教材はオンラインのマニュアルであるため、HTML の知識は必要不可欠である。また、JavaScript の教材であるため、JavaScript の知識も必要不可欠である。デザインを見やすく整えているため、CSS の技術も用いている。

卒業論文に取りかかる前は、自身の JavaScript に対する知識が不十分であったため、資料として『独習 JavaScript』を用いて勉強した。また、Web サイトの『とほほの JavaScript リファレンス』も使用した。

単に Word 文書でマニュアルを作成することも可能で、そうすれば必要な技術も Word の技術のみで難しい知識は不必要となるが、本教材にも導入している、「実行してみる」のボタンで実際の動作を見てもらうことができなかつたり、オンラインのマニュアルに比べると、欠点が多い。

オンラインマニュアルにすることにより、ブラウザ上のみで動作し、ブラウザ上で、問題などを処理することにより Word 文書と比べて知識の取得がしやすくなる。また、個々のコンピュータ環境に左右されることが少ない。

#### (4) 活用した資料

本教材を制作するに際し、活用した資料がある。

##### (i) 『独習 JavaScript』

この書籍は、自身の JavaScript の学習のためにも使用したし、本教材内の問題を作る際にも使用した書籍である。書籍のため、プログラムの動作例がわかりづらかったりするが、細部にまで解説が入れてあり、初心者が JavaScript を習得するのに必要な基本的な文法や、関数から Ajax<sup>(6)</sup>、Prototype.js<sup>(7)</sup> の内容まで載っている。また、各章でのサンプルコードが豊富である。情報量が多いため、この 1 冊で JavaScript の技術は十分に習得できる。

##### (ii) 『とほほの JavaScript リファレンス』

このサイトは、『独習 JavaScript』と同様に、基本的な知識が多く記載されているサイトである<sup>(8)</sup>。『独習 JavaScript』よりも詳しく説明している部分があったりするので活用させてもらった。また、『独習 JavaScript』ではなかなか目的のページが見つからなかったりしたが、オンラインの特性を活かし、リンクからすぐに飛べるようにしているので、目的のページがすぐに見つかった。また、自分でテキストエディタを用意したら、コピー&ペーストで記載されているプログラムを実行させることもでき、オンラインの特性十分に活かしたサイトになっている。

##### (iii) 『イヌでもわかる JavaScript 講座』

このサイトには各ページにプログラム実行例が表示されていて、非常にわかりやすかった<sup>(9)</sup>。また、とほほの JavaScript リファレンス と違い、トップページの目次が「〇〇してみましょう」と書いてあり、目的がはっきりしていたらすぐに目的のページに飛べるので素早く目的の

ページに飛ぶことができたし、非常に使いやすかった。具体的にプログラムを動かす、その楽しさを体感してもらうような作りになっているサイトである。

#### (iv) 『CSS レイアウト実践講座』

本教材のデザインを編集する際に、活用した Web サイトである<sup>(10)</sup>。自身の CSS に対する知識が不十分であったため、活用させていただいた。CSS のマニュアルサイトは他にもあり、いくつか見てみたがここが一番使いやすかった。ページデザインが見やすく、目的のページがすぐ見つかる。各プロパティの解説が載っており、ブラウザ上の表示例も載っていたので使いやすく、わかりやすいサイトだった。

#### (5) 制作の過程と見通し

本教材では問題を用意している。その問題は『独習 JavaScript』や『とほほの JavaScript リファレンス』を参考にして作っているため、自身も教材に載っている問題を解く必要があった。

私自身、HTML の知識は備えていたが、JavaScript の知識は十分に備えていなかった。知識を十分に身につけ JavaScript の仕組みを理解しないといけないと考え、先ほど挙げた書籍や Web ページを使用して学習をした。

JavaScript の歴史や、開発環境についても『独習 JavaScript』で学習した。また書籍に記載されている問題を基礎技術と並行して取り組むことで、知識習得を効率良くした。こうして勉強していくうちに、JavaScript の問題点や現状が見えてきたり、他のプログラミング言語との違いも理解できるようになっていった。

学習を終えてから、教材の見通しを立てていった。構造については、

前述してある。また、見やすいデザインというのを心がけており、CSS を使用している。あとは、実際にプログラムの動作を見てもらったりと、オンラインの特徴を活かした教材にし、他の教材との差別化を図るようにした。また、各演習ページの説明文は詳しく説明しすぎず、簡潔に説明しすぎずというのを意識している。詳しくすぎたら、自然と説明文も長文化してしまい、読む意欲をなくしてしまう危険がある。簡潔にしすぎたら、理解するのが難しくなる。

こうした下準備や、見通しを立てて本教材の製作に取り掛かることにした。

### 3 教材の詳細

本教材は、福田ゼミ生のみを対象に作った教材である。また、教材内容はプログラミング演習の Python 教材を JavaScript でも実行できるようにするために作っている。そのため、福田ゼミ生にはプログラミング演習の復習教材として本教材に取り組んでもらいたい。この教材に取り組んでもらう前提として、HTML の知識が必要になってくる。

#### (1) ファイル構成

ファイルの構成は、トップページ (index.html)、JavaScript の基本 (kihon.html)、演習 1 から演習 8 のページ (enshu1.html から enschu8.html)、各演習ページの実行例 (enshu ○-○.html) という構成になっている。あとは、各演習ページの解答 (○-○.html) と、教材のデザインに CSS (style.css) を使っている。本教材のメインである JavaScript だが、今回は JavaScript を外部ファイルとして呼び出すのではなく、HTML ファイルに埋め込んでいるため.js ファイルはない。

## (2) 本教材のデザインについて

本教材のデザインには CSS を使用している。CSS とは Cascading Style Sheets の頭文字を取った略語であり、Web サイトのデザインに用いられる技術である。CSS を使用すると、関連づけられている HTML 文書すべてが CSS の指示通りのデザインとなる。CSS を使用しなければ、1つ1つの HTML 文書を書き換えなければならないが、CSS を用いることで、デザイン変更の際に CSS ファイルに手を加えると、すべての HTML 文書のデザインを変更してくれる。CSS を用いることにより Web ページの統一感も出すことができる。このような利点から、ユーザーの教材の使いやすさ、自身の編集作業のためにも CSS を使うことにした。

## (3) マニュアルの構成

### (i) トップページ

トップページ (図1 参照) では本教材の簡単な説明、注意事項と本教材を使うためのソフトとブラウザの設定を指定している。

### (ii) JavaScript の基本

JavaScript の基本のページ (図2 参照) に関しては、プログラミング演習で使った Python 教材と内容は異なっている。これから JavaScript を入門するための基本中の基本となるページである。

- ・ HTML 文章内での JavaScript 記述方法について
- ・ <script>タグについて
- ・ コメントアウトの方法について
- ・ 関数について
- ・ 文字列や変数、関数の宣言方法などの基本事項について

HTML 文章内での JavaScript 記述方法には 4 種類紹介している。ここでは方法の紹介だけで、具体的な説明はしていない。なお、本教材で推奨する方法は、HTML 文書内に JavaScript を組み込む方法である。

また、コメントアウト方法では 1 行だけコメントアウト、複数行コメントアウト方法を載せてある。HTML 文書内でコメントアウトし、作業工程をコメントアウトでメモしていく作業というのはプログラミングを行う際には欠かせない。

基本事項には、プログラムを作る際に必ず使用する事項が載せてある。つまりこのページではこれから JavaScript を扱う際に絶対に必要となる知識が記載されている。

なお、このページには説明文、例文のみで問題はない。

### (iii) 演習 1

演習 1 のページ (図 3 参照) では「電卓としての利用」と題して、数値の計算方法や変数の宣言方法、変数を使用しての計算方法、応用として文字列と数値を組み合わせた表示方法、算術演算子の一覧を載せている。テキストボックス内には実行例文の横に一行ずつ解説を入れている箇所もある。このページからは「実行してみる」のリンクを貼っており、テキストボックス内に表示してある例文の実行画面を見ることができ、このページを理解し、問題に取り組むことにより、演習 2 のページに取り組みやすくなっている。

### (iv) 演習 2

演習 2 のページ (図 4 参照) では文字列と数値の違い、変数への代入方法、関数の呼び出し方法を記載している。演習 1 と同様、基礎の基礎を学習するページである。” document.write ” の時、変数と文字列をあ

わせる場合、文字列を「”」で囲み、「,」で区切れば良かったが、” alert” の場合は、「+」で区切らないといけないであったり、間違いやすい箇所はテキストボックス内に解説をいれてある。

### (v) 演習 3

演習 3 のページ (図 5 参照) では JavaScript におけるリストの使い方について記載している。一般的なリストの作り方から、リストの中のリストの作り方、リストの項目の取り出し方、範囲を指定してのリストの取り出し方など、リスト全般について説明しているページである。また、リストの項目を表示する際の注意点、<br>タグの使用方法、” .slice” の使い方も記載している。演習 1、2 を理解できていたら、演習 3 も十分に理解できるページになっている。

### (vi) 演習 4 ・ 演習 5

演習 4 ・ 演習 5 のページ ((図 6 ・ 図 7 参照)) ではプログラミング言語 (JavaScript 以外も) を使用する際に不可欠となる「制御文」について記載している。演習 4 と演習 5 の 2 ページ構成にした理由は、制御文すべてに対して 1 ページだと、1 ページに記載する内容が多すぎてしまうことや、制御文の内容を分割することにより、制御文の内容が頭に入りやすいということを考え、2 ページ構成にしている。「for 文」、「while 文」、「if / else 文」、「else if 文」についての説明ページである。

### (vii) 演習 6

演習 6 のページ (図 8 参照) では関数について記載している。関数の利点や、関数の定義方法について説明している。例題で、体積を求めるプログラムを記述してあるが、体積を求める際には円周率が必要となる。「Math.PI」で円周率が表示されるのだが、これは自身も知らなかった方法であった。テキストボックス内に「Math.PI」について解説を入

れている。

#### (viii) 演習 7・演習 8

演習 7・演習 8 のページ (図 9・図 10 参照) は、他のページとは違い比較的難しいことを説明している。九九の表の作り方と、タブ区切りのデータ (エクセルなどで作ったデータ) から HTML のテーブルに表示するプログラム (図 11 参照) の解説である。テキストボックス内のプログラムを理解してほしいと考え、実行例だけを用意し問題は用意していない。

#### (4) 工夫した点

デザインについて。「制作する際の注意点」の項でも説明しているが、心がけたのが、シンプルで見やすいデザインにすることである。比較的薄めの色を多く使い、同系色でまとめている。フォントサイズも大きめにし、ユーザーのモチベーションを下げないようなデザインとなっている。ページの上部には項目名が書いているので、現在どのページに取り組んでいるのかがわかりやすい。重要な部分は赤で色分けしている。後述するが、福田ゼミ 3 回生にやってもらったベータテストの結果でも本教材のデザインは好評価を得ている。メニューだが、どのページでも常に左部分に表示してある。これでどのページにも簡単に移動することが可能である。

各ページの問題についてだが、解答は掲載していない。これは一見、ユーザーにとって不親切にも見えるが、私なりに考えて解答は掲載しないということに決めている。解答を掲載してしまうと、ユーザーは理解していないうちに解答を見てしまい、理解ができていないにも関わらず、プログラムが正常に作動し、「理解できた」と誤解を起こしてしま

う。解答を載せないと答え合わせができないので、ギリギリまで載せようか迷っていたが、載せないことに決めた。その代わりに、解答は別ファイルとして保存してある。

各ページのテキストボックス内にはプログラムが記載されているが、一行ごとに解説を入れている箇所がある。(図 12 参照) このように 1 行ずつ解説を入れることにより、ユーザーの理解度を深めるようにしている。実行してみるのリンクをクリックすることにより、実行画面を見ることが可能である。(図 13 参照)

## 4 評価

### (1) アンケート

本教材が一通り完成した時点で、福田ゼミの 3 回生に 20 分程度、本教材のモニターとして取り組んでもらい、終了後にアンケートを取った。アンケートの内容は以下である。

1. デザインはどうだったか (5 段階評価)
2. 説明文は理解しやすかったか (5 段階評価)
3. メニューの使い勝手はどうだったか (5 段階評価)
4. 使いにくかった点
5. 改善すべき点
6. 欲しい機能など

回答者は 10 名。1. 2. 3 の質問には 5 段階評価で回答していただき、4. 5. 6 の質問には具体的に書いていただいた。

1 の質問だが、5 段階評価の平均点は 3.8 点という結果になった。私自身、デザインに関してはそれほど心配はしていなく、満足した出来に仕上がっていたのでこの評価は意外だったと共に、考えさせられる結果

となった。「文字が大きく、背景の色も邪魔をしていない」「色がカラフルで見やすかった」「重要なポイントは色分けしてある」とマイナスな意見はなかった。しかしなぜ点数が伸び悩んだのか。その考察については後述したいと思う。

2の質問の平均点は3.4点になった。説明文にも力を入れてこの結果だったので「思うようにはいかないな」と思った。この質問には「4」の評価をしてくれた人もいれば、「1」という厳しい評価をしてくれた人もいた。逆に間の「3」の評価を付けた人が非常に少なかった。

3の質問の平均点は3.4点だった。2の質問と同様の結果となった。しかし、回答の内訳は2とは異なり、「1」や「5」のハッキリした評価をしてくれた人はおらず、「2」「3」「4」だけの結果となった。

4の質問の意見としては以下の通りである。同様の意見は省略してある。

- ・一つ一つの実行例があって分かりやすい。
- ・説明文や、例がわかりやすい。
- ・すぐに実行ができ、結果がわかる点。
- ・特になし

5の質問の意見としては以下の通りである。同様の意見は省略してある。

- ・説明不足と感じる部分が所々にあった。
- ・メニューの部分に「演習○」としか書かれてないのでわかりにくい。  
それぞれの単元名にした方が探している項目を見つけやすい。
- ・メニューが上にいってしまうところ。

6の質問の意見としては以下の通りである。同様の意見は省略してある。

- textarea 内の注釈は「←」より「//」のほうがよい。
- 実行画面もみやすくすべき。
- メニューの部分と本文の部分の隙間をもう少し空けるべき。
- 演習の中の項目にもリンクで飛べるようにしたほうがよい。
- <div id="main">の中にある文章は padding を設定した方がよい。

7の質問の意見としては以下の通りである。同様の意見は省略してある。

- 問題の解答が欲しい。
- ブログのようにタイトルをクリックしたら TOP に戻れるようにして欲しい。

この項目については、ほぼ全員が「解答が欲しい」と回答してくれた。

#### (i) アンケートの分析

今回のアンケートの結果を受けて、自分なりに分析を試みた。

1の質問は前述してある通り、意見はそれほど悪くはないのに、思ったより点数が伸び悩んだ。これは細かいところのデザインが足りなかったのだと考える。例えば、本教材ではフッターがない。今回デザインを考えている中で、フッターは必要ないと考えていた。フッターを導入してもフッターに記述する内容がないし、他のサイトを見ても Copyright<sup>(11)</sup> が書かれているだけだ。このように考えていたが、デザインの見栄えを良くするにはフッターを導入すべきだったのかもしれない。また、メニューの位置だが、各ページにメニューを表示して、いつ、どのページにもアクセスできるようにしてあるが、各ページで下にスクロールしてしまうと、わざわざ上にスクロールしてメニューに戻らなければならない。こういった不便さも点数が伸び悩んだ理由ではない

かと考える。

2の質問も前述している通り、内訳が極端な結果になった。この結果から、プログラミングをある程度経験したことのある人には、わかりやすい教材になっているがHTMLの知識だけで、プログラミングもあまりやったことがない人には分かりにくい教材になったのではないかと考える。例えば、演習5で出てきている「++」「+=」の意味が記述されていない。こういった説明不足がこの結果につながってしまったのではないかと考える。

3の質問だが、ここの内訳は「1」「5」の評価をつけた人はいなかった。この結果から特に問題ないと一見思ってしまうが、修正すべき点は存在した。それは先ほども挙げた、ページを下にスクロールすると、わざわざ上にまで戻らないとメニューがないという点である。

4の質問は、ほとんどの人が「実行例があって分かりやすい」と書いてくれた。この結果には非常に満足しているが、それ以外の意見が非常に少なかった。使いやすかった点が1点というのは非常に問題があると思うので、もっと深く分析し利点をもっと多くなるような教材にしたい。

5の質問だが、この質問でもメニューのスクロール問題を挙げた人がいた。また、説明不足の点として「例文を載せるだけでなく、きちんと解説してくれないと応用するにしてもどうしていいかわからない」という意見があった。こういった意見を聞いて、初めて自分で気づく点があった。

6の質問は「演習の中の項目にもリンクで飛べるようにしたほうがよい。」という意見がいくつかあった。しかし、1ページがあまり長くないし、1単元も短いのでページ内リンクは作成しなかった。また、テキ

ストボックス内の注釈は指摘を受けたとおり、「←」から「」に改善した。これはテキストエディタにそのままコピーして使えるようにするためである。他にも参考になる意見をいただいたが、指摘をされている分、教材制作の詰めが甘かったということだと考える。指摘されたところは改善するとともに、自身の反省点にもなった。

7の質問は、欲しい機能として、ほぼ全員が問題の解答を欲していた。「本教材の構造」の項でも述べたが、解答を載せてしまうと問題に取り組む前に見てしまう恐れがある。そうなってしまうと意味がないので解答は載せないことにした。解答は別ファイルとして保存してある。

## (ii) アンケートを受けての改良点

アンケート終了後の改良点は以下の通りである。

- ・ テキストボックス内の注釈の変更。
- ・ 誤字・脱字の訂正。

誤字、脱字はアンケートで何箇所か指摘されていた点である。

また、今後改良する点としては以下の通りである。

- ・ フッターの導入。
- ・ 説明文の改善（「++」「+=」の意味修正など）
- ・ メニューの改良（スクロールの問題、単元名追加）
- ・ ブログのようにタイトルをクリックしたら TOP に戻れるようにする。

## (2) 自己評価

アンケートに協力してもらい、自身でも教材を見直したことにより改善すべき点が多く見つかった。改善すべき点が多かったということは、使いにくかった点が多かったということである。一度、完成してからの

改善点の発見だったので詰めが甘かったといえる。説明文も自分では簡潔に丁寧に書いたつもりであったが、用語の解説などが抜けており、わかりづらい教材となってしまった。

「実行する」ボタンにより、例題の実行例が実際に体験できるのは良かったところであり、オンラインマニュアルの特性を十分に生かされたと考える。また、本教材のデザインも「フッターがあればよい」という指摘はあったが、自分で合格点を出せるデキにはできたと考える。ゼミ生に教材に取り組んでもらった時も、みんな真剣に取り組んでくれて、アンケートを書ってくれたしデザインで教材に取り組む意欲を損なわなかったのは大きな点だと考える。しかし、改良する余地はまだあるのでもっと使いやすく、意欲を駆り立てる教材にしていきたいと思う。

## 5 まとめ

### (1) 今後の課題

今後の課題としては、「アンケートを受けての改良点」項で挙げた、今後改良する点を修正することが第一の課題となる。また、よりよい教材にするためにはオンラインマニュアルの特性をもっと活かした教材にすると、使いやすく、わかりやすい教材になると考える。例えば、現在ではテキストエディタを各自用意してもらい、問題に取り組んでもらっているが、教材内に空のテキストボックスを作り、ボタンをクリックすると新規ウィンドウで実行結果が表示されるようにする。こうすることにより、テキストエディタを用意する必要がなく、簡単に実行結果を閲覧することが出来る。

また、Python 教材を基に JavaScript 教材を作成したので、数値の計算方法や、文字列の連結方法など、具体的なサンプルを用いて比較でき

るようなコンテンツがあれば、福田ゼミ生にとってより良い教材になると考える。

## (2) 終わりに

今回、本教材の制作に取り組み、決して一流の教材とは言えないが、福田ゼミの基本テーマである「人の役に立つ Web アプリケーションを作る」は達成できたと思う。しかし、その制作過程は決して順調といえるものではなく、何度も壁にぶち当たった。試行錯誤を繰り返し、時には友人の手を借りたりと。教材を作ることの大変さ、教材を使用する側の視点に立ち、いかに使いやすくするか考えることの大変さを思い知った。本教材を使い、JavaScript がどういった言語なのか、またプログラミングの楽しさというのを感じていただきたいと思う。私自身が Python を習っていたが、JavaScript も学んだことで、両言語の違いなどが理解でき、プログラミングの楽しさ、奥深さを深く感じる事ができた。

本教材を使用し、プログラミングの重要性を見出してもらい、JavaScript だけではなく、様々なプログラミング言語に興味を持っていただけるのではないだろうか。また、本教材をきっかけにプログラミングを本格的に学びたいという人が出てくれば幸いである。

**注**

- (1) Python = Guido van Rossum 氏が開発した、オブジェクト指向のプログラミング言語
- (2) HTML = Hyper Text Markup Language : Web ページを記述する際に使用するマークアップ言語
- (3) Web ブラウザ = Web ページを閲覧するためのソフトウェア。
- (4) ActiveX コントロール = Microsoft 社が開発したソフトウェア部品化技術。サイト側がパソコンにプログラムをインストールするためのもの。
- (5) テキストエディタ = コンピュータで文字のみのファイル (テキストファイル) を作成するためのソフトウェアの総称。本教材では J-editX を使用。
- (6) Ajax = Asynchronous JavaScript and XML JavaScript と XML の技術を使って非同期で通信を行う開発手法のこと。Google Maps などに使用されている。
- (7) Prototype.js = もっとも有名な JavaScript ライブラリの 1 つ。
- (8) <http://www.tohoho-web.com/js/index.htm>
- (9) <http://www.red.oit-net.jp/tatsuya/java/>
- (10) <http://css.uka-p.com/>
- (11) Copyright = 著作権。Web ページにも著作権があり、ページ下部に書かれていることが多い。

## 文献表

高橋和也 / 竹添直樹 / 里見知宏

2008 『独習 JavaScript』 翔泳社

『とほほの JavaScript リファレンス』

<http://www.tohohoweb.com/js/index.htm>

『イヌでもわかる JavaScript 講座』

<http://www.red.oit-net.jp/tatsuya/java/>

『CSS レイアウト実践講座』

<http://css.uka-p.com/>