

# 見えない部分での操作性の向上を 図ることもときには必要

Web ページを閲覧している状況は、常にパソコンでモニタに向かってマウスを持っているとは限りません。身体的な制約によってマウスがうまく使えない、ノートパソコンを利用しているためキーボードでほとんどを操作するという状況、携帯電話など画面遷移の手段が限られた環境もあります。このように閲覧状態に制約がある場合でもコンテンツの閲覧がページの遷移を楽にすることは可能です。ここではそのようなアクセスを手助けするための方法を紹介しましょう。

## tabindex 属性と accesskey 属性の利用

Web ページに含まれるコンテンツにはさまざまな要素が含まれます。中には応募フォームのような複数のテキストボックスが連続するような状況も考えられます。そのような Web ページでは、とくに制作者側が何も指定しなくても tab キーによるボックス間の遷移がほとんどのブラウザで有効です。前項で解説した WCAG では、このフォーム部分のコントロールに関して tabindex 属性と accesskey 属性を組み込むことが優先事項の 1 つとして記載されています。

普段これらの属性を利用されない方には耳慣れないかもしれませんが、この 2 つの属性の実際の利用法は以下ようになります。tabindex 属性には「0」から「32767」までの数値を指定します。tabindex が指定されている場合は、tab キーを押すごとに数値の小さいほうから大きいほうへ順にフォーカスが移動していきます。一方の accesskey 属性は、「0」から「9」までの数値の他、アルファベットも指定可能です。accesskey として指定された数値やアルファベットは、一般的なブラウザでは固有の修飾キー<sup>\*1</sup> (Windows では alt キー、Macintosh では Ctrl キー) と同時に入力する

### ※1 修飾キー

ショートカットを行う際に数字キーやアルファベットのキーと同時に入力するキーのこと。一般的なアプリケーションでは、Windows は Ctrl キー、Macintosh は command キーが修飾キーとして割り当てられていることが多いでしょう。最近ではそれ以外に Shift キーなどを併用することも増えています。

### 01

```
■ tabindex 属性の記述例
<form method="get" action="hoge hoge.cgi"
  <input type="text" name="search" id="search" tabindex="1" />
  <input type="text" name="name" id="name" tabindex="2" />
</form>

■ accesskey 属性の記述例
<a href="index.html" title=" トップページ " accesskey="1"> トップページ </a>
<a href="prev.html" title=" 前のページ " accesskey="2"> 前のページ </a>
<a href="next.html" title=" 次のページ " accesskey="3"> 次のページ </a>
```

tabindex や accesskey 属性を採用することで表向きは目に見えなくても、入力フォームの遷移やページの移動などがしやすくなる場合もある

ことで該当部分へのアクセスが可能になります。いずれの属性もフォーム部分だけでなくページ内の a 要素に対しても指定することが可能です。これらの属性を Web ページ中で指定しておくことで、連続するフォームや特定の a 要素だけをキーボードからの入力のみで簡単に遷移させることが可能になるというわけです。これらの指定は、特定の利用環境ではとくに有効なアクセス手段の提供にも繋がるので覚えておきましょう (01)。

## accesskey 属性の 利用シーン

accesskey 属性<sup>\*2</sup>は、PC 用の Web ページだけを作成しているとあまり聞き慣れないかもしれませんが、実は皆さんの身近なところでよく利用されています。それはズバリ、モバイル向けの Web ページです。携帯電話のように十字のカーソルキーだけでコンテンツを遷移しようとすると、複数のリンクが点在しているようなものは非常に使い勝手が悪くなります。そこでアクセスが容易になるよう、多くのモバイルコンテンツでは、a 要素のリンクに accesskey 属性を数字で指定してデバイスの数字キーに対応して遷移できるような仕掛けを施しているのです。これは国内の携帯電話各キャリアのサイトだけとは限りません。欧米の Web サービス<sup>\*3</sup>の多くではモバイル対応が当然のように行われているので、その多くでこの accesskey 属性はすでに利用されています (02)。

### ※2 accesskey 属性の利用

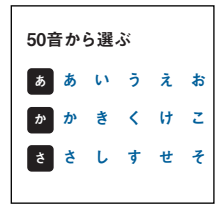
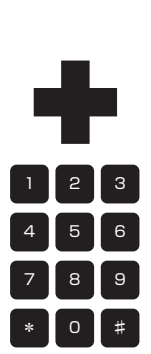
accesskey 属性の利用については「そんなものは誰も使わない」というような意見を耳にすることもあります。しかし、それはあくまでも健常な方や、視覚的なブラウザでしか閲覧しないと考えている方の視野の狭い意見だともいえます。ユーザーに accesskey 属性をうまく利用してもらうためには、利用できる旨や使い方の解説ページなどが必要となる場合も考えられますが、accesskey 属性自体があることが害となるわけではありません。

### ※3 Web サービス

インターネットに関連する技術を利用して、異なるプラットフォーム上でも動作するよう標準的な手段を提供するもの、またはそれを適用したサービスのこと。複数の Web サービスを組み合わせて、新しいサービスを提供することをマッシュアップと呼びます。

### 02

#### 携帯電話でありがちな状況



1つのリンクごとに十字キーを下に入力する必要があり、下のほうのリンクをたどるためにはキー入力は何十回も必要



accesskeyがあれば、キー入力で特定行へ移動できる

コンテンツの遷移手段が限られている携帯電話などのデバイスでは、accesskey 属性などを有効に活用することでアクセスがしやすくなる

#### 欧米圏の Web サービスのモバイル版



欧米圏で展開している Web サイトの多くでは、モバイルデバイス向けのコンテンツにはすでに accesskey が設定されていて使いやすい。画面下のリンクの隣にみえる数字が併記されたリンクにはアクセスキーが指定されている