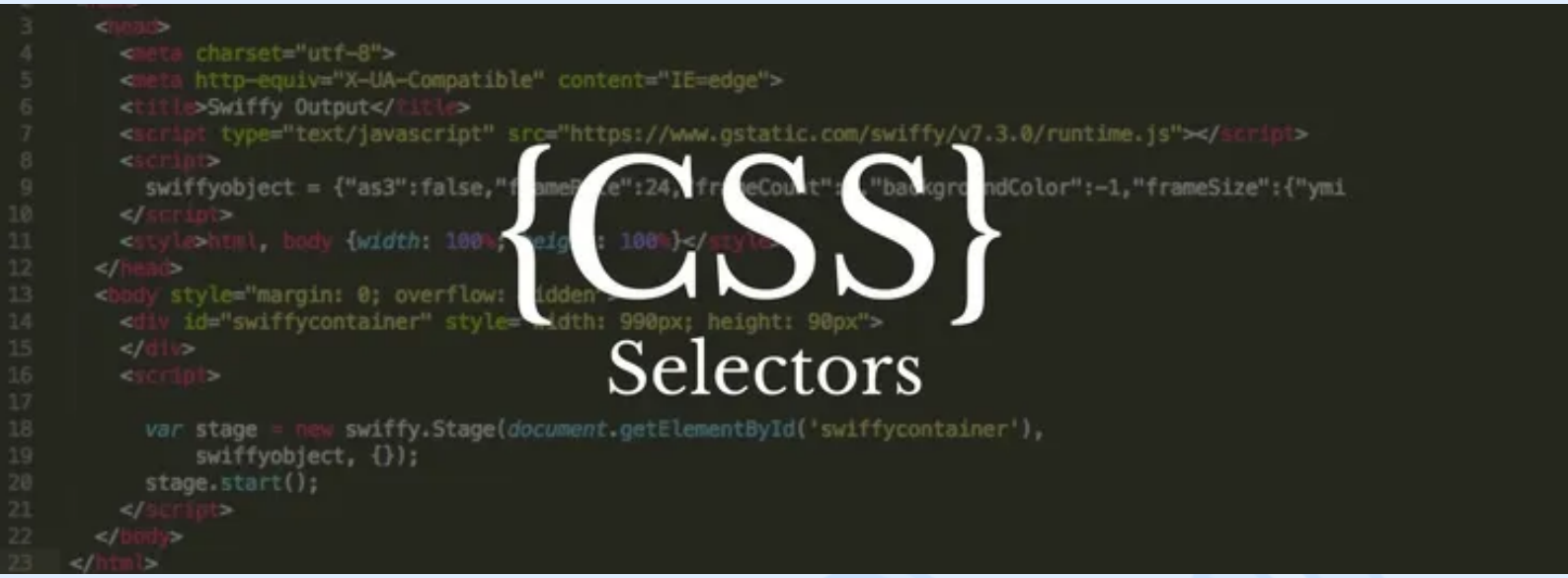

Today's post is dedicated to anyone interested in learning different tricks and techniques for more efficient HTML coding. One of the essential problems that every HTML designer faces sooner or later is how to style elements while keeping the code precise and clear. The easiest way is to attribute a dozen styles to each element, but that's not what we're going to deal with here. Instead, we're going to tell you about more efficient ways to represent the target elements.
Selectors are one of the most basic key elements of CSS, which often doesn't get enough attention when working on CSS styles. Most people who work with CSS know how to target an ID, link or class, but sometimes have no idea about the meaning of different selectors. We'll try to clarify how different CSS selectors are used and how they can help you build a simple yet outstanding website.

CSS Hierarchy

Before we discuss complex CSS selectors, you need to get an understanding of HTML hierarchy. You'll no doubt have heard of it, so let's piece together the various bits you might have already come across so you can clearly grasp how it works.
CSS elements are like one big family that has:

- Children selectors
- Parent selectors
- Sibling selectors
- Ancestor selectors
- Descendant selectors

Let's use some simple code to see what this means in practice:

```
<div id="example001">
    <p>Hello world.</p>
    <p>Here is a <a href="#">black cat</a>.</p>
</div>
```


+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk

There are two paragraphs and an anchor in this div - they are all its descendants. The two paragraphs are the div's children, while the anchor is the child of the second paragraph. If we go back up, we see that div is the parent of the paragraphs and the ancestor of the anchor.
Now here's the easiest part - the paragraphs are siblings, because they are both children of the div element. The anchor has no sibling, because it is the only child of paragraph two.
As you can see, the terminology is pretty intuitive; you just have to think of the family relations and you will easily get what it all means.

Child Selectors

The child selector is represented by the symbol ">". You can use it to introduce changes that will only affect the child elements. Let's look at it in the example:

```
<div id="example002">
    <p>Paragraph one <a href="#">with a link</a>.</p>
    <p>Paragraph two</p>
</div>
```

Paragraph one with a link.

Paragraph two

Let's say we want to change the link colour to red in one simple step. It's obvious that the element we want to change, the anchor, is the child of paragraph one. Here's what you need to do:

```
p > a {
    color: red;
}
```

Paragraph one with a link.

Paragraph two

Sibling Selectors

The adjacent sibling selectors are represented by the plus symbol (+). Using +, we can target any element that has a sibling:

```
<div id="example003">
    <p>Paragraph one</p>
    <p>Paragraph two</p>
    <p>Paragraph three</p>

    <div id="nestedDiv">
        <p>Paragraph four</p>
    </div>
</div>
```


+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk

```
p + p {
    color: green;
}
```

Paragraph one

Paragraph two

Paragraph three

Paragraph four

You may ask why only the second and third paragraphs change their colour. After all, the first one is also a sibling of the other elements on its level. The answer is that the selector only works for elements which are preceded directly by some other element (in our case by a sibling paragraph). For example, the following code changes the background of the nested div to yellow because it is preceded by a paragraph:

```
p + div {
    background: yellow;
}

```

Paragraph one

Paragraph two

Paragraph three

Paragraph four

You could find dozens of other ways to target the elements given in the examples above, descendant selectors being only one of them. Nonetheless, it is a very important aspect of CSS, because understanding CSS selectors and their relationships to HTML elements will make you a better coder and designer who knows how to handle styles quickly and efficiently.

+44 20 7183 5820
info@magora.co.uk
sales@magora.co.uk