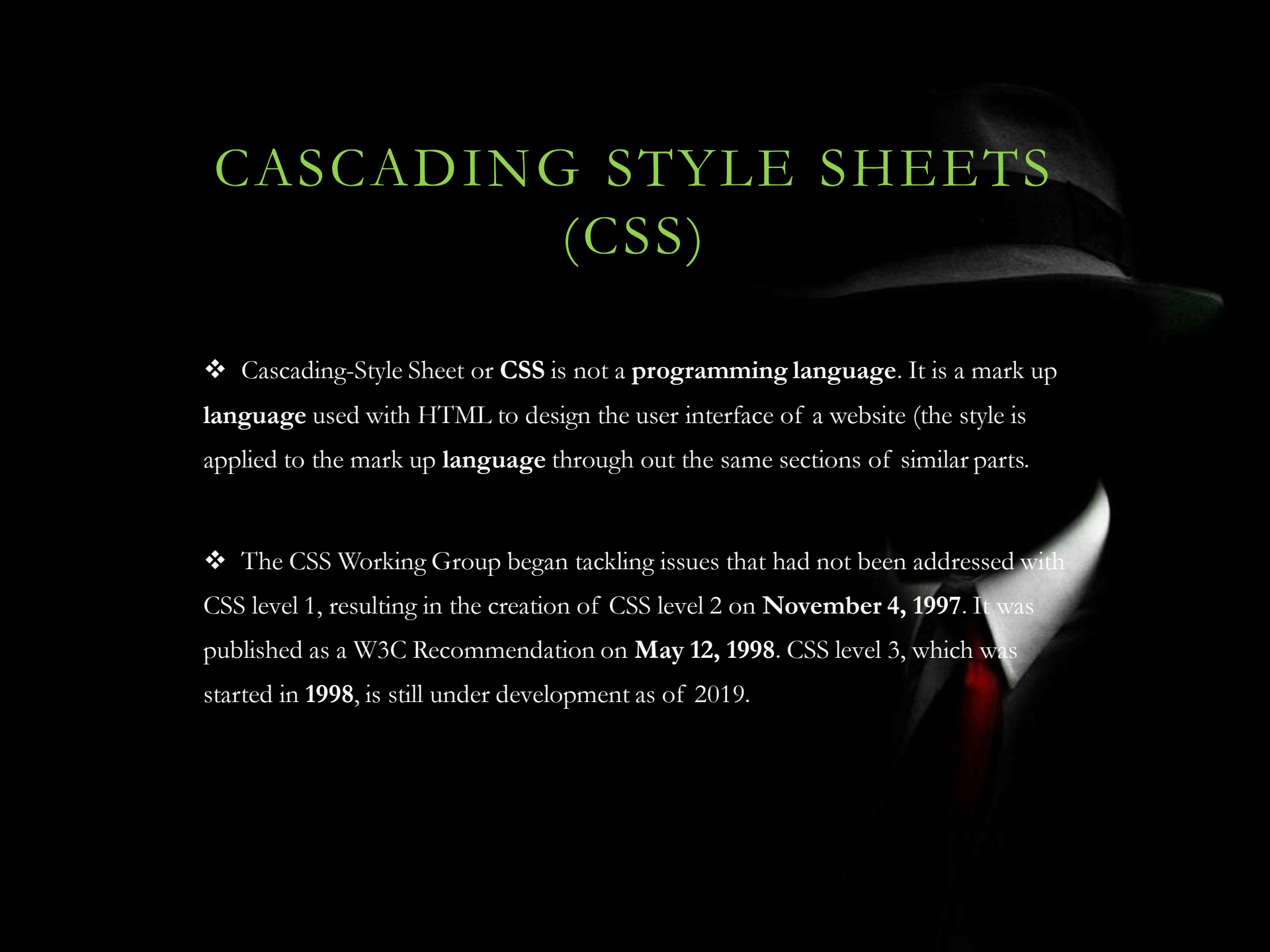
DESIGN FOR WEB USING MARKUP LANGUAGE AND STYLE SHEET

Design By AJ

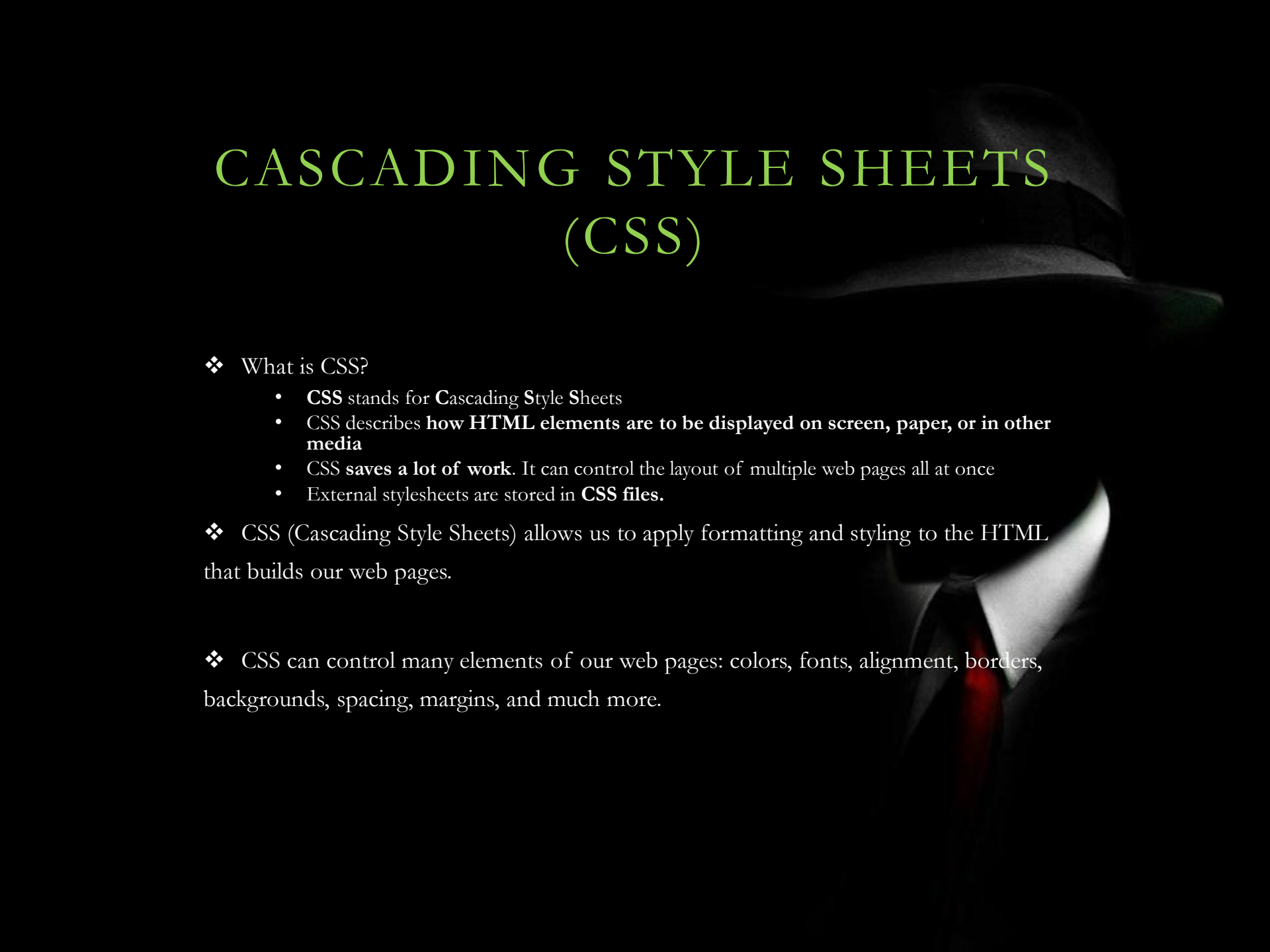
CASCADING STYLE SHEETS (CSS)



❖ Cascading-Style Sheet or **CSS** is not a **programming language**. It is a mark up **language** used with HTML to design the user interface of a website (the style is applied to the mark up **language** through out the same sections of similar parts.

❖ The CSS Working Group began tackling issues that had not been addressed with CSS level 1, resulting in the creation of CSS level 2 on **November 4, 1997**. It was published as a W3C Recommendation on **May 12, 1998**. CSS level 3, which was started in **1998**, is still under development as of 2019.

CASCADING STYLE SHEETS (CSS)

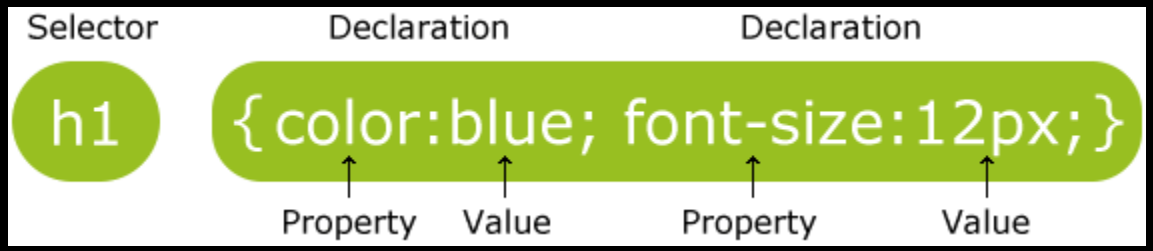


- What is CSS?
  - **CSS** stands for **C**ascading **S**tyle **S**heets
  - CSS describes **how HTML elements are to be displayed on screen, paper, or in other media**
  - CSS **saves a lot of work**. It can control the layout of multiple web pages all at once
  - External stylesheets are stored in **CSS files.**
- CSS (Cascading Style Sheets) allows us to apply formatting and styling to the HTML that builds our web pages.

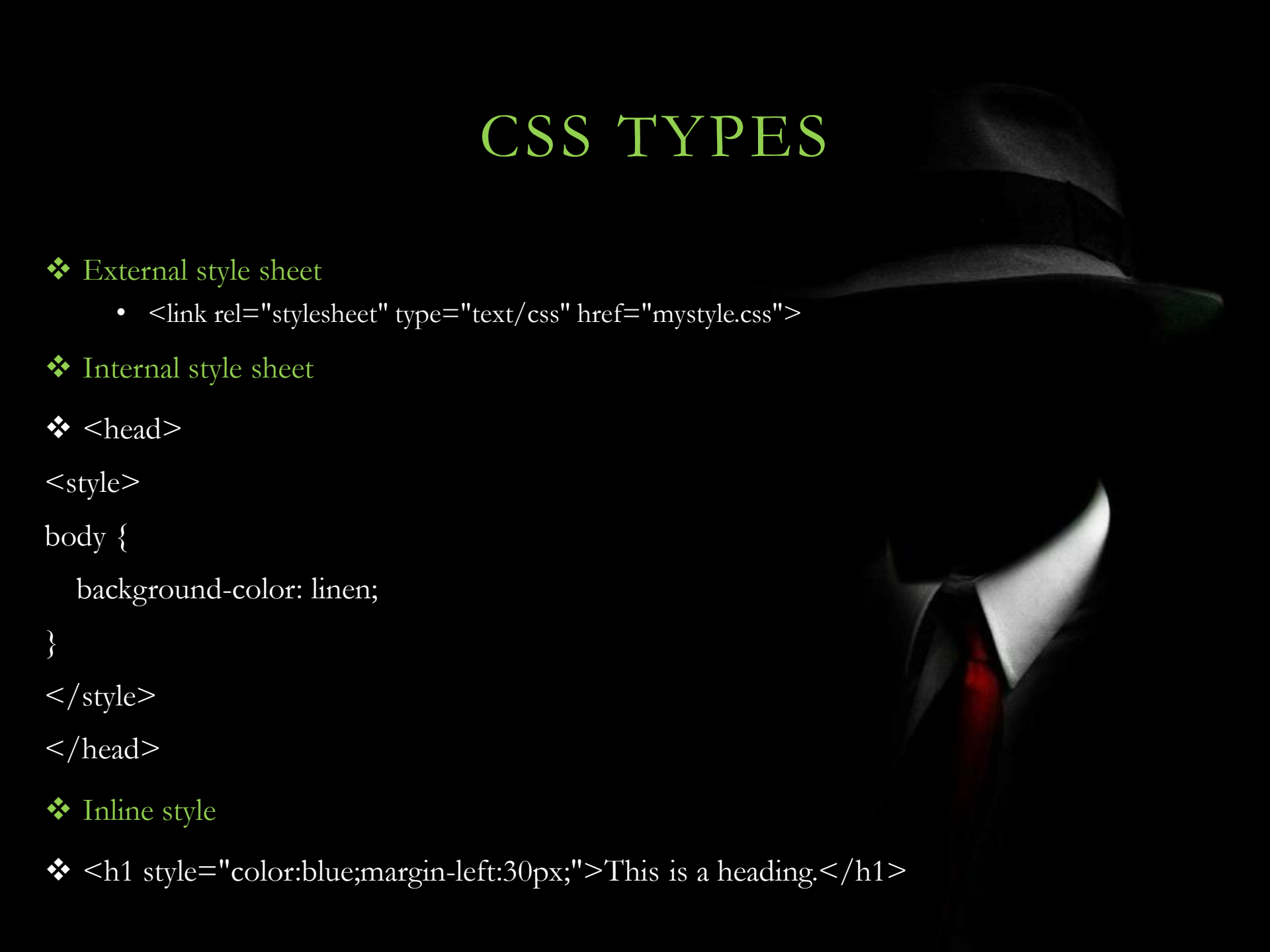
- CSS can control many elements of our web pages: colors, fonts, alignment, borders, backgrounds, spacing, margins, and much more.

CASCADING STYLE SHEETS (CSS)

- CSS Syntax



CSS TYPES



- External style sheet
  - <link rel="stylesheet" type="text/css" href="mystyle.css">
- Internal style sheet
- <head>

```
<style>
body {
    background-color: linen;
}
</style>
</head>
```

- Inline style
- <h1 style="color:blue;margin-left:30px;">This is a heading.</h1>


CSS SELECTOR

CSS SELECTOR

❖ **The Universal Selectors**

- Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type:

```
* {
color: #000000;
}
```

❖ **Tag Selectors**

- Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type:

```
h1 {
color: #000000;
}
```

CSS SELECTOR

❖The Class Selectors

- You can define style rules based on the class attribute of the elements. All the elements having that class will be formatted according to the defined rule.

```
.black {
color: #000000;
}
```

❖The ID Selectors

- You can define style rules based on the *id* attribute of the elements. All the elements having that *id* will be formatted according to the defined rule.

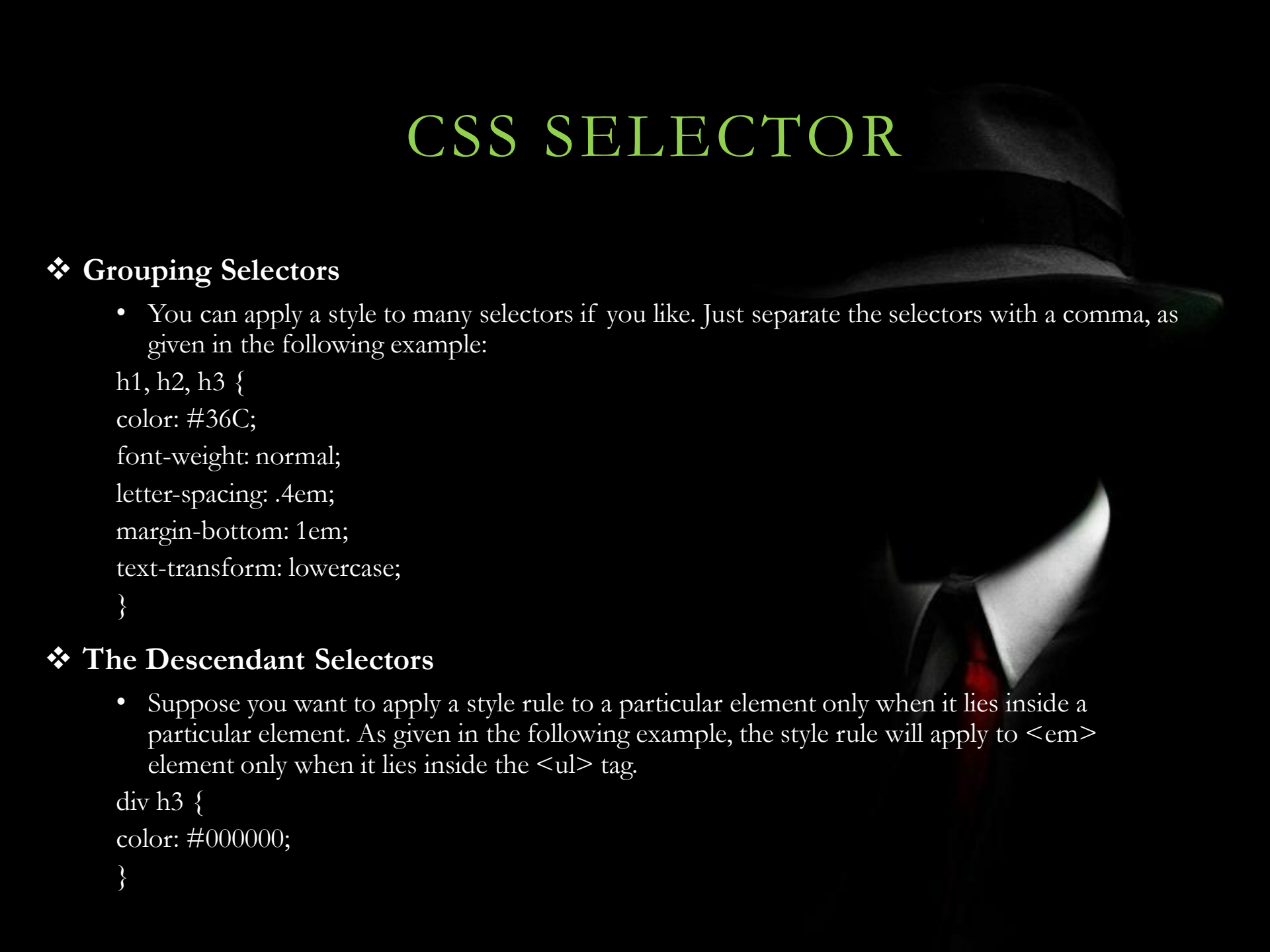
```
#black {
color: #000000;
}
```

What is the difference between ID and class?

- IDs and classes function the same way – they can both provide the same styling functionality to an HTML element, however…
  - IDs are unique; each element can only have one ID, and that ID can only be on the page once.
  - Classes are not unique; an element can have multiple classes, and multiple elements can have the same class.

- What does that mean?
  - IDs can be used to style elements that are different from anything else on the page.
  - Classes can be used to style multiple elements on a single page that have things in common, like font size, color, or style.

CSS SELECTOR



- **Grouping Selectors**
  - You can apply a style to many selectors if you like. Just separate the selectors with a comma, as given in the following example:

```
h1, h2, h3 {
color: #36C;
font-weight: normal;
letter-spacing: .4em;
margin-bottom: 1em;
text-transform: lowercase;
}
```

- **The Descendant Selectors**
  - Suppose you want to apply a style rule to a particular element only when it lies inside a particular element. As given in the following example, the style rule will apply to element only when it lies inside the <ul> tag.

```
div h3 {
color: #000000;
}
```

CSS SELECTOR

❖ **The Child Selectors**

- You have seen the descendant selectors. There is one more type of selector, which is very similar to descendants but have different functionality. Consider the following example:

```
body > p {
color: #000000;
}
```

❖ **The Attribute Selectors**

- You can also apply styles to HTML elements with particular attributes. The style rule below will match all the input elements having a type attribute with a value of *text*:

```
input[type="text"]{
color: #000000;
}
```