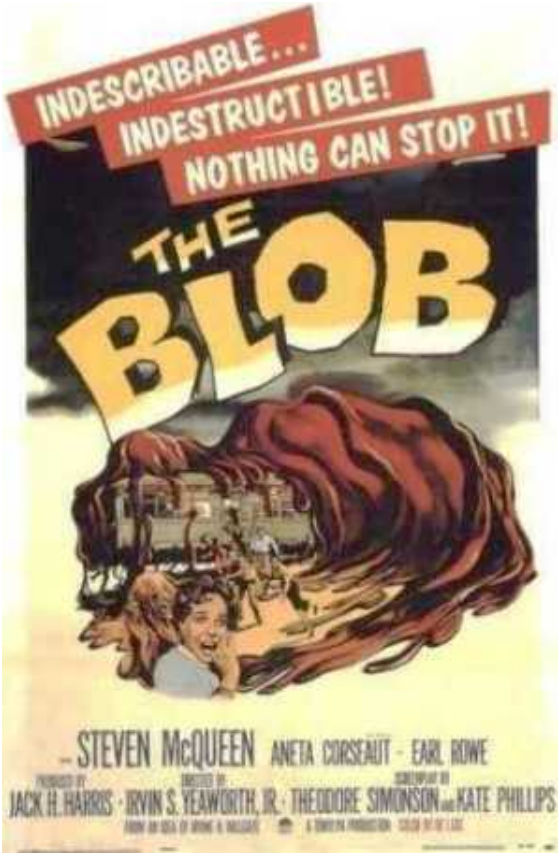
Blob and File: what are they?

THE HTML5 FILE API SPECIFICATION INTRODUCES SEVERAL NEW INTERFACES



- **The `FileList` interface**(we already met it: the `files` property is a `FileList`),
- **the `File` interface**that is useful for getting details about a file (the `file`variable in the `for`loop of the last example illustrates this) .
- **the `Blob` interface**helps read binary data (only) that is accessed slice by slice (as chunks of data, each one being a "Blob"),
- **and a `FileReader` interface** for reading file content (we will see how to use it in the next section of the course),

We will not use all of these new interfaces, but let's explain the difference between `Blob` and `File`, as most of the methods exposed by the `FileReader` interface take indiscriminately a`Blob` or a `File` as parameter.

THE BLOB OBJECT

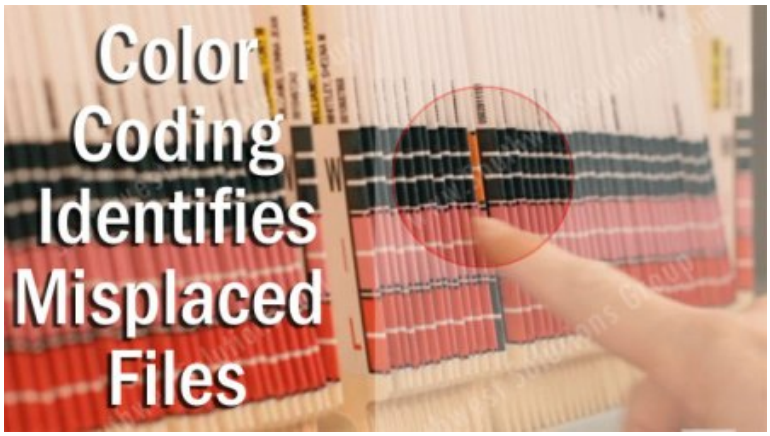
An object of type `Blob` is a structure that represents binary data available as read-only. Most of the time, you will only encounter these objects when you handle files.

Blob objects have two properties, namely:`size` and `type`, which respectively retrieve the size in bytes of the data handled by the Blob and their MIME type.

There is also a method called `slice()`, but this is not used in common applications. If

you are curious, check the "slicing a file" section of this "Reading files in JavaScript using the File APIs" article.

THE FILE OBJECT



File objects are useful for manipulating... files! They inherit the properties and methods of Blob objects, and have two additional properties that are`name`, for the file name, and `lastModifiedDate` to get the date of the last modification of the file (in the form of a JavaScript Date object, obviously) .

Most of the time, we will work with `File` objects. `Blob` objects will have real interest when the Filesystem API is widely available (at the moment there is only an experimental version in Chrome), or when you download binary files using Ajax (see example below). This last API will be covered in the HTML5 part-2 course.

[*Advanced*] If you are interested in seeing how `Blob` objects can be used, here is an example "as is" that shows how to download an image using Xhr2 (Xml Http Request version 2). The examples uses a `<progress>` element to show the download progress, and uses `xhr.responseType = 'blob';` to indicate that the file we are going to download is a binary file (a blob). Try the example, then comment the line with `responseType='blob'`. In this case, you will notice that the image file is not properly decoded by the browser and is not displayed in the page. Xhr2 will be covered in the HTML5 part-2 course.