

[Hummingbird](#) / MediaType

Structure

MediaType

Define media type of file

```
struct MediaType
```

Topics

Structures

`struct Category`

Media type categories

`struct FileExtension`

Type safe wrapper for file extensions. Additionally provides case insensitive comparison.

Operators

```
static func ~= (`Self`, `Self`) -> Bool
```

Initializers

`init?(from: String)`

Construct MediaType from header value

```
init(type: Category, subType: String, parameter: (String, String)?)
```

Initialize MediaType

Instance Properties

```
var description: String
```

Output

```
let parameter: (name: String, value: String)?
```

optional parameter

```
let subType: String
```

exact kind of data specified

```
let type: Category
```

general category

Instance Methods

```
func isType(MediaType) -> Bool
```

Return if media type matches the input

```
func withParameter(name: String, value: String) -> MediaType
```

Return media type with new parameter

Type Properties

```
static var application: `Self`
```

```
static var application7z: `Self`
```

7-zip archive

```
static var applicationAbiWord: `Self`
```

AbiWord document

```
static var applicationAmzKindleEBook: `Self`
```

Amazon Kindle eBook format

`static var applicationArc: `Self``
Archive document (multiple files embedded)

`static var applicationBinary: `Self``
Any kind of binary data

`static var applicationBzip: `Self``
BZip archive

`static var applicationBzip2: `Self``
BZip2 archive

`static var applicationCsh: `Self``
C-Shell script

`static var applicationDocx: `Self``
Microsoft Word (OpenXML)

`static var applicationEot: `Self``
MS Embedded OpenType fonts

`static var applicationEpub: `Self``
Electronic publication (EPUB)

`static var applicationGzip: `Self``
GZip Compressed Archive

`static var applicationJar: `Self``
Java Archive (JAR)

`static var applicationJson: `Self``
JSON format

`static var applicationJsonLD: `Self``
JSON-LD format

`static var applicationManifest: `Self``
Manifest

```
static var applicationMpkg: `Self`
```

Apple Installer Package

```
static var applicationMword: `Self`
```

Microsoft Word

```
static var applicationOdp: `Self`
```

OpenDocument presentation document

```
static var applicationOds: `Self`
```

OpenDocument spreadsheet document

```
static var applicationOdt: `Self`
```

OpenDocument text document

```
static var applicationOgg: `Self`
```

OGG

```
static var applicationPdf: `Self`
```

Adobe Portable Document Format

```
static var applicationPhp: `Self`
```

Hypertext Preprocessor

```
static var applicationPpt: `Self`
```

Microsoft PowerPoint

```
static var applicationPptx: `Self`
```

Microsoft PowerPoint (OpenXML)

```
static var applicationRar: `Self`
```

RAR archive

```
static var applicationRtf: `Self`
```

Rich Text Format (RTF)

```
static var applicationSh: `Self`
```

Bourne shell script

```
static var applicationSwf: `Self`
```

Small web format (SWF) or Adobe Flash document

```
static var applicationTar: `Self`
```

Tape Archive (TAR)

```
static var applicationUrlEncoded: `Self`
```

URL encoded form data

```
static var applicationVsd: `Self`
```

Microsoft Visio

```
static var applicationWasm: `Self`
```

WebAssembly

```
static var applicationXhtml: `Self`
```

XHTML

```
static var applicationXls: `Self`
```

Microsoft Excel

```
static var applicationXlsx: `Self`
```

Microsoft Excel (OpenXML)

```
static var applicationXml: `Self`
```

XML

```
static var applicationZip: `Self`
```

ZIP archive

```
static var audio: `Self`
```

```
static var audio3g2: `Self`
```

3GPP2 audio/video container

```
static var audio3gp: `Self`
```

3GPP audio/video container

```
static var audioAac: `Self`
```

AAC audio

```
static var audioMidi: `Self`  
    Musical Instrument Digital Interface (MIDI)
```

```
static var audioMpeg: `Self`  
    MP3 audio
```

```
static var audioOgg: `Self`  
    OGG audio
```

```
static var audioOpus: `Self`  
    Opus audio
```

```
static var audioWave: `Self`  
    Waveform Audio Format
```

```
static var audioWebm: `Self`  
    WEBM audio
```

```
static var example: `Self`
```

```
static var font: `Self`
```

```
static var fontOtf: `Self`  
    OpenType font
```

```
static var fontTtf: `Self`  
    TrueType Font
```

```
static var fontWoff: `Self`  
    Web Open Font Format (WOFF)
```

```
static var fontWoff2: `Self`  
    Web Open Font Format (WOFF)
```

```
static var image: `Self`
```

```
static var imageBmp: `Self`  
    Windows OS/2 Bitmap Graphics
```

`static var imageGif: `Self``
Graphics Interchange Format (GIF)

`static var imageIco: `Self``
Icon format

`static var imageJpeg: `Self``
JPEG images

`static var imagePng: `Self``
Portable Network Graphics

`static var imageSvg: `Self``
Scalable Vector Graphics (SVG)

`static var imageTiff: `Self``
Tagged Image File Format (TIFF)

`static var imageWebp: `Self``
WEBP image

`static var message: `Self``

`static var model: `Self``

`static var multipart: `Self``

`static var multipartForm: `Self``
Multipart formdata

`static var text: `Self``

`static var textCss: `Self``
Cascading Style Sheets (CSS)

`static var textCsv: `Self``
Comma-separated values (CSV)

`static var textHtml: `Self``
HyperText Markup Language (HTML)

```
static var textIcalendar: `Self`  
    iCalendar format
```

```
static var textJavascript: `Self`  
    JavaScript
```

```
static var textPlain: `Self`  
    Text, (generally ASCII or ISO 8859-n)
```

```
static var video: `Self`
```

```
static var video3g2: `Self`  
    3GPP2 audio/video container
```

```
static var video3gp: `Self`  
    3GPP audio/video container
```

```
static var videoMp4: `Self`  
    AVI: Audio Video Interleave
```

```
static var videoMpeg: `Self`  
    MPEG Video
```

```
static var videoOgg: `Self`  
    OGG video
```

```
static var videoTs: `Self`  
    MPEG transport stream
```

```
static var videoWebm: `Self`  
    WEBM video
```

Type Methods

```
static func getMediaType(forExtension: FileExtension) -> MediaType?  
    Get media type from a file extension
```

```
static func getMediaType(forExtension: String) -> MediaType?
```


Get media type from a file extension

Relationships

Conforms To

`Swift.CustomStringConvertible`, `Swift.Sendable`

See Also

Request/Response

`struct Request`

Holds all the values required to process a request

`typealias Parameters`

Parameters is a special case of `FlatDictionary` where both the key and value types are `Substrings`. It is used for parameters extracted from URIs

`struct CacheControl`

Associates cache control values with filename

`struct Response`

Holds all the required to generate a HTTP Response

`protocol ResponseBodyWriter`

HTTP Response Body part writer

`struct EditedResponse`

`struct Cookie`

Structure holding a single cookie

struct Cookies

Structure holding an array of cookies
