🔲 **Documentation**   Language: Swift

/ Cookie

Structure

Cookie

Structure holding a single cookie

```
struct Cookie
```

Topics

Structures

```
struct Properties
```
    Cookie properties table

Initializers

```
init(name: String, value: String, expires: Date?, maxAge: Int?,
domain: String?, path: String?, secure: Bool, httpOnly: Bool)
```
    Create Cookie

```
init(name: String, value: String, expires: Date?, maxAge: Int?,
domain: String?, path: String?, secure: Bool, httpOnly: Bool, same
Site: SameSite)
```
    Create Cookie

Instance Properties

```
var description: String
```

Output cookie string

```
var domain: String?
```

specifies those hosts to which the cookie will be sent

```
var expires: Date?
```

indicates the maximum lifetime of the cookie

```
var httpOnly: Bool
```

The HttpOnly attribute limits the scope of the cookie to HTTP requests

```
var maxAge: Int?
```

indicates the maximum lifetime of the cookie in seconds. Max age has precedence over expires (not all user agents support max-age)

```
let name: String
```

Cookie name

```
var path: String?
```

The scope of each cookie is limited to a set of paths, controlled by the Path attribute

```
let properties: Properties
```

properties

```
var sameSite: SameSite?
```

The SameSite attribute lets servers specify whether/when cookies are sent with cross-origin requests

```
var secure: Bool
```

The Secure attribute limits the scope of the cookie to "secure" channels

```
let value: String
```

Cookie value

Enumerations

`enum SameSite`

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 MediaType`

Define media type of file

`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 Cookies

Structure holding an array of cookies