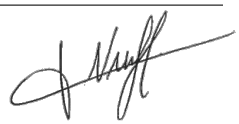

See *read()*

Attributes

buffer
Return the image data as bytes. This is useful to reduce disk access, the data can be send to an image library.

Example with Pillow:

```
>>> from PIL import Image
>>> import io
>>> import pyexiv2
>>> meta = pyexiv2.ImageMetadata("lena.jpg")
>>> meta.read()
>>> byteio = io.BytesIO(meta.buffer)
>>> img = Image.open(byteio)
>>> img.show()
```

comment
The image comment.

dimensions
A tuple containing the width and height of the image, expressed in pixels.

exif_keys
List of the keys of the available EXIF tags.

iptc_charset
An optional character set the IPTC data is encoded in.

iptc_keys
List of the keys of the available IPTC tags.

mime_type
The mime type of the image, as a string.

previews
List of the previews available in the image, sorted by increasing size.

xmp_keys
List of the keys of the available XMP tags.

Methods

copy(*other*, *exif=True*, *iptc=True*, *xmp=True*, *comment=True*)
Copy the metadata to another image. The metadata in the destination is overridden. In particular, if the destination contains e.g. EXIF data and the source doesn't, it will be erased in the destination, unless explicitly omitted.

Arguments:

- *other* An instance of :class:pyexiv2.metadata.ImageMetadata, the destination metadata to copy to (it must have been read() beforehand)
- *exif* (boolean) – Whether to copy the EXIF metadata
- *iptc* (boolean) – Whether to copy the IPTC metadata
- *xmp* (boolean) – Whether to copy the XMP metadata
- *comment* (boolean) – Whether to copy the image comment

__delitem__(*key*)
Delete a metadata tag for a given key.