

Discover Packages > Standard library > image > png 

png







package



standard library

Version: [go1.20.1](#) **Latest** | Published: Feb 14, 2023 | License: [BSD-3-Clause](#) | Imports: [10](#) |

Imported by: [12,392](#)

Details

 Valid [go.mod](#) file   Redistributable license   Tagged version 

 Stable version 



[Learn more](#)

Repository

cs.opensource.google/go/go

Links

 [Report a Vulnerability](#)

 Documentation 

<> Documentation

Overview

Package png implements a PNG image decoder and encoder.

The PNG specification is at <https://www.w3.org/TR/PNG/>.

Index

```
func Decode(r io.Reader) (image.Image, error)
func DecodeConfig(r io.Reader) (image.Config, error)
func Encode(w io.Writer, m image.Image) error
type CompressionLevel
type Encoder
    func (enc *Encoder) Encode(w io.Writer, m image.Image) error
type EncoderBuffer
type EncoderBufferPool
type FormatError
    func (e FormatError) Error() string
type UnsupportedError
    func (e UnsupportedError) Error() string
```

Examples

[Decode](#)

[Encode](#)

Constants

This section is empty.

Variables

This section is empty.

Functions

func Decode

```
func Decode(r io.Reader) (image.Image, error)
```

Decode reads a PNG image from `r` and returns it as an `image.Image`. The type of `Image` returned depends on the PNG contents.

► [Example](#)

func DecodeConfig

```
func DecodeConfig(r io.Reader) (image.Config, error)
```

DecodeConfig returns the color model and dimensions of a PNG image without decoding the entire image.

func Encode

```
func Encode(w io.Writer, m image.Image) error
```

Encode writes the `Image` `m` to `w` in PNG format. Any `Image` may be encoded, but images that are not `image.NRGBA` might be encoded lossily.

► [Example](#)

Types

type CompressionLevel

added in go1.4

```
type CompressionLevel int
```

CompressionLevel indicates the compression level.

```
const (  
    DefaultCompression CompressionLevel = 0  
    NoCompression      CompressionLevel = -1  
    BestSpeed           CompressionLevel = -2  
)
```

```
BestCompression CompressionLevel = -3
)
```

type **Encoder**

added in go1.4

```
type Encoder struct {
    CompressionLevel CompressionLevel

    // BufferPool optionally specifies a buffer pool to get temporary
    // EncoderBuffers when encoding an image.
    BufferPool EncoderBufferPool
}
```

Encoder configures encoding PNG images.

func (*Encoder) **Encode**

added in go1.4

```
func (enc *Encoder) Encode(w io.Writer, m image.Image) error
```

Encode writes the Image m to w in PNG format.

type **EncoderBuffer**

added in go1.9

```
type EncoderBuffer encoder
```

EncoderBuffer holds the buffers used for encoding PNG images.

type **EncoderBufferPool**

added in go1.9

```
type EncoderBufferPool interface {
    Get() *EncoderBuffer
    Put(*EncoderBuffer)
}
```

EncoderBufferPool is an interface for getting and returning temporary instances of the EncoderBuffer struct. This can be used to reuse buffers when encoding multiple images.

type **FormatError**

```
type FormatError string
```

A FormatError reports that the input is not a valid PNG.

func (FormatError) **Error**

```
func (e FormatError) Error() string
```

type **UnsupportedError**

```
type UnsupportedError string
```

An `UnsupportedError` reports that the input uses a valid but unimplemented PNG feature.

func (UnsupportedError) Error

```
func (e UnsupportedError) Error() string
```



Source Files

[View all](#) 

[paeth.go](#)

[reader.go](#)

[writer.go](#)

Why Go

[Use Cases](#)

[Case Studies](#)

Get Started

[Playground](#)

[Tour](#)

[Stack Overflow](#)

[Help](#)

Packages

[Standard Library](#)

[About Go Packages](#)

About

[Download](#)

[Blog](#)

[Issue Tracker](#)

[Release Notes](#)

[Brand Guidelines](#)

[Code of Conduct](#)

Connect

[Twitter](#)

[GitHub](#)

[Slack](#)

[r/golang](#)

[Meetup](#)

[Golang Weekly](#)

[Copyright](#)

[Terms of Service](#)

[Privacy Policy](#)

[Report an Issue](#)



Google