Package: blob (via r-universe)

June 22, 2024

Title A Simple S3 Class for Representing Vectors of Binary Data ('BLOBS')

Version 1.2.4.9003

Description R's raw vector is useful for storing a single binary object. What if you want to put a vector of them in a data frame? The 'blob' package provides the blob object, a list of raw vectors, suitable for use as a column in data frame.

License MIT + file LICENSE

URL <https://blob.tidyverse.org>, <https://github.com/tidyverse/blob>

BugReports <https://github.com/tidyverse/blob/issues>

Imports methods, rlang, vctrs (>= 0.2.1)

Suggests covr, crayon, pillar (>= 1.2.1), testthat

Config/autostyle/scope line_breaks

Config/autostyle/strict false

Config/Needs/website tidyverse/tidytemplate

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Repository https://tidyverse.r-universe.dev

RemoteUrl https://github.com/tidyverse/blob

RemoteRef HEAD

RemoteSha 37b4fc5ef82ab866e4864e3a6fd2bfa89305f8b0

Contents

1

---

blob   *Construct a blob object*

Description

new_blob() is a low-level constructor that takes a list of raw vectors. blob() constructs a blob from
individual raw vectors. as_blob() and is_blob() are simple forwarders to vctrs::vec_cast()
and inherits(), respectively.

Usage

```
blob(...)

new_blob(x = list())

validate_blob(x)

as_blob(x)

is_blob(x)
```

Arguments

...	Individual raw vectors
x	A list of raw vectors, or other object to coerce

See Also

as.blob() for the legacy interface for specifying casts.

Examples

```
x1 <- charToRaw("Good morning")
x2 <- as.raw(c(0x48, 0x65, 0x6c, 0x6c, 0x6f))

new_blob(list(x1, x2))
blob(x1, x2)

as.blob(c("Good morning", "Good evening"))
```

---

vec_ptype2.blob *Coercion*

Description

Double dispatch methods to support `vctrs::vec_ptype2()`.

Usage

```
## S3 method for class 'blob'
vec_ptype2(x, y, ..., x_arg = "", y_arg = "")
```

Arguments

x, y	Vector types.
...	These dots are for future extensions and must be empty.
x_arg, y_arg	Argument names for x and y. These are used in error messages to inform the user about the locations of incompatible types (see `stop_incompatible_type()`).

Index