

# Package: pbr (via r-universe)

October 18, 2024

**Type** Package

**Title** Find a Cold One Near You

**Version** 0.0.2

**Maintainer** Paul Frater <pfrater@wisc.edu>

**Description** In short, this package is a locator for cool, refreshing beverages. It will find and return the nearest location where you can get a cold one.

**Depends** R (>= 4.2.0)

**Imports** httr (>= 1.4.2), jsonlite (>= 1.8.0), leaflet (>= 2.1.1),  
htmltools (>= 0.5.2)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Paul Frater [aut, cre]  
(<<https://orcid.org/0000-0002-7237-6563>>)

**Repository** CRAN

**Date/Publication** 2023-08-25 15:20:02 UTC

## Contents

format_leaflet_labels . . . . .	2
format_pbr_url . . . . .	2
ip_zip . . . . .	3
location_query . . . . .	3
milwaukee . . . . .	4
pbr_me . . . . .	4
pbr_query . . . . .	5

**Index****6**


---

format\_leaflet\_labels *Format outlet name and address for use as a leaflet label*

---

**Description**

Takes name and address and formats it into an HTML label. This is a shortcut helper function that is used in [pbr\\_me](#)

**Usage**

```
format_leaflet_labels(name, address, city, state, zip)
```

**Arguments**

name	Character. Name of the business
address	Character. Address of the business
city	Character. City name
state	Character. State name
zip	Character or numeric. Zip code

**Value**

An HTML label

---

format\_pbr\_url *Formats url based on the location provided*

---

**Description**

This function simply readies the url for use in a GET request

**Usage**

```
format_pbr_url(location, dist = 100, lim = 50, brand_id = 333)
```

**Arguments**

location	Zip code or (portion of) city name
dist	Numeric. The distance in miles to search from the location
lim	Numeric. The number of results to be returned
brand_id	Numeric. The brand ID to be returned

**Value**

A url to be passed to [pbr\\_query](#)

---

`ip_zip`*Functions to retrieve IP address and ZIP code from IP*

---

**Description**

These are just helper shortcut functions. `get_ip_address` retrieves a computer's IP address from <https://ipinfo.io/what-is-my-ip>. This is easier than getting the IP right off local computer because of bogon IP addresses. `get_zip` pulls ZIP code location from <https://ipapi.co/>

**Usage**

```
get_ip_address()
```

```
get_zip()
```

**Value**

Either the IP address that a web browser sees (`get_ip_address`), or a zip code (`get_zip`)

---

`location_query`*Query information about a location*

---

**Description**

This function is used to find the information needed for an entire GET request url based on just the zip code or city name (or regex)

**Usage**

```
location_query(location)
```

**Arguments**

`location`      Zip code or (portion of) city name

**Value**

A data.frame with city name, zip code, lat, and long

---

milwaukee	<i>Retailers in and around Milwaukee, WI</i>
-----------	--

---

**Description**

A dataset containing the names and locations for the 50 closest retailers closest to ZIP code 53210.

**Usage**

```
milwaukee
```

**Format**

A data.frame with 50 rows and 12 columns

**Source**

```
milwaukee <- pbr_me(53210, map = FALSE)
```

---

pbr_me	<i>Retrieve and print interactive map of closest locations</i>
--------	--

---

**Description**

These functions will retrieve and display the locations of the closest outlets for a cold one. `pbr_me` requires a location to be input. This is handy for when you're going somewhere and want to scout out the available outlets that sell what you're looking for. `pbr_me_asap` is for when you just need a one now and don't have time to enter your zip code.

**Usage**

```
pbr_me(location, map = TRUE, ...)
```

```
pbr_me_asap()
```

**Arguments**

location	A zip code, city name, or regular expression of such
map	Logical. Output a leaflet map (TRUE, default) or not (FALSE)
...	Additional arguments passed on to <a href="#">format_pbr_url</a>

**Details**

`pbr_me` will display a [leaflet](#) map of the closest retailer locations. If `map = FALSE` it will return a data.frame of the locations. `pbr_me_asap` displays a leaflet map of retailer locations within your current proximity for those moments when you just need a cold one now

**Value**

A leaflet map displaying closest retailer locations, or (optionally) a data.frame of retailer locations

**Examples**

```
## Not run:  
  
pbr_me(54481)  
pbr_me_asap() # for when you just don't have time to enter your zip code  
  
## End(Not run)
```

---

pbr\_query

*Query for locations that sell cold ones*

---

**Description**

Query for locations that sell cold ones

**Usage**

```
pbr_query(location, ...)
```

**Arguments**

location	Zip code or (portion of) city name
...	Additional arguments to be passed to <a href="#">format_pbr_url</a>

**Value**

A data.frame of retailers

# Index

## \* datasets

- milwaukee, 4
- format\_leaflet\_labels, 2
- format\_pbr\_url, 2, 4, 5
- get\_ip\_address(ip\_zip), 3
- get\_zip(ip\_zip), 3
- ip\_zip, 3
- leaflet, 4
- location\_query, 3
- milwaukee, 4
- pbr\_me, 2, 4
- pbr\_me\_asap(pbr\_me), 4
- pbr\_query, 2, 5