

# Package: tidyUSDA (via r-universe)

November 1, 2024

**Type** Package

**Title** A Minimal Tool Set for Gathering USDA Quick Stat Data for Analysis and Visualization

**Version** 0.4.1.9000

**Description** Provides a consistent API to pull United States Department of Agriculture census and survey data from the National Agricultural Statistics Service (NASS) QuickStats service.

**License** MIT + file LICENSE

**URL** <https://bradlindblad.github.io/tidyUSDA/>,  
<https://github.com/bradlindblad/tidyUSDA/>

**Depends** R (>= 3.6)

**Imports** checkmate, crayon, dplyr, fuzzyjoin (>= 0.1.6), ggplot2, httr, jsonlite, magrittr, sf, tigris (>= 1.0)

**Suggests** covr, knitr, nlme, rmarkdown, spelling, stringi, testthat (>= 2.1.0), usethis, waldo

**VignetteBuilder** knitr

**Encoding** UTF-8

**Language** en-US

**LazyData** true

**RoxygenNote** 7.1.2

**Repository** <https://bradlindblad.r-universe.dev>

**RemoteUrl** <https://github.com/bradlindblad/tidyusda>

**RemoteRef** HEAD

**RemoteSha** f72e05aa43d1fbce9c6ea511b4490fe7ce5ceb75

## Contents

allCategory . . . . .	2
allCommodity . . . . .	2

allCounty . . . . .	3
allDataItem . . . . .	3
allDomain . . . . .	4
allGeogLevel . . . . .	4
allGroup . . . . .	5
allProgram . . . . .	5
allSector . . . . .	6
allState . . . . .	6
getQuickstat . . . . .	7
plotUSDA . . . . .	8
tidyUSDA . . . . .	9

**Index** **10**

---

allCategory *All possible values from the CATEGORY field.*

---

**Description**

All possible values from the CATEGORY field.

**Usage**

allCategory

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allCommodity *All possible values from the COMMODITY field.*

---

**Description**

All possible values from the COMMODITY field.

**Usage**

allCommodity

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allCounty

*All possible values from the COUNTY field.*

---

**Description**

All possible values from the COUNTY field.

**Usage**

allCounty

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allDataItem

*All possible values from the DATA ITEM field.*

---

**Description**

All possible values from the DATA ITEM field.

**Usage**

allDataItem

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allDomain	<i>All possible values from the DOMAIN field.</i>
-----------	---

---

**Description**

All possible values from the DOMAIN field.

**Usage**

allDomain

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allGeogLevel	<i>All possible values from the GEOGRAPHY LEVEL field.</i>
--------------	--

---

**Description**

All possible values from the GEOGRAPHY LEVEL field.

**Usage**

allGeogLevel

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

<code>allGroup</code>	<i>All possible values from the GROUP field.</i>
-----------------------	--

---

**Description**

All possible values from the GROUP field.

**Usage**

`allGroup`

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

<code>allProgram</code>	<i>All possible values from the PROGRAM field.</i>
-------------------------	--

---

**Description**

All possible values from the PROGRAM field.

**Usage**

`allProgram`

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allSector	<i>All possible values from the SECTOR field.</i>
-----------	---

---

**Description**

All possible values from the SECTOR field.

**Usage**

allSector

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

allState	<i>All possible values from the STATE field.</i>
----------	--

---

**Description**

All possible values from the STATE field.

**Usage**

allState

**Format**

A vector with 1 variable

**Source**

<https://quickstats.nass.usda.gov>

---

getQuickstat	<i>getQuickstat</i>
--------------	---------------------

---

### Description

Get values from USDA Quick Stats in a dataframe with optional sf (simple features) geometry field

### Usage

```
getQuickstat(
  key = NULL,
  program = NULL,
  data_item = NULL,
  sector = NULL,
  group = NULL,
  commodity = NULL,
  category = NULL,
  domain = NULL,
  geographic_level = NULL,
  state = NULL,
  county = NULL,
  year = NULL,
  geometry = FALSE,
  lower48 = FALSE,
  weighted_by_area = FALSE
)
```

### Arguments

key	your USDA api key. Get one at <a href="https://quickstats.nass.usda.gov/api">https://quickstats.nass.usda.gov/api</a> - string
program	program field - string
data_item	data_item field - string
sector	sector field - string
group	group field - string
commodity	commodity field - string
category	category field - string
domain	domain field - string
geographic_level	geographic_level field - string
state	state field - either a string or character vector with multiple states
county	county field - either a string or character vector with multiple states
year	year field - string

geometry	geometry field (TRUE or FALSE), set to TRUE if you would like a simple features (SF) geometry field included. Only works when geographic_level is set to 'COUNTY' or 'STATE'
lower48	limit data to the lower 48 states? - TRUE or FALSE
weighted_by_area	option to mutate a new column that takes the target ('Value') and divides it by the square miles in that state or county; only works when GEOMETRY = TRUE - TRUE or FALSE

**Note**

Go to the webpage <https://quickstats.nass.usda.gov/>. As a best practice, select the items in these fields and test that that data item exists in the browser before using those parameters in this function. When you have a dataset that works, enter those values in the function as parameters. Ideally, only enter values for your key obviously, then PROGRAM, DATA\_ITEM, GEOGRAPHIC\_LEVEL and then if necessary, DOMAIN, STATE, COUNTY or YEAR.

**Examples**

```
## Not run:
getQuickstat(
  key = "your_key",
  program = "CENSUS",
  data_item = "CROP TOTALS - OPERATIONS WITH SALES",
  geographic_level = "COUNTY",
  domain = "TOTAL",
  year = "2017",
  state = NULL,
  geometry = T,
  lower48 = T
)

## End(Not run)
```

---

plotUSDA

*plotUSDA*


---

**Description**

Quickly plot a data frame produced by the getQuickstat() function.

**Usage**

```
plotUSDA(df, fill_by = "Value")
```

**Arguments**

df	a data frame with a simple feature column (geometry)
fill_by	the value you would like to fill your choropleth output



**Examples**

```
## Not run:  
# Use output from getQuickstat()  
plotUSDA(df = df_from_getQuickstat)  
  
## End(Not run)
```

---

tidyUSDA

*tidyUSDA: An Interface to USDA QuickStats Data with Mapping Capabilities.*

---

**Description**

A minimal toolset for gathering USDA Quick Stat data for analysis and visualization.

**Author(s)**

**Maintainer:** Brad Lindblad <me@bradlindblad.com>

Other contributors:

- Michael Thomas <mthomas@ketchbrookanalytics.com> [contributor]
- Alex Mindeman <alexandramindeman@gmail.com> [contributor]

**See Also**

Useful links:

- <https://bradlindblad.github.io/tidyUSDA/>
- <https://github.com/bradlindblad/tidyUSDA/>

# Index

## \* datasets

- allCategory, 2
- allCommodity, 2
- allCounty, 3
- allDataItem, 3
- allDomain, 4
- allGeogLevel, 4
- allGroup, 5
- allProgram, 5
- allSector, 6
- allState, 6

- allCategory, 2
- allCommodity, 2
- allCounty, 3
- allDataItem, 3
- allDomain, 4
- allGeogLevel, 4
- allGroup, 5
- allProgram, 5
- allSector, 6
- allState, 6

- getQuickstat, 7

- plotUSDA, 8

- tidyUSDA, 9

- tidyUSDA-package (tidyUSDA), 9