CHRISTOPHER PRENER, PH.D.

SPRING, 2018

WEEK 09

LECTURE 08B

INTRO TO GISc

ACCESSING CENSUS DATA IN R

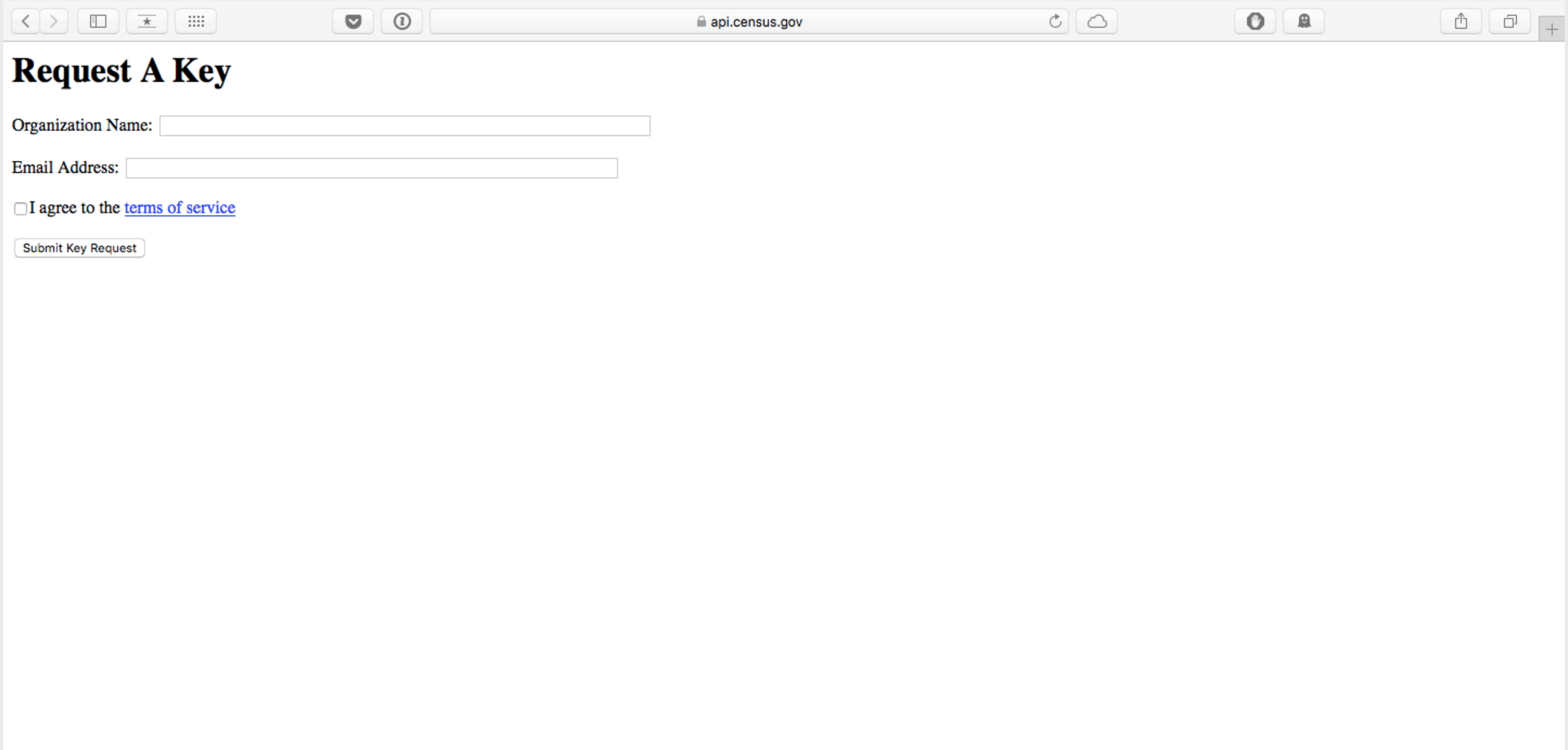
AGENDA

1. `tidycensus` Set-up
2. Census Data
3. American Community Survey Data

1 TIDYCENSUS SET-UP

OBTAIN CENSUS API KEY

Go to https://api.census.gov/data/key_signup.html and complete the form. A key should arrive in your email shortly later!



Request A Key

Organization Name:

Email Address:

□ I agree to the terms of service

Submit Key Request

ADD API KEY

f(x)

```
census_api_key("KEY", install = TRUE)
```

Parameters:

- KEY ue iden
- inst n future sessions

Available in tidycensus
Download via CRAN

ADD API KEY

f(x)

```
census_api_key("KEY", install = TRUE)
```

Parameters:

- `KEY` is the string of letters and numbers given you to as your unique identification for the API

- `install = TRUE` will install the key in your `.Renviron` file for use in future sessions

ADD API KEY

f(x)

```
census_api_key("KEY", install = TRUE)
```

Using a fake API key:

```
> census_api_key("klnahfna4747", install = TRUE)
```

Restart your R session after completing this by either closing and re-opening RStudio or by going to Session ▶ New Session

2 CENSUS DATA

PREVIEW VARIABLES

```
load_variables(year, "product", cache = TRUE)
```

Parameters:

- `year` should be one of `2010`, `2000`, or `1990` (tract boundaries are *not* the same!)
- `product` should be either `"sf1"`, `"sf3"`, or `"acs5"`
- `cache = TRUE` will make re-downloading these data in the same session much faster

PREVIEW VARIABLES

f(x)

```
load_variables(year, "product", cache = TRUE)
```

Using the 2010 census:

```
> census <- load_variables(2010, "sf1", cache = TRUE)
```

Assign the output of this to an object. It will take a few seconds to load these data since there are ~8,000 variables.

DOWNLOAD CENSUS DATA

f(x)

```
get_decennial(geography = "geo", year = year, state =
    "st", county = "county", variable = "variable")
```

Parameters:

- `geography` should be either `"state"`, `"county"`, `"tract"`, or `"block group"`
- `year` should be one of `2010`, `2000`, or `1990` (tract boundaries are *not* the same!)
- `state` should be either numeric FIPS codes (e.g. 29) or string two-letter postal abbreviations (e.g. `"MO"`)

DOWNLOAD CENSUS DATA

f(x)

```
get_decennial(geography = "geo", year = year, state =
    "st", county = "county", variable = "variable")
```

Parameters:

- `county` should be either numeric FIPS codes (e.g. `510`) or string names (e.g. “`St. Louis city`”)
- `variable` should be a string identifying a specific variable within the Decennial Census

DOWNLOAD CENSUS DATA

f(x)

```
get_decennial(geography = "geo", year = year, state =
    "st", county = "county", variable = "variable")
```

Using the population variable for all U.S. states:

```
> statePop <- get_decennial(geography = ‘‘state’’,
    year = ‘‘2010’’, variable = ‘‘P0010001’’)
```

Valid geographies include “state”, “county”, “tract”, “block group”, and “block”. Any sub-county data must be downloaded for specific states (i.e. only for Missouri).

DOWNLOAD CENSUS DATA

f(x)

```
get_decennial(geography = "geo", year = year, state =
    "st", county = "county", table = "table", output =
    "wide")
```

Parameters:

- `table` can be used in place of variable to download an entire table worth of data at once.
- `output` is used to download the data in either long (i.e. `"tidy"`) or `"wide"` form - we almost always want “wide” data in GIS applications.

3 AMERICAN COMMUNITY SURVEY DATA

PREVIEW VARIABLES

f(x)

```
load_variables(year, "product", cache = TRUE)
```

Using the 2016 ACS data:

```
> census <- load_variables(2016, "acs5",
    cache = TRUE)
```

Like the census data, assign the output of this to an object.

DOWNLOAD ACS DATA

```
get_acs(geography = "geo", year = year, state = "st",
    county = "county", variable = "variable",
    survey = "acs5")
```

Parameters:

- `variable` should be a string identifying a specific variable within the American Community Survey

- `survey` refers to the specific type of ACS data; we’ll always use the five year average estimates for this course (e.g. `"acs5"`)

DOWNLOAD ACS DATA

f(x)

```
get_acs(geography = "geo", year = year, state = "st",
    county = "county", table = "table", output =
    "wide", survey = "acs5")
```

Parameters:

- `table` can be used in place of variable to download an entire table worth of data at once.

- `output` is used to download the data in either long (i.e. `"tidy"`) or `"wide"` form - we almost always want “wide” data in GIS applications.