EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Wirtschafts- und
Sozialwissenschaftliche
Fakultät

LEHRSTUHL FÜR MARKETING
Nauklerstr. 47 72074 Tübingen

Assignment V: GitHub and the ticketmaster.com API

Data Science Project Management | Winter Term 2021/22

In this assignment, you will apply what you have learned about APIs and about version control with Git(Hub). First, you will acquire data about event venues using the API provided by ticketmaster.com. You will then use the geospatial data to visualize the extracted data on a map. Finally, you will repeat the same steps for a different country. It is further required that the entire project and its version history is documented in your personal GitHub repository.

1. Setting up a new GitHub repository

- Register on github.com in case you have not done this already.
- Initialize a new public repository for this assignment on GitHub.
- For the following exercises of this assignment, follow the standard Git workflow (i.e., pull the latest version of the project to your local computer, then stage, commit, and push all the modifications that you make throughout the project). Every logical programming step should be well documented on GitHub with a meaningful commit message, so that other people (e.g., your course instructor) can follow and understand the development history. You can do this either using Shell commands or a Git GUI of your choice.
- In the HTML file that you submit, include the hyperlink to the project repository (e.g., *https://github.com/yourUserName/yourProjectName*)

2. Getting to know the API

- Visit the documentation website for the API provided by ticketmaster.com (see here).
- Familiarize yourself with the features and functionalities of the Ticketmaster *Discovery API*. Have a particular look at rate limits.
- Whithin the scope of this assignment, you do **not** have to request your own API key. Instead retrieve a valid key from the API Explorer. This API key enables you to perform the `GET` requests needed throughout this assignment.
- Even though this API key is not secret *per se* (it is publicly visible on the API Explorer website), please comply to the common secrecy practices discussed in the lecture and the tutorial: Treat the API key as a **secret** token. Your API key should neither appear in the code that you are submitting nor in your public GitHub repository.

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Wirtschafts- und
Sozialwissenschaftliche
Fakultät

LEHRSTUHL FÜR MARKETING
Nauklerstr. 47 72074 Tübingen

3. Interacting with the API - the basics

- Load the packages needed to interact with APIs using R.
- Perform a first `GET` request, that searches for event venues in Germany (`countryCode = "DE"`). Extract the `content` from the `response` object and inspect the resulting list. Describe what you can see.
- Extract the `name`, the `city`, the `postalCode` and `address`, as well as the `url` and the `longitude` and `latitude` of the venues to a data frame. This data frame should have the following structure:

```
glimpse(venue_data)
```

```
## Rows: 20
## Columns: 7
## $ name       <chr> "Gruenspan", "Philharmonie (Großer Saal)", "Beatpol", "Klei~
## $ city       <chr> "Hamburg", "Berlin", "Dresden", "Munich", "Berlin", "Freibu~
## $ postalCode <chr> "22767", "10785", "01157", "80809", "12435", "79108", "7647~
## $ address    <chr> "Grosse Freiheit 58", "Herbert-von-Karajan-Straße 1", "Altb~
## $ url        <chr> "http://www.ticketmaster.de/venue/287155", "https://www.tic~
## $ longitude  <dbl> 9.958075, 13.368960, 13.674710, 11.550920, 13.420540, 7.840~
## $ latitude   <dbl> 53.55188, 52.51007, 51.06498, 48.17543, 52.49945, 48.01587,~
```

4. Interacting with the API - advanced

- Have a closer look at the list element named `page`. Did your `GET` request from exercise 3 return *all* event locations in Germany? Obviously not - there are of course much more venues in Germany than those contained in this list. Your `GET` request only yielded the first results page containing the first 20 out of several thousands of venues.
- Check the API documentation under the section *Venue Search*. How can you request the venues from the remaining results pages?
- Write a `for` loop that iterates through the results pages and performs a `GET` request for *all* venues in Germany. After each iteration, extract the seven variables `name`, `city`, `postalCode`, `address`, `url`, `longitude`, and `latitude`. Join the information in one large data frame.
- The resulting data frame should look something like this (note that the exact number of search results may have changed since this document has been last modified):

```
glimpse(venue_data)
```

```
## Rows: 12,175
## Columns: 7
## $ name       <chr> "Gruenspan", "Philharmonie (Großer Saal)", "Beatpol", "West~
## $ city       <chr> "Hamburg", "Berlin", "Dresden", "Essen", "Ludwigsburg", "Be~
## $ postalCode <chr> "22767", "10785", "01157", "45127", "71638", "10117", "0774~
## $ address    <chr> "Grosse Freiheit 58", "Herbert-von-Karajan-Straße 1", "Altb~
## $ url        <chr> "http://www.ticketmaster.de/venue/287155", "https://www.tic~
## $ longitude  <dbl> 9.958075, 13.368960, 13.674710, 7.002760, 9.182400, 13.3880~
## $ latitude   <dbl> 53.55188, 52.51007, 51.06498, 51.45813, 48.89121, 52.52394,~
```

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Wirtschafts- und
Sozialwissenschaftliche
Fakultät

LEHRSTUHL FÜR MARKETING
Nauklerstr. 47 72074 Tübingen

5. Visualizing the extracted data

- Below, you can find code that produces a map of Germany. Add points to the map indicating the locations of the event venues across Germany.
- You will find that some coordinates lie way beyond the German borders and can be assumed to be faulty. Set coordinate values to `NA` where the value of `longitude` is outside the range (`5.866944, 15.043611`) or where the value of `latitude` is outside the range (`47.271679, 55.0846`) (these coordinate ranges have been derived from the extreme points of Germany as listed on Wikipedia (see here). For extreme points of other countries, see here).

```
ggplot() +
  geom_polygon(
    aes(x = long, y = lat, group = group), data = map_data("world", region = "Germany"),
    fill = "grey90",color = "black") +
  theme_void() + coord_quickmap() +
  labs(title = "Event locations across Germany", caption = "Source: ticketmaster.com") +
  theme(title = element_text(size=8, face='bold'),
        plot.caption = element_text(face = "italic"))
```

6. Event locations in other countries

- Repeat exercises 2 to 5 for another European country of your choice. (Hint: Clean code pays off! If you have coded the exercises efficiently, only very few adaptions need to be made.)

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Wirtschafts- und
Sozialwissenschaftliche
Fakultät

LEHRSTUHL FÜR MARKETING
Nauklerstr. 47 72074 Tübingen

Note

Submit your assignment by **February 01, 12 noon**. The assignment must include your code, results, and **explanations for code and results**. Submit your assignment via ILIAS using R Markdown (upload both the *.Rmd file **and** a knitted *.html-file). The *.html-file must include a link to your personal GitHub account where the project's version history is documented. Specify the names of the two upload files as follows: `Lastname_Firstname_AssignmentV.Rmd` and `Lastname_Firstname_AssignmentV.html`.

Code of Conduct

You will only successfully complete this assignment if you adhere to the following rules and guidelines. You are allowed to work in groups, however,

- the work submitted must be yours.
- the code must be your own.
- you must make clear whom you worked with.

By submitting your assignment you agree to the above mentioned terms.