Testing Raser

Global adm. boundaries

Using the package `raster` to get admin. boundaries data for Austria.

```
#install.packages("raster")
library(raster)
```

```
## Loading required package: sp
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::extract() masks raster::extract()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x dplyr::select()  masks raster::select()
```

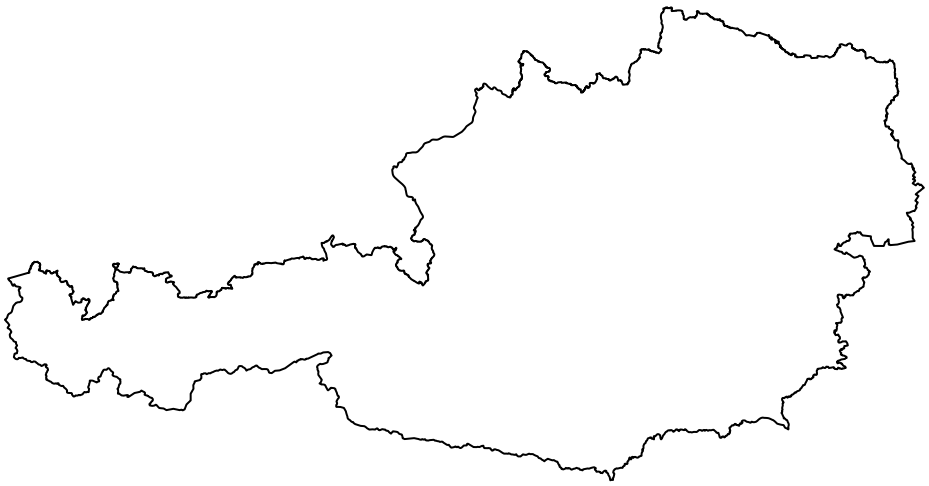
- installs and loads the package

```
austria0 <- getData('GADM', country='AUT', level=0)
```

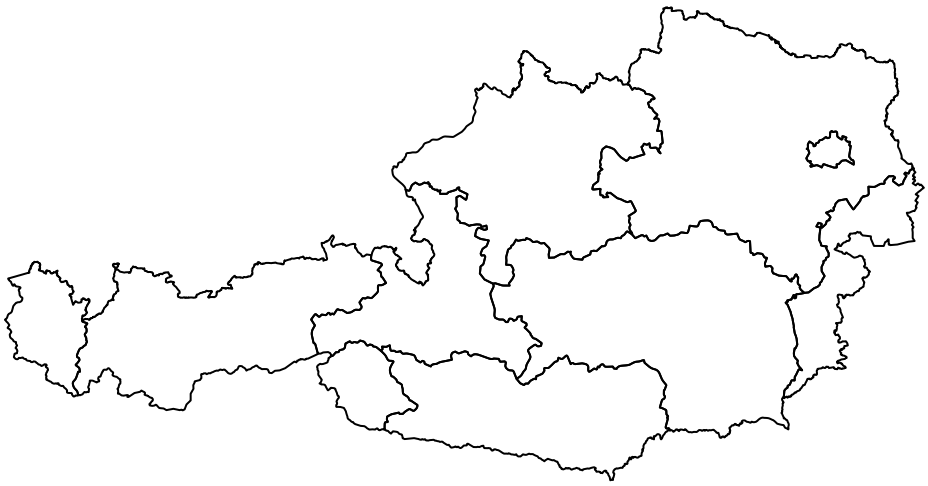
- Arg_1 specifies data set
  - GDAM – global administrative boundaries
- Arg_2 specifies country
  - Use ISO A3 country code
- Arg_3 specifies level of administrative subdivision
  - 0 = country, 1 = state
- Above code returns country boundaries of Austria

```
#Get Data
austria0 <- getData('GADM' , country="AUT", level=0)
austria1 <- getData('GADM' , country="AUT", level=1)

#Plot
#par(mfrow(2,1))
plot(austria0, main="Adm. Boundaries Austria Level 0")
```




```
plot(austria1, main="Adm. Boundaries Austria Level 1")
```



- Code prints Country and State borders for Austria

World Climate

- Similar process for world climate
  - Three arguments
    - Dataset (here, 'worldclim')
    - Variable (here, 'bio', but also 'tmin', 'tmax', 'prec')
    - Resolutaion (0.5, 2.5, 5, and 10 (minutes of a degree).)
      - res=0.5 requires lon + lat

```
climate <- getData('worldclim', var='bio', res=2.5)
```

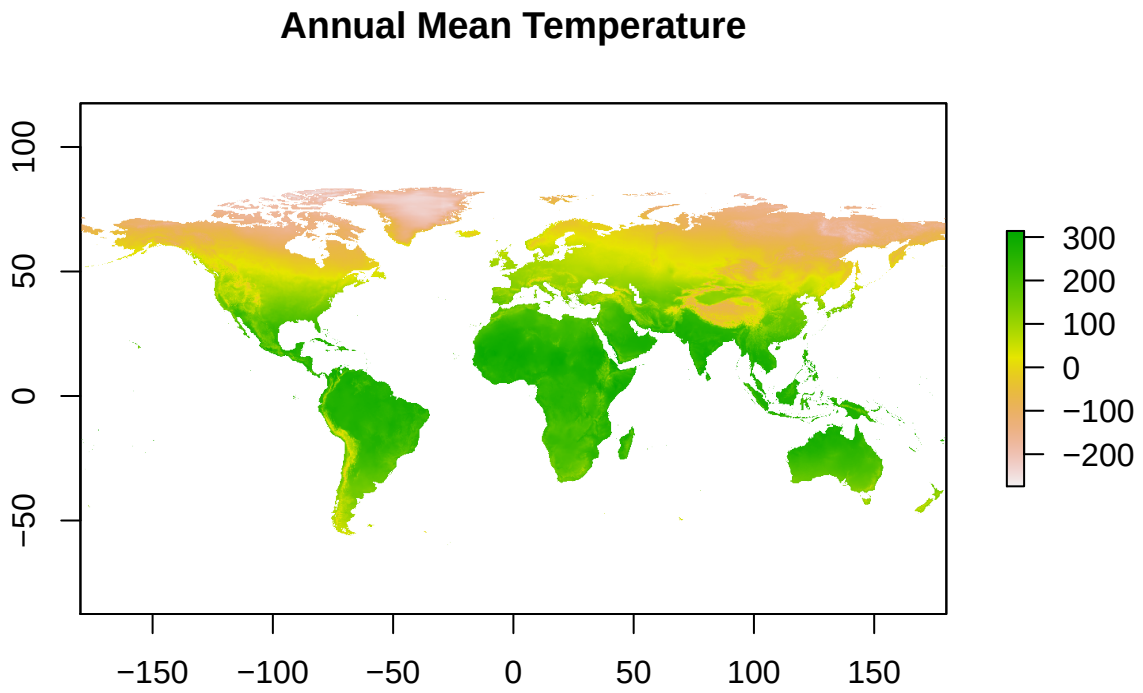
THIS RETURNS A LOT

- BIO1 = Annual Mean Temperature
- BIO2 = Mean Diurnal Range (Mean of monthly (max temp – min temp))
- BIO3 = Isothermality (BIO2/BIO7) (* 100)
- BIO4 = Temperature Seasonality (standard deviation *100)
- BIO5 = Max Temperature of Warmest Month
- BIO6 = Min Temperature of Coldest Month
- BIO7 = Temperature Annual Range (BIO5-BIO6)
- BIO8 = Mean Temperature of Wettest Quarter
- BIO9 = Mean Temperature of Driest Quarter
- BIO10 = Mean Temperature of Warmest Quarter
- BIO11 = Mean Temperature of Coldest Quarter
- BIO12 = Annual Precipitation

- BIO13 = Precipitation of Wettest Month
- BIO14 = Precipitation of Driest Month
- BIO15 = Precipitation Seasonality (Coefficient of Variation)
- BIO16 = Precipitation of Wettest Quarter
- BIO17 = Precipitation of Driest Quarter
- BIO18 = Precipitation of Warmest Quarter
- BIO19 = Precipitation of Coldest Quarter

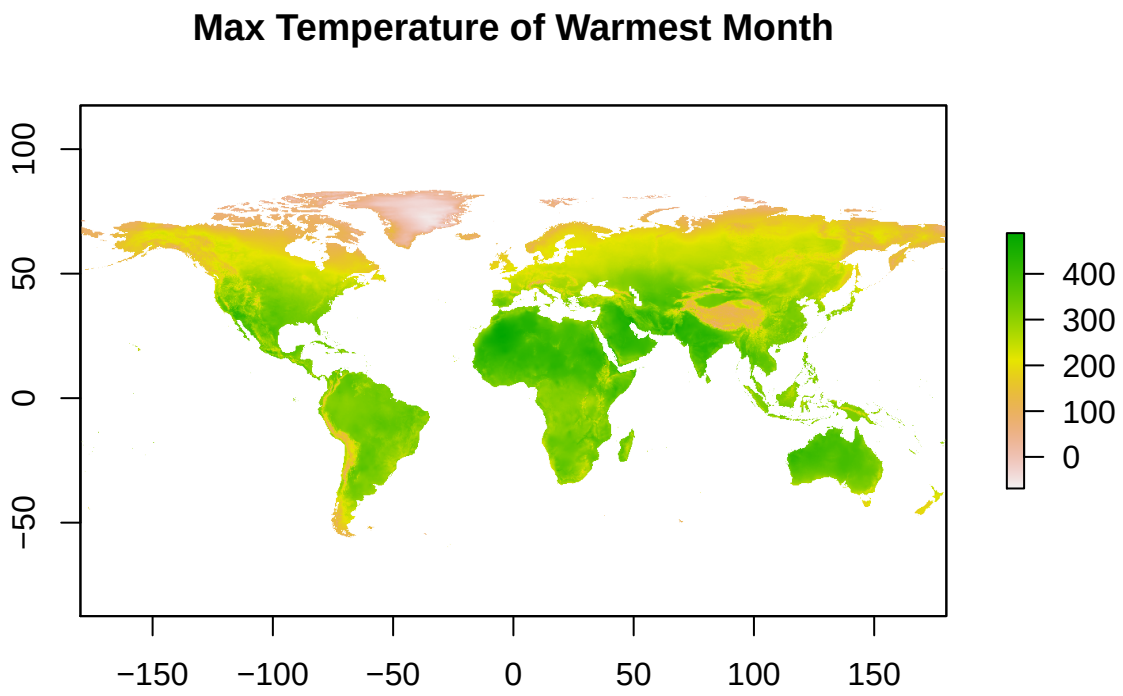
Tutorial goes on to plot the annual mean temp

```
#Plot
plot(climate$bio1, main="Annual Mean Temperature")
```



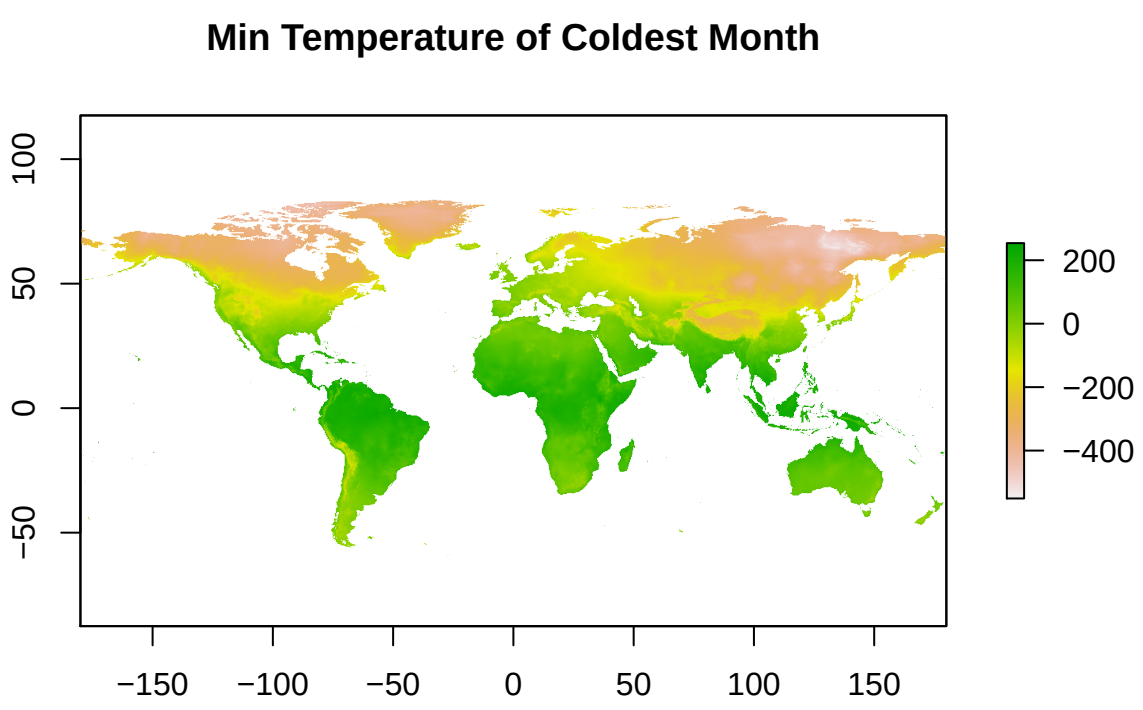
Lets sidebar and do `BIO5 = Max Temperature of Warmest Month`

```
#Plot
plot(climate$bio5, main="Max Temperature of Warmest Month")
```



And BIO6 = Min Temperature of Coldest Month

```
#Plot
plot(climate$bio6, main="Min Temperature of Coldest Month")
```

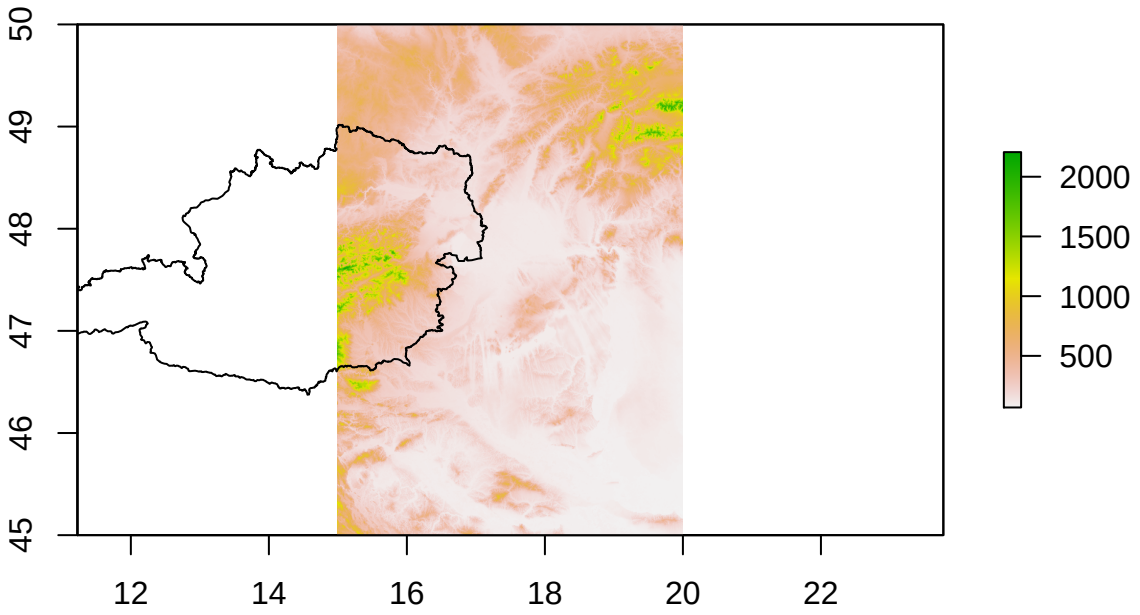


SRTM 90 Elevation

```
srtm <- getData('SRTM', lon=16, lat=48)
```

- Arg_1: returns SRTM 90 data
- Lon: longitude
- Lat: latitude

```
plot(srtm)
plot(austria0, add=TRUE)
```

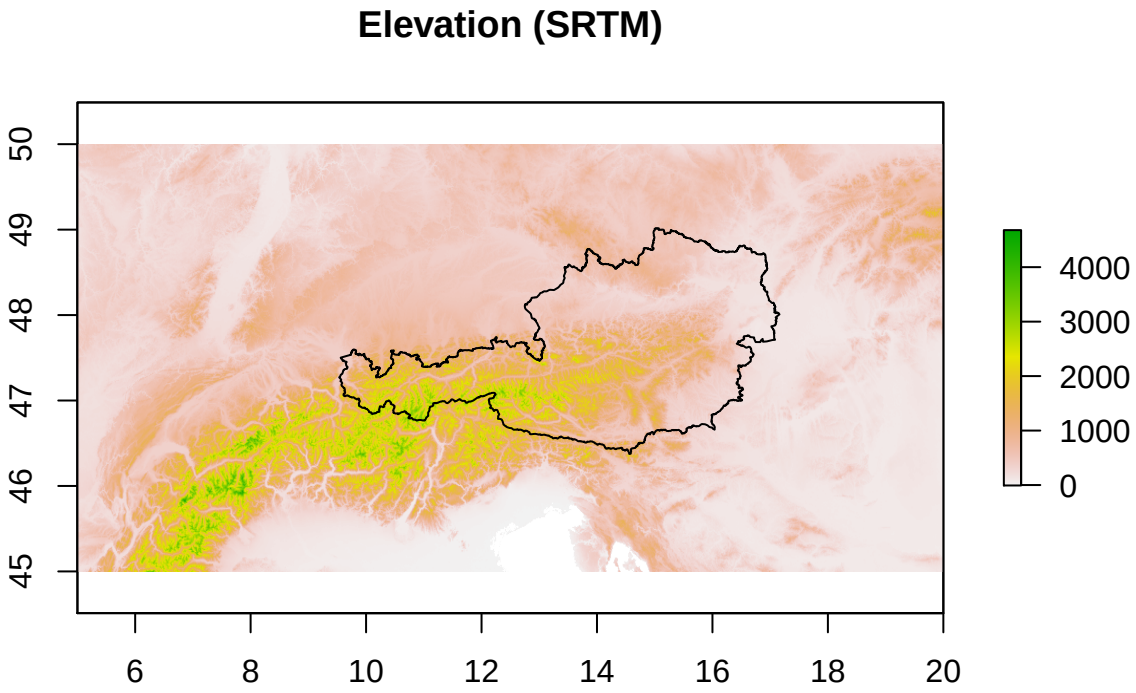


LMAO it's missing a bit of info. Need to tile pieces together.

```
# Download two more tiles
srtm2 <- getData('SRTM', lon=13, lat=48)
srtm3 <- getData('SRTM', lon=9, lat=48)

#Mosaic/merge srtm tiles
srtmmosaic <- mosaic(srtm, srtm2, srtm3, fun=mean)

plot(srtmmosaic, main="Elevation (SRTM)")
plot(austria0, add=TRUE)
```


Elevation (SRTM)
50
49
48
47
46
45
6
8
10
12
14
16
18
20
4000
3000
2000
1000
0