CS 416 Data Visualization

NARRATIVE VISUALIZATION

EXTREME WEATHER EFFECTS ON THE WESTERN STATES 100 ENDURANCE RUN

Photo credit: Hoka One One

Introduction

The Western States Endurance Run (or Western States 100) is the world's oldest 100-mile (100.2 miles) trail ultra-marathon. The Western States Trail originates with the Indigenous People who lived in Utah and moved across Northern Nevada and the Sierra mountains in

California. During the Gold Rush period in California, portions of the trail were used as a passage between the gold mines near Auburn, California, and the silver mines in Nevada. This study will analyze the effects of climate change and extreme weather on finish times from this race, using the official data from 2010 to 2022.

The data from the study comes from official data provided by the race organization, and weather data is integrated by crossing the published weather data with the splits from the race. The data is originally split on a year-by-year basis in the proprietary Excel format. Using an R script (provided in the appendix), I generated a single data CSV file, the one used in the slideshow.

Note: Before more information on the expected format was provided, I created an introduction to the visualization, which I split into two different pages after the rubric was published.

- The introduction (and it links to the slideshow page) is available at: https://ivan.vc/wser
- The interactive slideshow is available at: https://ivan.vc/wser/slideshow.html
- And the source code is at https://github.com/ivanvc/wser.

Messaging

This study analyzes how climate change can impact day-to-day activities. In this case, using the data from the Western States 100 Endurance run, one of the most prestigious hundred miles races in the US.

Narrative Structure

The structure of the narrative visualization follows the interactive slideshow approach. Each scene follows the same template; on the left sidebar, there's a commentary discussing the data on the right chart. However, the user can freely explore the data by changing the filters on the screen's top. Still, the user is guided through the different slides.

Visual Structure

As mentioned above, the template and **structure** used across the scenes is the same, intending to not disorient the user. The controls to navigate the scenes (or slides) is next to the controls that filter the data presented in the charts, so the user can immediately notice that there is a way to modify the displayed data. The left commentary sidebar discusses the displayed chart and aims to help the user navigate the different scenes and to help **highlight** some of the discussed observations, along with the annotations in the charts. It also uses the d3 annotations package to add some messaging on top of the charts. Maintaining the same layout template across all scenes makes the user less likely to be distracted. Keeping the controls in order when possible should help understand how the data in the other scenes are connected, which helps with the **transition**, along with animation when changing slides.

Scenes

The scenes from the narrative visualization are:

1. The distribution of the finishers from the race. I decided to start with this one because it is a good overview of the data that will be analyzed. Without getting yet into details of how the temperature affects the finish time.
2. The finish time vs. finish place curves per year. This chart starts to dig into how the temperature affects the finish time of the runners.
3. Finish rate per year. It concludes with the rate of finishers, with temperature data, which is the topic of the study and shows the effects of extreme temperatures. The user can also select to use the sub-24-hour finishers instead, as explained in the slideshow, as it is a goal many runners aim to.

Annotations

There are two kinds of annotations in the visualization.

1. Tooltips: Although each scene has different data displayed in the tooltip, they all have the same style. Because of the nature of the data and the type of chart displayed, it would be impossible to have the same template. However, emphasis on

trying to keep consistency by using a standard format of font weight, layout, and colors.

2. Annotation callouts: Done by using the d3 annotation library. I kept it with minimal information due to the more extensive commentary on the left sidebar. However, following the same approach of the tooltips, even though the style needs to be different per scene, the user interface is consistent.

These annotations depend on the displayed data, so they are updated if the user selects different filters.

Parameters

The parameters of the overall visualization are all consistent on each slide. They are presented in the same place and, as mentioned before, in the same order when possible. These parameters include:

1. The current slide. The current state is shown in the button element in the controls bar.
2. Year. Filters the data from the selected year, the first and second scenes provide this parameter, and its value is presented in text form next to the slider control.
3. Gender. Filters the data from the selected gender. All the scenes provide this control, and the current value is shown by the radio element in the controls.
4. Temperature. Changes the color to represent the temperature. It could be either high or low. The darker the hue, the more extreme the temperature is (i.e., the darker for the high temperature represents the higher the temperature, and the darker for the low temperature, represents the lower). This control is present in the second and third scenes, kept in the same order.
5. Finish rate vs. sub-24-hour rate. This control is exclusive to the last scene. The current value is shown in the radio control in the controls section. It is shown in the first position to keep the position of the other controls that are consistent with the previous scenes.

Animations are minimal, just when changing the parameters inside a scene.

Triggers

As I mentioned, I tried to keep the controls evident to the user by positioning them in a shared location and not hiding them in the chart itself, so she could interact with the data displayed in each one of the scenes. Because it's a slideshow, and the charts shown are different in each scene, each one clears the annotations from the previous scene, and the animations are limited to changes within the same scene.

Conclusion

Ultimately, I tried to put myself in the end user's shoes. And tried not to obviate anything from the interface. Applying the concepts learned from beginning to end of this course and exploring the different visualization resources shared in the course. That's how I landed with the format from the New York Times, which I used as inspiration, but not copying it, trying to give my personal seal to this work. As mentioned before, I decided to keep the introduction, as I believe I could publish and share this resource with the ultra-running community and the insights from analyzing the effects of climate change and extreme weather on this running race.

References

Western States Endurance Run. *Western States Endurance Run*. Retrieved on July 10th 2022, from: https://www.wser.org/.

Western States Endurance Run. *About Our Research Program — Western States Endurance Run*. Retrieved on July 13th 2022, from: https://www.wser.org/research.

Western States Endurance Run. *How it All Began — Western States Endurance Run*. Retrieved on July 13th 2022, from: https://www.wser.org/how-it-all-began.

Western States Endurance Run. *Snowpack — Western States Endurance Run*. Retrieved on July 15th 2022, from: https://www.wser.org/snowpack.

Western States Endurance Run. *Splits — Western States Endurance Run*. Retrieved on July 15th 2022, from: https://www.wser.org/splits.

Western States Endurance Run. *Weather — Western States Endurance Run*. Retrieved on July 15th 2022, from: https://www.wser.org/weather.

Western States Endurance Run. *Snow Routes — Western States Endurance Run*. Retrieved on July 7th 2022, from: https://www.wser.org/course/snow-routes/.

Appendix

Grading Reference

A. What is the URL of your narrative visualization?

https://ivan.vc/wser/slideshow.html

Before more information on the expected format was provided, I created an introduction to the dataset, which I split into two different pages after the rubric was published. The introduction (and it links to the slideshow page) is available at: https://ivan.vc/wser.

B. Upload a PDF file essay describing your narrative visualization as required by the assignment instructions.

✔️

C. Narrative Structure

3. Does the essay indicate which structure the narrative visualization was designed to follow (martini glass, interactive slide show or drop-down story)?
    a. Yes, it uses the interactive slide show
4. Does the narrative visualization follow that structure?
    a. ✔️

D. Visual Structure

5. Does the essay indicate what visual structure is used for each scene?
    a. ✔️

6. Does the essay indicate how the visual structure ensures the viewer can understand the data?
    a. ✔️

7. Does the essay indicate how highlighting is used to get the viewer to focus on the important parts of the data in each scene?

a. ✔️

8. Does the essay indicate how the visual structure helps the viewer transition to other scenes, to understand how the data connects to the data in other scenes?
    a. ✔️

E. Scenes and Visual Ordering

9. Does the essay identify the scenes of the narrative visualization?
    a. ✔️

10. Does the essay discuss ordering (e.g. the order of elements in a chart or the ordering of scenes)?
    a. ✔️

11. Do the charts used as scenes effectively present the data?
    a. Yes, I think so.

F. Annotations

12. Does the essay discuss annotations?
    a. ✔️

13. Does the essay discuss a template for the annotations?
    a. ✔️

14. Are the annotations in the narrative visualization effective and consistent?
    a. Yes, I think they are effective. They are consistent.

G. Parameters and States

15. Does the essay identify the parameters of the narrative visualization?

a. ✔️

16. Does the essay identify the states of the narrative visualization?
    a. Yes, detailed in each one of the parameters.

17. Does the essay indicate how are the parameters are used to define the state and each scene?
    a. ✔️

18. Does the narrative visualization use parameters to control its state?
    a. ✔️

19. Does the narrative visualization use parameters to control each scene?
    a. ✔️

H. Triggers

20. Does the essay indicate the triggers that connect user actions to changes of state in the narrative visualization?
    a. ✔️

21. Does the essay indicate what affordances are provided to the user to communicate to them what options are available to them in the narrative visualization?
    a. ✔️

22. Does the narrative visualization implement and respond to user events properly?
    a. Yes

23. Does the narrative visualization make any effort at all to communicate what options are available to the user?
    a. ✔️

Data Preparation Script

```
time_to_seconds = function(x) {
  x = as.numeric(x)
  x[1] * 60 * 60 + x[2] * 60 + x[3]
}

seconds_to_time = function(x) {
  h = as.integer(x / 3600)
  m = as.integer(x / 60) - h*60
  s = x %% 60

  paste(sprintf("%02d", h), sprintf("%02d", m), sprintf("%02d", s), sep = ":")
}

seconds_to_time_v = function(vec) {
  sapply(vec, function(x) {
x = strsplit(x, ":")
sapply(x, function(item) {
       item = as.numeric(item)
       item[1]*3600 + item[2]*60 + item[3]
})
  })
}


f = read.csv("./2010.csv")
d = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = f$Snow,
  Snow.Amount = f$Snow.amount,
  River.Crossing = f$River.Crossing,
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gen,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Place.2,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Place.4,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Place.5,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Place.6,
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Place.7,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Place.9,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Place.10,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Place.11,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky..near.),
```

```
  RuckyChucky.Place = f$Place.12,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Place.13,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Place.14,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Place.18,

  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Finish.Place
)

f = read.csv("./2011.csv")
d1 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = f$Snow,
  Snow.Amount = f$Snow.amount,
  River.Crossing = f$River.Crossing,
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = ifelse(nchar(f$State.or.Country) == 2, "United States",
f$State.or.Country),
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Place.2,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Place.4,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Place.5,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Place.6,
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Place.7,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Place.9,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Place.10,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Place.11,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky..near.),
  RuckyChucky.Place = f$Place.12,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Place.13,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Place.14,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Place.17,
  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Place.18
)

f = read.csv("./2012.csv")
d2 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
```

```
  Snow = "None",
  Snow.Amount = 0,
  River.Crossing = "swim",
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gen,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Place.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Place.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Place.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Place.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Place.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Place.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Place.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Place.12,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky..near.),
  RuckyChucky.Place = f$Place.13,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Place.14,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Place.15,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Place.19,
  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Place.20
)

f = read.csv("./2013.csv")
d3 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = "None",
  Snow.Amount = 0,
  River.Crossing = "swim",
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gen,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Place.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Place.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Place.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Place.7,
```

```
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Place.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Place.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Place.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Place.12,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky..near.),
  RuckyChucky.Place = f$Place.13,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Place.14,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Place.15,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Place.19,
  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Place.20
)

f = read.csv("./2014.csv")
d4 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = "None",
  Snow.Amount = 0,
  River.Crossing = "swim",
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky..near.),
  RuckyChucky.Place = f$Pos.14,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.16,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.17,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.21,
  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Pos.22
```

```
)

f = read.csv("./2015.csv")
d5 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = "None",
  Snow.Amount = 0,
  River.Crossing = "swim",
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Pos.14,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.15,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.16,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.20,
  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Pos.21
)

f = read.csv("./2016.csv")
d6 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = "None",
  Snow.Amount = 0,
  River.Crossing = "swim",
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
```

```
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devils.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill.School),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Pos.14,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.15,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.16,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.20,
  Finish.Time = seconds_to_time_v(f$Auburn.Finish.Line),
  Finish.Place = f$Pos.21
)
f = read.csv("./2017.csv")
d7 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = f$Snow,
  Snow.Amount = f$Snow.amount,
  River.Crossing = f$River.Crossing,
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devil.s.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Pos.15,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.16,
```

```
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.17,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.21,
  Finish.Time = seconds_to_time_v(f$Finish),
  Finish.Place = f$Pos.22
)

f = read.csv("./2018.csv")
d8 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = "None",
  Snow.Amount = 0,
  River.Crossing = "swim",
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devil.s.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Pos.15,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.16,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.17,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.21,
  Finish.Time = seconds_to_time_v(f$Finish),
  Finish.Place = f$Pos.22
)

f = read.csv("./2019.csv")
d9 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = f$Snow,
  Snow.Amount = f$Snow.amount,
  River.Crossing = f$River.Crossing,
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
```

```
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devil.s.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Pos.15,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.16,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.17,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.21,
  Finish.Time = seconds_to_time_v(f$Finish),
  Finish.Place = f$Pos.22
)

f = read.csv("./2021.csv")
f$status = ifelse(is.na(f$Pos.21), "DNF", "F")
d10 = data.frame(
  Year = f$year,
  Temp.Low = f$Temp.Low,
  Temp.High = f$Temp.High,
  Snow = f$Snow,
  Snow.Amount = f$Snow.amount,
  River.Crossing = f$River.Crossing,
  Name = paste(f$First.Name, f$Last.Name),
  Gender = f$Gender,
  Age = f$Age,
  Country = f$Country,
  Bib = f$Bib,
  Overall.Place = f$Overall.Place,
  Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Pos.3,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Pos.5,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Pos.6,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Pos.7,
  DevilsThumb.Time = seconds_to_time_v(f$Devil.s.Thumb),
  DevilsThumb.Place = f$Pos.8,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Pos.10,
```

```
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill),
  ForestHillSchool.Place = f$Pos.11,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Pos.13,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Pos.15,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Pos.16,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Pos.17,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Pos.20,
  Finish.Time = seconds_to_time_v(f$Finish),
  Finish.Place = f$Pos.21
)

f = read.csv("./2022.csv")
f$status = ifelse(is.na(f$Position.22), "DNF", "F")
d11 = data.frame(
              Year = 2022,
              Temp.Low = 70,
              Temp.High = 97,
              Snow = "None",
              Snow.Amount = 0,
              River.Crossing = "swim",
              Name = paste(f$First.Name, f$Last.Name),
              Gender = f$Gender,
              Age = f$Age,
              Country = f$Country,
              Bib = f$Bib,
              Overall.Place = f$Overall.Place,
              Finished = f$status,
  DuncanCanyon.Time = seconds_to_time_v(f$Duncan.Canyon),
  DuncanCanyon.Place = f$Position.4,
  MillersDefeat.Time = seconds_to_time_v(f$Miller.s.Defeat),
  MillersDefeat.Place = f$Position.6,
  DustyCorners.Time = seconds_to_time_v(f$Dusty.Corners),
  DustyCorners.Place = f$Position.7,
  LastChance.Time = seconds_to_time_v(f$Last.Chance),
  LastChance.Place = f$Position.8,
  DevilsThumb.Time = seconds_to_time_v(f$Devil.s.Thumb),
  DevilsThumb.Place = f$Position.9,
  MichiganBluff.Time = seconds_to_time_v(f$Michigan.Bluff),
  MichiganBluff.Place = f$Position.11,
  ForestHillSchool.Time = seconds_to_time_v(f$Foresthill),
  ForestHillSchool.Place = f$Position.12,
  Peachstone.Time = seconds_to_time_v(f$Peachstone..Cal.2.),
  Peachstone.Place = f$Position.14,
  RuckyChucky.Time = seconds_to_time_v(f$Rucky.Chucky),
  RuckyChucky.Place = f$Position.16,
  GreenGate.Time = seconds_to_time_v(f$Green.Gate),
  GreenGate.Place = f$Position.17,
  AuburnLakeTrails.Time = seconds_to_time_v(f$Auburn.Lake.Trails),
  AuburnLakeTrails.Place = f$Position.18,
  RobiePoint.Time = seconds_to_time_v(f$Robie.Point),
  RobiePoint.Place = f$Position.21,
  Finish.Time = seconds_to_time_v(f$Finish),
  Finish.Place = f$Position.22
)

f[f$Country == 'USA', ]$Country = 'United States'
```

```
f[f$Country == 'FRA', ]$Country = 'France'
f[f$Country == 'NZL', ]$Country = 'New Zealand'
f[f$Country == 'CAN', ]$Country = 'Canada'
f[f$Country == 'CHE', ]$Country = 'Czech Republic'
f[f$Country == 'ZWE', ]$Country = 'Zimbabwe'
f[f$Country == 'ZAF', ]$Country = 'South Africa'
f[f$Country == 'ESP', ]$Country = 'Spain'
f[f$Country == 'IRL', ]$Country = 'Ireland'
f[f$Country == 'POL', ]$Country = 'Poland'
f[f$Country == 'JPN', ]$Country = 'Japan'
f[f$Country == 'SWE', ]$Country = 'Sweden'
f[f$Country == 'HKG', ]$Country = 'Hong Kong'
f[f$Country == 'AUS', ]$Country = 'Australia'
f[f$Country == 'SVK', ]$Country = 'Slovakia'
f[f$Country == 'BGR', ]$Country = 'Bulgaria'
f[f$Country == 'HRV', ]$Country = 'Croatia'
f[f$Country == 'ITA', ]$Country = 'Italy'
f[f$Country == 'SVN', ]$Country = 'Slovenia'
f[f$Country == 'CRI', ]$Country = 'Costa Rica'
f[f$Country == 'PER', ]$Country = 'Peru'
f[f$Country == 'MEX', ]$Country = 'Mexico'
f[f$Country == 'BRA', ]$Country = 'Brazil'
f[f$Country == 'AUT', ]$Country = 'Austria'
f[f$Country == 'DEU', ]$Country = 'Germany'
f[f$Country == 'DNK', ]$Country = 'Denmark'
f[f$Country == 'IDN', ]$Country = 'Indonesia'
f[f$Country == 'SGP', ]$Country = 'Singapore'
f[f$Country == 'BEL', ]$Country = 'Belgium'
f[f$Country == 'NOR', ]$Country = 'Norway'
f[f$Country == 'SLV', ]$Country = 'El Salvador'
f[f$Country == 'GB',  ]$Country = 'UK'
f[f$Country == 'GBR', ]$Country = 'UK'

ws100 = rbind(d, d1, d2, d3, d4, d5, d6, d7, d8, d9, d10, d11)
ws100[ws100$Country == "UK", ]$Country = "United Kingdom"
ws100$Gender = as.factor(ws100$Gender)
ws100$Snow = as.factor(ws100$Snow)
ws100$Finished = as.factor(ws100$Finished)
ws100$Year = as.factor(ws100$Year)
ws100$Country = as.factor(ws100$Country)
ws100$River.Crossing = as.factor(ws100$River.Crossing)
ws100$Finish.Place = as.factor(ws100$Finish.Place)
str(ws100)
write.csv(ws100, file="../ws100.csv", row.names = F)
```