Alligator Population

By James Elander, Mai Houa Hang, Mee Thao, and Susan Thao

Project Outline

The aim of our project is to explore alligator populations, sightings, and human-related incidents like attacks. Using interactive visualizations, we aim to highlight trends in alligator behavior and interactions with humans, helping raise awareness about alligators in the United States.

Questions We Are Looking to Answer:

Are there any population distribution trends that we are seeing with alligators in Florida?

Is there an increase of alligator attacks over time? Where in the United States are the alligator attacks or human-related incidents occurring most frequently?

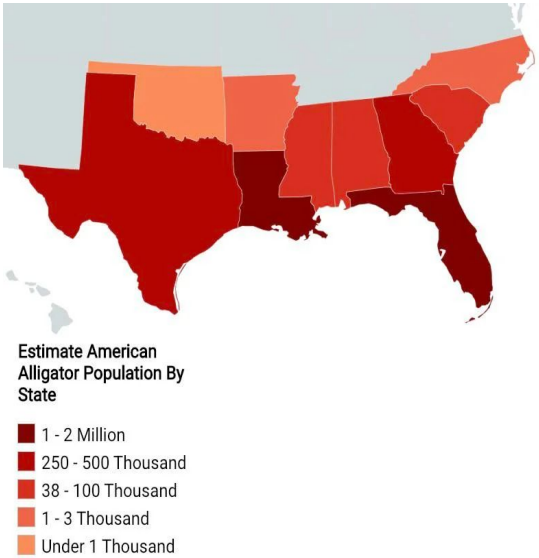
Has alligator sightings decreased or increased over time in the United States?

What are the average age and gender of alligator attack victims?

Alligator Population Distribution

- I wanted to see how the alligator population was distributed throughout the state

- On the right, is the population distribution throughout the US



Data Challenges for population distribution

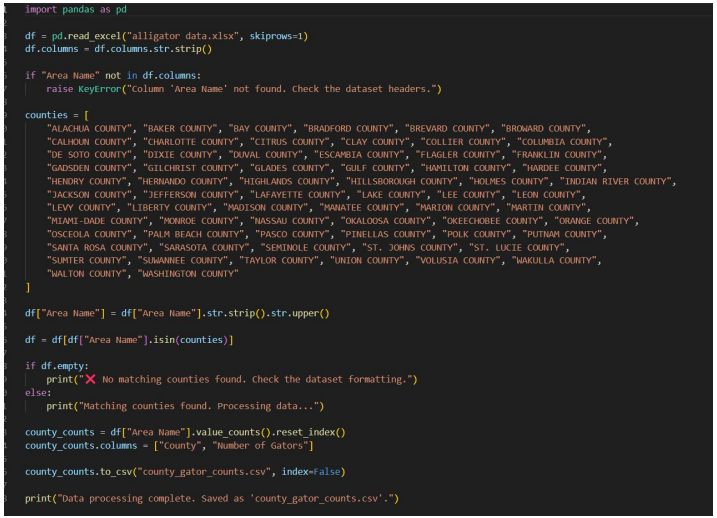
- Raw data of alligator population county by county in Florida was difficult to obtain
- Data of alligator sightings are often inaccurate and unreliable

Gator Harvest as proxy

- The Florida Fish and Wildlife Conservation Commission kills alligators for population control and publishes their data
- I used the FWC's alligator harvest data as an indication of population distribution across counties

Data Issues

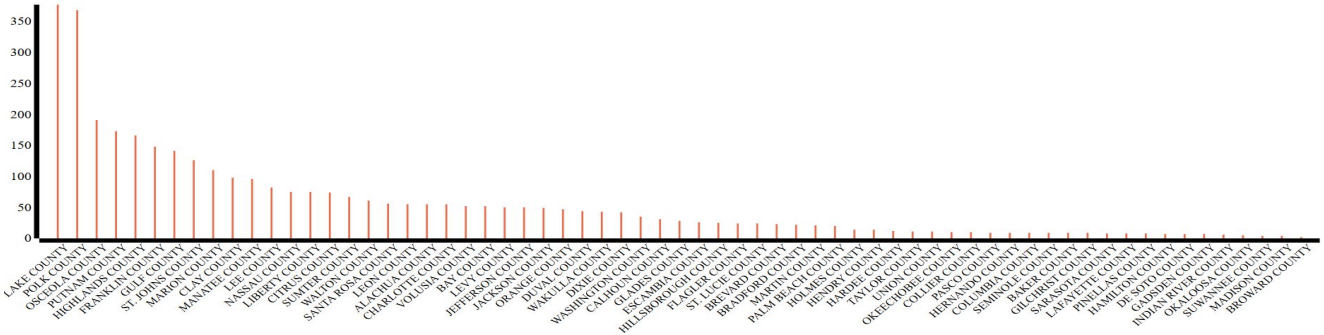
- The dataset included data for counties, but also for non county entities like lakes and ponds
- I used python to clean up the data before proceeding



```python
import pandas as pd

df = pd.read_excel("alligator data.xlsx", skiprows=1)
df.columns = df.columns.str.strip()

if "Area Name" not in df.columns:
    raise KeyError("Column 'Area Name' not found. Check the dataset headers.")

counties = [
    "ALACHUA COUNTY", "BAKER COUNTY", "BAY COUNTY", "BRADFORD COUNTY", "BREVARD COUNTY", "BROWARD COUNTY",
    "CALHOUN COUNTY", "CHARLOTTE COUNTY", "CITRUS COUNTY", "CLAY COUNTY", "COLLIER COUNTY", "COLUMBIA COUNTY",
    "DE SOTO COUNTY", "DIXIE COUNTY", "DUVAL COUNTY", "ESCAMBIA COUNTY", "FLAGLER COUNTY", "FRANKLIN COUNTY",
    "GADSDEN COUNTY", "GILCHRIST COUNTY", "GLADES COUNTY", "GULF COUNTY", "HAMILTON COUNTY", "HARDEE COUNTY",
    "HENDRY COUNTY", "HERNANDO COUNTY", "HIGHLANDS COUNTY", "HILLSBOROUGH COUNTY", "HOLMES COUNTY", "INDIAN RIVER COUNTY",
    "JACKSON COUNTY", "JEFFERSON COUNTY", "LAFAYETTE COUNTY", "LAKE COUNTY", "LEE COUNTY", "LEON COUNTY",
    "LEVY COUNTY", "LIBERTY COUNTY", "MADISON COUNTY", "MANATEE COUNTY", "MARION COUNTY", "MARTIN COUNTY",
    "MIAMI-DADE COUNTY", "MONROE COUNTY", "NASSAU COUNTY", "OKALOOSA COUNTY", "OKEECHOBEE COUNTY", "ORANGE COUNTY",
    "OSCEOLA COUNTY", "PALM BEACH COUNTY", "PASCO COUNTY", "PINELLAS COUNTY", "POLK COUNTY", "PUTNAM COUNTY",
    "SANTA ROSA COUNTY", "SARASOTA COUNTY", "SEMINOLE COUNTY", "ST. JOHNS COUNTY", "ST. LUCIE COUNTY",
    "SUMTER COUNTY", "SUWANNEE COUNTY", "TAYLOR COUNTY", "UNION COUNTY", "VOLUSIA COUNTY", "WAKULLA COUNTY",
    "WALTON COUNTY", "WASHINGTON COUNTY"
]

df["Area Name"] = df["Area Name"].str.strip().str.upper()

df = df[df["Area Name"].isin(counties)]

if df.empty:
    print("❌ No matching counties found. Check the dataset formatting.")
else:
    print("Matching counties found. Processing data...")

county_counts = df["Area Name"].value_counts().reset_index()
county_counts.columns = ["County", "Number of Gators"]

county_counts.to_csv("county_gator_counts.csv", index=False)

print("Data processing complete. Saved as 'county_gator_counts.csv'.")
```

```
document.addEventListener("DOMContentLoaded", function () {
    d3.csv("county_gator_counts.csv", function (error, data) {
        if (error) {
            console.error("❌ Error loading CSV:", error);
            return;
        }

        console.log("✅ CSV Loaded Successfully!", data);

        var margin = { top: 50, right: 50, bottom: 220, left: 100 }, // space for labels
            width = 1600 - margin.left - margin.right, // width for spacing
            height = 600 - margin.top - margin.bottom; //height for better visibility

        var svg = d3.select("#bar-chart-container")
            .append("svg")
            .attr("width", width + margin.left + margin.right)
            .attr("height", height + margin.top + margin.bottom)
            .append("g")
            .attr("transform", "translate(" + margin.left + "," + margin.top + ")");

        var x = d3.scale.ordinal()
            .domain(data.map(function (d) { return d.County; }))
            .rangeBands([0, width], 0.9); //spacing between bars

        var y = d3.scale.linear()
            .domain([0, d3.max(data, function (d) { return +d["Number of Gators"]; })])
            .range([height, 0]);

        var xAxis = d3.svg.axis().scale(x).orient("bottom");
        var yAxis = d3.svg.axis().scale(y).orient("left");

        svg.append("g")
            .attr("transform", "translate(0," + height + ")")
            .call(xAxis)
            .selectAll("text")
            .attr("transform", "rotate(-45)")
            .style("text-anchor", "end")
            .style("font-size", "14px"); // Bigger font for readability

        svg.append("g").call(yAxis);

        svg.selectAll(".bar")
            .data(data)
            .enter()
            .append("rect")
            .attr("class", "bar")
            .attr("x", function (d) { return x(d.County); })
            .attr("y", function (d) { return y(+d["Number of Gators"]); })
            .attr("width", x.rangeBand())
            .attr("height", function (d) { return height - y(+d["Number of Gators"]); })
            .attr("fill", "#e76f51");
    });
});
```

Results



I found that alligators were the most densely populated around the Orlando metro area. This makes sense because alligators are freshwater animals.
[Alligator bar graph]

Has Alligator Sightings Increase or Decreased Over Time?

A brief overview of the coding used to create an interactive heatmap

```
// Initialize the map
initMap();

// Define the initMap function
function initMap() {
    const vs = new ol.source.Vector();

    const heatmapLayer = new ol.layer.Heatmap({
        source: vs,
        radius: 5,
        blur: 10,
        weight: feature => feature.get('weight') || 0.5
    });

    const map = new ol.Map({
        layers: [
            new ol.layer.Tile({ source: new ol.source.OSM() }),
            heatmapLayer
        ],
        target: 'map',
        view: new ol.View({
            center: ol.proj.fromLonLat([-90, 30]),
            zoom: 6
        })
    });
```

First, I created a map using the Openlayers library

Next, I loaded my data that came in a CSV file and process it

```
// Load the data from CSV and process it
Papa.parse('inaturalist-alligator-data.csv', {
    download: true,
    header: true,
    dynamicTyping: true,
    complete: ({ data }) => {
        sightings = data
            .filter(({ latitude, longitude }) => latitude && longitude)
            .map(({ latitude, longitude, observed_on }) => ({
                lat: latitude,
                lon: longitude,
                year: new Date(observed_on).getFullYear(),
                month: new Date(observed_on).getMonth() + 1
            }));
```

cript_Mee.js

Has Alligator Sightings Increase or Decreased Over Time?

Continuation

I created filters, in this case, dropdowns with checkboxes for users to be able to select the years and months they want to view on the heatmap

```
// Populate the year dropdown with checkboxes
function popYrDropdown(ranges) {
    const dropdown = document.getElementById('year-range-dropdown');
    ranges.forEach(range => {
        const checkboxContainer = document.createElement('div');
        checkboxContainer.className = 'checkbox-item';

        const checkbox = document.createElement('input');
        checkbox.type = 'checkbox';
        checkbox.value = range;
        checkbox.id = range;
        checkbox.checked = true;

        const label = document.createElement('label');
        label.setAttribute('for', range);
        label.textContent = range;

        checkboxContainer.append(checkbox, label);
        dropdown.appendChild(checkboxContainer);
    });
}
```

```
months.forEach((month, index) => {
    const checkboxContainer = document.createElement('div');
    checkboxContainer.className = 'checkbox-item';

    const checkbox = document.createElement('input');
    checkbox.type = 'checkbox';
    checkbox.value = index + 1;
    checkbox.id = month;
    checkbox.checked = true;

    const label = document.createElement('label');
    label.setAttribute('for', month);
    label.textContent = month;

    checkboxContainer.append(checkbox, label);
    dropdown.appendChild(checkboxContainer);
});
```

Has Alligator Sightings Increase or Decreased Over Time?

Continuation

I then created a function to filter the data to show only the sightings that match those selections.

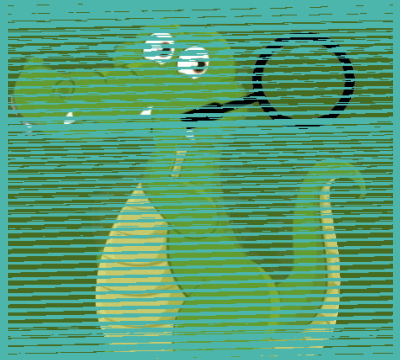
The heatmap

```
// Function to update the heatmap with filtered sightings
function updHeatmap(filteredSightings) {
    vs.clear();
    const counts = filteredSightings.reduce((acc, { lat, lon }) => {
        const key = `${lat},${lon}`;
        acc[key] = (acc[key] || 0) + 1;
        return acc;
    }, {});

    filteredSightings.forEach(({ lat, lon }) => {
        const coords = ol.proj.fromLonLat([lon, lat]);
        vs.addFeature(new ol.Feature({
            geometry: new ol.geom.Point(coords),
            weight: Math.min(0.5, counts[`${lat},${lon}`] / 10)
        }));
```

```
// Function to filter the sightings based on the selected years and months
function fils() {
    const selectedYears = Array.from(document.querySelectorAll('#year-range-dropdown input:checked'))
        .map(checkbox => checkbox.value);
    const selectedMonths = Array.from(document.querySelectorAll('#month-dropdown input:checked'))
        .map(checkbox => parseInt(checkbox.value));

    const filteredSightings = sightings.filter(({ year, month }) =>
        selectedYears.some(range => {
            const [start, end] = range.split('-').map(Number);
            return year >= start && year <= end;
        }) && selectedMonths.includes(month)
```

Lastly I create a function to update the heatmap with the filtered data

Findings



Alligator sightings have grown significantly over time. Per the heatmap it has been growing steady from 2013-2017 and onwards.

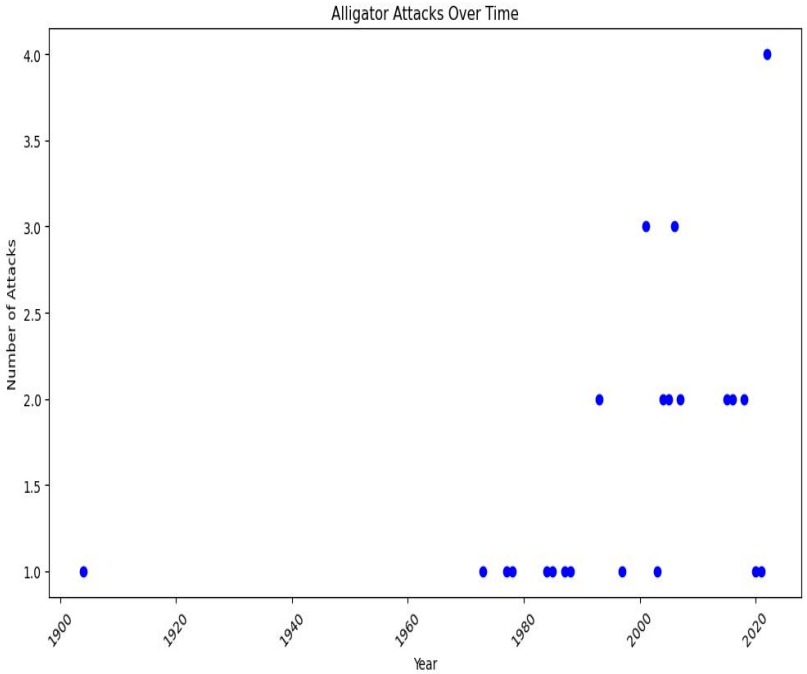
The top states with the most alligator sighting are Florida, Texas, South Carolina, Louisiana, and Georgia.

There is a lot of alligator sightings during spring time. Around March to May the sightings for alligators increase and then in June to August, the sighting decrease. In October to December, the alligator sightings increase again.

Is there an increase of alligator attacks over time?

```
import pandas as pd
import matplotlib.pyplot as plt
```

```
# Load the data
Data = pd.read_csv('fatal_alligator_attacks_US.csv')

# Convert the 'Date' column to datetime format
Date = pd.to_datetime(data['Date'], errors='coerce')

# Extract the year from the 'Date' column
Year = data['Date'].dt.year

# Group the data by 'Year' and count the number of attacks per year
attack_count_per_year = data['Year'].value_counts().sort_index()

# Plot the data as a scatter plot
plt.figure(figsize=(10,6))
plt.scatter(attack_count_per_year.index, attack_count_per_year.values, color='blue')

# Add labels and title
plt.title('Alligator Attacks Over Time')
plt.xlabel('Year')
plt.ylabel('Number of Attacks')
plt.xticks(rotation=45)
plt.tight_layout()

# Show the plot
plt.show()
```



Where in the United States are the alligator attacks or human related incidents occurring most frequently?

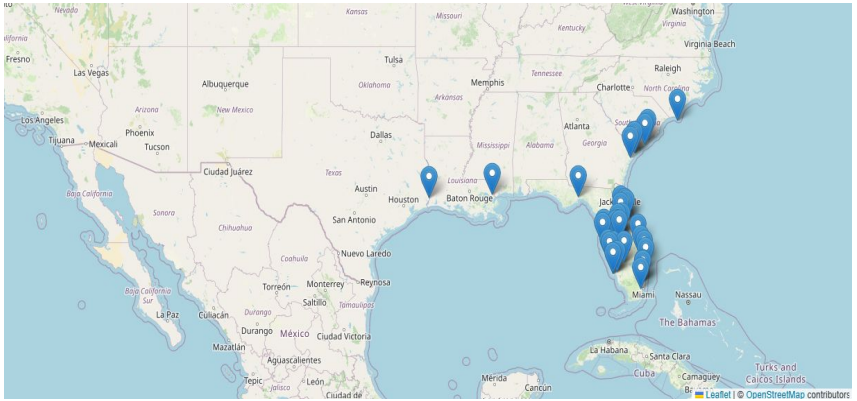
index.html

JavaScript

Alligator Attack Map

```
// Initialize the map to the USA only
const map = L.map('map').setView([30, -95], 5);

// Add OpenStreetMap tiles
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
    attribution: '&copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors'
}).addTo(map);

// Geocode (latitude, longitude, name)
const attackData = [
    {
        name: "Nancy Becker",
        location: "Hilton Head Island, South Carolina",
        lat: 32.216,
        lon: -80.752,
        details: "The victim fell into a pond in a gated community in Hilton Head Island, South Carolina,
    },
    {
        name: "Rose Marie Wiegand",
        location: "Englewood, Florida",
        lat: 26.9805,
        lon: -82.3483,
        details: "The victim fell into a pond along a golf course in Englewood, Florida, and was attacked
    },
    {
        name: "Micheal Burstein",
        location: "Myrtle Beach, South Carolina",
        lat: 33.6803,
        lon: -78.0332,
        details: "The victim was dragged into a pond in Myrtle Beach, South Carolina. The victim's body wa
    },
    {   name: "Sean McGuinness",
        location: "Largo, Florida",
        lat: 27.9093,
        lon: -82.7766,
        details: "The victim was attacked while trying to retrieve frisbee golf discs in the lake at John
    },
```

```
// Create markers from the attack data
function createMarkers() {
    attackData.forEach(function(attack) {
        const marker = L.marker([attack.lat, attack.lon]).addTo(map);
        marker.bindPopup(`
            <b>${attack.name}</b><br>
            Location: ${attack.location}<br>
            Details: ${attack.details}
        `);
    });
}

// Create the markers on the map
createMarkers();
```



Alligator Attack Victim Data

Alligator Attack Victim Data

Date	Name	Age	Sex	Details
August 15, 2022	Nancy Becker	88	female	The victim fell into a pond in a gated community in Hilton Head Island, South Carolina, and was attacked by a nearly 10-foot-long alligator. The victim's body was found being guarded by the alligator and was later recovered.
July 15, 2022	Rose Marie Wiegand	80s	female	The victim fell into a pond along a golf course in Englewood, Florida, and was attacked by two alligators.
June 24, 2022	Micheal Burstein	75	male	The victim was dragged into a pond in Myrtle Beach, South Carolina. The victim's body was recovered later and a subsequent post-mortem established that Mr Burstein drowned.
May 31, 2022	Sean McGuinness	47	male	The victim was attacked while trying to retrieve frisbee golf discs in the lake at John S. Taylor Park in Largo, Florida.
August 30, 2021	Timothy Satterlee	71	male	The victim was attacked in floodwaters of Hurricane Ida near the city of Slidell, Louisiana. The attack was witnessed by the victim's wife, who said the alligator bit his arm off.
May 1, 2020	Cynthia Covert	58	female	The victim was pulled under and drowned by an alligator in a pond behind a home near Salt Cedar Lane, Kiawah Island, South Carolina.
August 15, 2022	Nancy Becker	88	female	The victim fell into a pond in a gated community in Hilton Head Island, South Carolina, and was attacked by a nearly 10-foot-long alligator. The victim's body was found being guarded by the alligator and was later recovered.
July 15, 2022	Rose Marie Wiegand	80s	female	The victim fell into a pond along a golf course in Englewood, Florida, and was attacked by two alligators.
June 24, 2022	Micheal Burstein	75	male	The victim was dragged into a pond in Myrtle Beach, South Carolina. The victim's body was recovered later and a subsequent post-mortem established that Mr Burstein drowned.
May 31, 2022	Sean McGuinness	47	male	The victim was attacked while trying to retrieve frisbee golf discs in the lake at John S. Taylor Park in Largo, Florida.

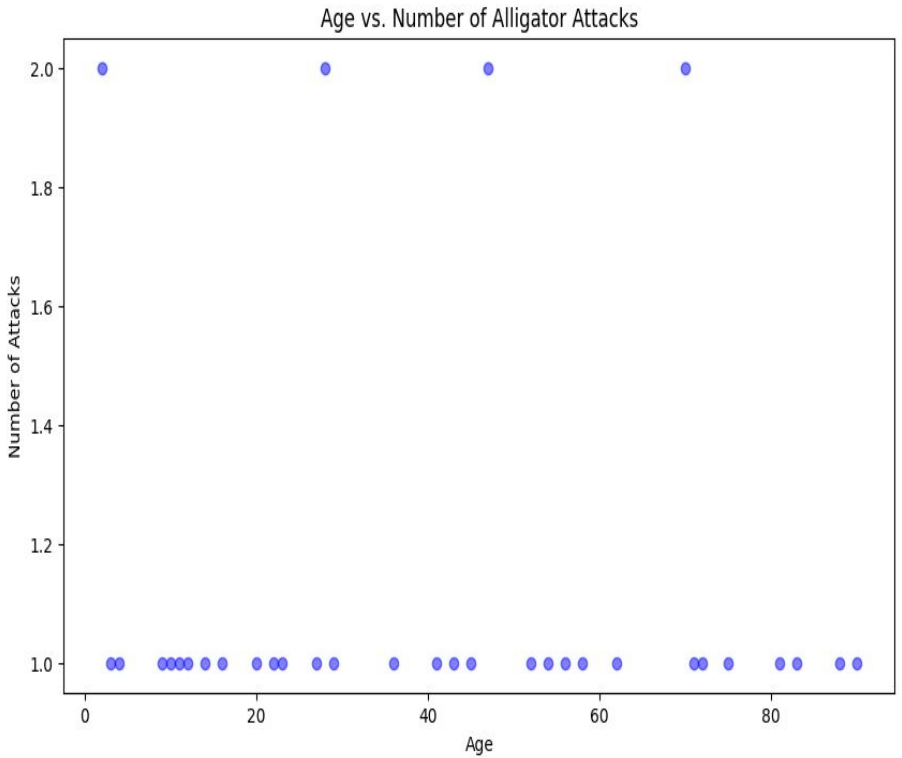
```
// Function to populate the table with the data
function populateTable() {
    const tableBody = document.querySelector("#victimTable tbody");
    tableBody.innerHTML = '';  // Clear any existing rows

    data.forEach(victim => {
        // Create a row for each victim
        const row = document.createElement("tr");

        // Create cells for each piece of data
        const dateCell = document.createElement("td");
        dateCell.textContent = victim.Date;
        row.appendChild(dateCell);

        const nameCell = document.createElement("td");
        nameCell.textContent = victim.Name;
        row.appendChild(nameCell);

        const ageCell = document.createElement("td");
        ageCell.textContent = victim.Age;
        row.appendChild(ageCell);

        const sexCell = document.createElement("td");
        sexCell.textContent = victim.Sex;
        row.appendChild(sexCell);

        const detailsCell = document.createElement("td");
        detailsCell.textContent = victim.Details;
        row.appendChild(detailsCell);

        // Add the row to the table
        tableBody.appendChild(row);
    });
}

// Populate the table when the page loads
window.onload = populateTable;
script>
```

Average age and gender of alligator attack victims


```
// Create the bar chart
const ctx = document.getElementById('genderAgeChart').getContext('2d');
const genderAgeChart = new Chart(ctx, {
    type: 'bar',
    data: {
        labels: ['Male', 'Female'],
        datasets: [{
            label: 'Average Age of Victims',
            data: [averages.male, averages.female],
            backgroundColor: ['rgba(54, 162, 235, 0.2)', 'rgba(255, 99, 132, 0.2)'],
            borderColor: ['rgba(54, 162, 235, 1)', 'rgba(255, 99, 132, 1)'],
            borderWidth: 1
        }]
    },
    options: {
        responsive: true,
        scales: {
            y: {
                beginAtZero: true,
                title: {
                    display: true,
                    text: 'Average Age'
                }
            }
        }
    }
});
```

Scatter plot of age and number of attacks (age)

```
import pandas as pd
import matplotlib.pyplot as plt
# Step 1: Load the CSV file into a DataFrame
data = pd.read_csv('fatal_alligator_attacks_US.csv')
# Step 2: Strip any leading or trailing spaces from column names
data.columns = data.columns.str.strip()
# Step 3: Convert the 'Age' column to numeric, handling errors (e.g., '80s'
to NaN)
data['Age'] = pd.to_numeric(data['Age'], errors='coerce')
# Step 4: Drop rows where 'Age' or 'Details' (indicating an attack) is
missing
data_cleaned = data.dropna(subset=['Age', 'Details'])
# Step 5: Count the number of attacks per age (assuming each row is an
attack)
# We'll assume each row represents a unique attack. So, for plotting, we can
just use 'Age' and count them.
attack_counts_by_age = data_cleaned.groupby('Age').size()
# Step 6: Plot the data
plt.figure(figsize=(10, 6))
plt.scatter(attack_counts_by_age.index, attack_counts_by_age.values,
color='blue', alpha=0.5)
# Add titles and labels
plt.title('Age vs. Number of Alligator Attacks')
plt.xlabel('Age')
plt.ylabel('Number of Attacks')
# Show the plot
plt.show()
```



Questions?