



Gotta Cache 'Em All



# Pokémon Hunters

Tiffany Harris

➤ [tiffanyharris711](#)

Jessica Chester

➤ [jessches5](#)

Bao Pham

➤ [baodphm](#)

Tom Munson

➤ [TJM-TGC](#)



**GitHub**

# Pokémon

## The Evolution

- Pokemon APIs
- Python
- SQLite
- Flask App
- HTML
- Javascript





## The Extraction

```
In [1]: ➜ import requests
import json
import pandas as pd
import sqlite3
# youtube vid for help https://youtu.be/LYh8ih2X50o

In [2]: ➜ #url for api documentation https://pogoapi.net/documentation/
stub_url = "https://pogoapi.net"
endpoints = {
    "pokemon_names": "/api/v1/pokemon_names.json",
    "pokemon_types": "/api/v1/pokemon_types.json",
    "pokemon_stats": "/api/v1/pokemon_stats.json",
    "fast_moves": "/api/v1/fast_moves.json",
    "charged_moves": "/api/v1/charged_moves.json",
    "current_pokemon_moves": "/api/v1/current_pokemon_moves.json",
}

url_list = []
for key in endpoints:
    url_list.append(stub_url + endpoints[key])
```

The team extracted relevant Pokémon data through numerous API calls from <https://pogoapi.net>.

**Pokemon Names - JSON to Data Frame**

```
In [3]: print(url_list[0])
https://pogodata.net/api/v1/pokemon_names.json

In [4]: url_response = requests.get(f"{url_list[0]}").json() #first endpoint, pokemon_names
url_response
```

```
Out[4]: [{"1": {"id": 1, "name": "Bulbasaur"}, "2": {"id": 2, "name": "Ivysaur"}, "3": {"id": 3, "name": "Venusaur"}, "4": {"id": 4, "name": "Charmander"}, "5": {"id": 5, "name": "Charmeleon"}, "6": {"id": 6, "name": "Charizard"}, "7": {"id": 7, "name": "Squirtle"}, "8": {"id": 8, "name": "Wartortle"}, "9": {"id": 9, "name": "Blastoise"}, "10": {"id": 10, "name": "Caterpie"}, "11": {"id": 11, "name": "Metapod"}, "12": {"id": 12, "name": "Butterfree"}, "13": {"id": 13, "name": "Weedle"}, "14": {"id": 14, "name": "Beedrill"}, "15": {"id": 15, "name": "Beedrill"}, "16": {"id": 16, "name": "Pidgey"}, "17": {"id": 17, "name": "Pidgeotto"}, "18": {"id": 18, "name": "Pidgeot"}, "19": {"id": 19, "name": "Rattata"}, "20": {"id": 20, "name": "Raticate"}]

In [5]: id_list = []
name_list = []
for i in range(1,len(url_response)+1):
    id_list.append(url_response[str(i)][['id']])
    name_list.append(url_response[str(i)][['name']])

In [6]: complete_list = [id_list,name_list]
df_names = pd.DataFrame(complete_list).transpose()
df_names.columns = ['id','pokemon_name']
df_names.head()
```

```
Out[6]:
   id  pokemon_name
0   1      Bulbasaur
1   2      Ivysaur
2   3      Venusaur
3   4     Charmander
4   5     Charmeleon
```

**Pokemon Forms and Types - JSON to Data Frame**

```
In [7]: print(url_list[1])
https://pogodata.net/api/v1/pokemon_types.json

In [8]: url_response = requests.get(f"{url_list[1]}").json() #second endpoint, pokemon_types (also has forms)
```

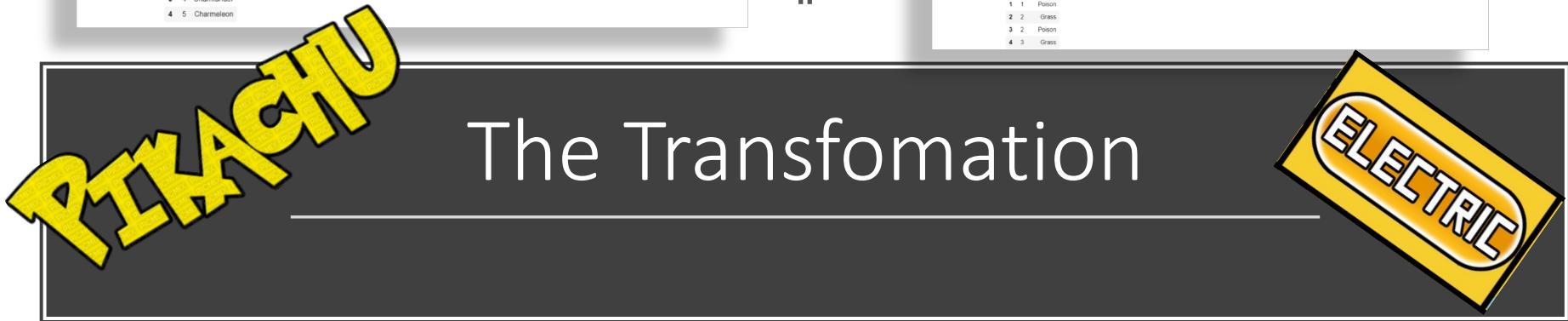
```
Out[8]: [{"form": "Falt_2019", "pokemon_id": 1, "pokemon_name": "Bulbasaur", "type": ["Grass", "Poison"]}, {"form": "Normal", "pokemon_id": 1, "pokemon_name": "Bulbasaur", "type": ["Grass", "Poison"]}, {"form": "Purified", "pokemon_id": 1, "pokemon_name": "Bulbasaur", "type": ["Grass", "Poison"]}, {"form": "Shadow", "pokemon_id": 1, "pokemon_name": "Bulbasaur", "type": ["Grass", "Poison"]}, {"form": "Normal", "pokemon_id": 2, "pokemon_name": "Ivysaur", "type": ["Grass", "Poison"]}, {"form": "Normal", "pokemon_id": 3, "pokemon_name": "Venusaur", "type": ["Grass", "Poison"]}, {"form": "Normal", "pokemon_id": 4, "pokemon_name": "Charmander", "type": ["Fire", "Poison"]}, {"form": "Normal", "pokemon_id": 5, "pokemon_name": "Charmeleon", "type": ["Fire", "Poison"]}, {"form": "Normal", "pokemon_id": 6, "pokemon_name": "Charizard", "type": ["Fire", "Poison"]}, {"form": "Normal", "pokemon_id": 7, "pokemon_name": "Squirtle", "type": ["Water", "Poison"]}, {"form": "Normal", "pokemon_id": 8, "pokemon_name": "Wartortle", "type": ["Water", "Poison"]}, {"form": "Normal", "pokemon_id": 9, "pokemon_name": "Blastoise", "type": ["Water", "Poison"]}, {"form": "Normal", "pokemon_id": 10, "pokemon_name": "Caterpie", "type": ["Bug", "Poison"]}, {"form": "Normal", "pokemon_id": 11, "pokemon_name": "Metapod", "type": ["Bug", "Poison"]}, {"form": "Normal", "pokemon_id": 12, "pokemon_name": "Butterfree", "type": ["Bug", "Poison"]}, {"form": "Normal", "pokemon_id": 13, "pokemon_name": "Weedle", "type": ["Bug", "Poison"]}, {"form": "Normal", "pokemon_id": 14, "pokemon_name": "Beedrill", "type": ["Bug", "Poison"]}, {"form": "Normal", "pokemon_id": 15, "pokemon_name": "Beedrill", "type": ["Bug", "Poison"]}, {"form": "Normal", "pokemon_id": 16, "pokemon_name": "Pidgey", "type": ["Normal", "Poison"]}, {"form": "Normal", "pokemon_id": 17, "pokemon_name": "Pidgeotto", "type": ["Normal", "Poison"]}, {"form": "Normal", "pokemon_id": 18, "pokemon_name": "Pidgeot", "type": ["Normal", "Poison"]}, {"form": "Normal", "pokemon_id": 19, "pokemon_name": "Rattata", "type": ["Normal", "Poison"]}, {"form": "Normal", "pokemon_id": 20, "pokemon_name": "Raticate", "type": ["Normal", "Poison"]}], [{"id": 1, "type": "Grass"}, {"id": 1, "type": "Poison"}, {"id": 2, "type": "Grass"}, {"id": 2, "type": "Poison"}, {"id": 3, "type": "Grass"}, {"id": 3, "type": "Poison"}, {"id": 4, "type": "Fire"}, {"id": 4, "type": "Poison"}, {"id": 5, "type": "Fire"}, {"id": 5, "type": "Poison"}, {"id": 6, "type": "Fire"}, {"id": 6, "type": "Poison"}, {"id": 7, "type": "Water"}, {"id": 7, "type": "Poison"}, {"id": 8, "type": "Water"}, {"id": 8, "type": "Poison"}, {"id": 9, "type": "Water"}, {"id": 9, "type": "Poison"}, {"id": 10, "type": "Bug"}, {"id": 10, "type": "Poison"}, {"id": 11, "type": "Bug"}, {"id": 11, "type": "Poison"}, {"id": 12, "type": "Bug"}, {"id": 12, "type": "Poison"}, {"id": 13, "type": "Bug"}, {"id": 13, "type": "Poison"}, {"id": 14, "type": "Bug"}, {"id": 14, "type": "Poison"}, {"id": 15, "type": "Bug"}, {"id": 15, "type": "Poison"}, {"id": 16, "type": "Normal"}, {"id": 16, "type": "Poison"}, {"id": 17, "type": "Normal"}, {"id": 17, "type": "Poison"}, {"id": 18, "type": "Normal"}, {"id": 18, "type": "Poison"}, {"id": 19, "type": "Normal"}, {"id": 19, "type": "Poison"}, {"id": 20, "type": "Normal"}, {"id": 20, "type": "Poison"}]
```

```
In [9]: df = pd.DataFrame()
for i in range(1, len(url_response)):
    url = url_list[i]
    url_response[i] = df.append(pd.DataFrame(url), ignore_index = True)
df.head()
```

```
Out[9]:
   form  pokemon_id  pokemon_name  type
0  Falt_2019         1      Bulbasaur  Grass
1  Falt_2019         1      Bulbasaur  Poison
2  Normal            1      Bulbasaur  Grass
3  Normal            1      Bulbasaur  Poison
4  Purified          1      Bulbasaur  Grass
```

```
In [10]: df_type = pd.DataFrame(df_type)
columns = ['pokemon_id', 'type']
df_type = df_type[['pokemon_id', 'type']]
df_type = df_type.drop_duplicates().reset_index(drop=True)
df_type = df_type.rename(columns={'pokemon_id': 'id', 'type': 'poketype'})
df_type.head()
```

```
Out[10]:
   id  poketype
0   1      Grass
1   1      Poison
2   2      Grass
3   2      Poison
4   3      Grass
```



# LOADING

## Insert DataFrames to SQL Lite as Database Tables

```
conn = sqlite3.connect('pokemonGo.sqlite')
c = conn.cursor()
try:
    c.execute('DROP TABLE pokemon_names')
    c.execute('DROP TABLE pokemon_forms')
    c.execute('DROP TABLE pokemon_types')
    c.execute('DROP TABLE pokemon_base_stats')
    c.execute('DROP TABLE pokemon_fast_moves')
    c.execute('DROP TABLE pokemon_charged_moves')
    c.execute('DROP TABLE pokemon_fast_move_stats')
except:
    pass

try:
    #create db table to hold pokemon id and names
    c.execute('''CREATE TABLE pokemon_names(
        id integer PRIMARY KEY,
        pokename text
    )''')

    #create db table to hold pokemon forms
    c.execute('''CREATE TABLE pokemon_forms(
        autoid integer PRIMARY KEY,
        id integer,
        pokeform text
    )''')

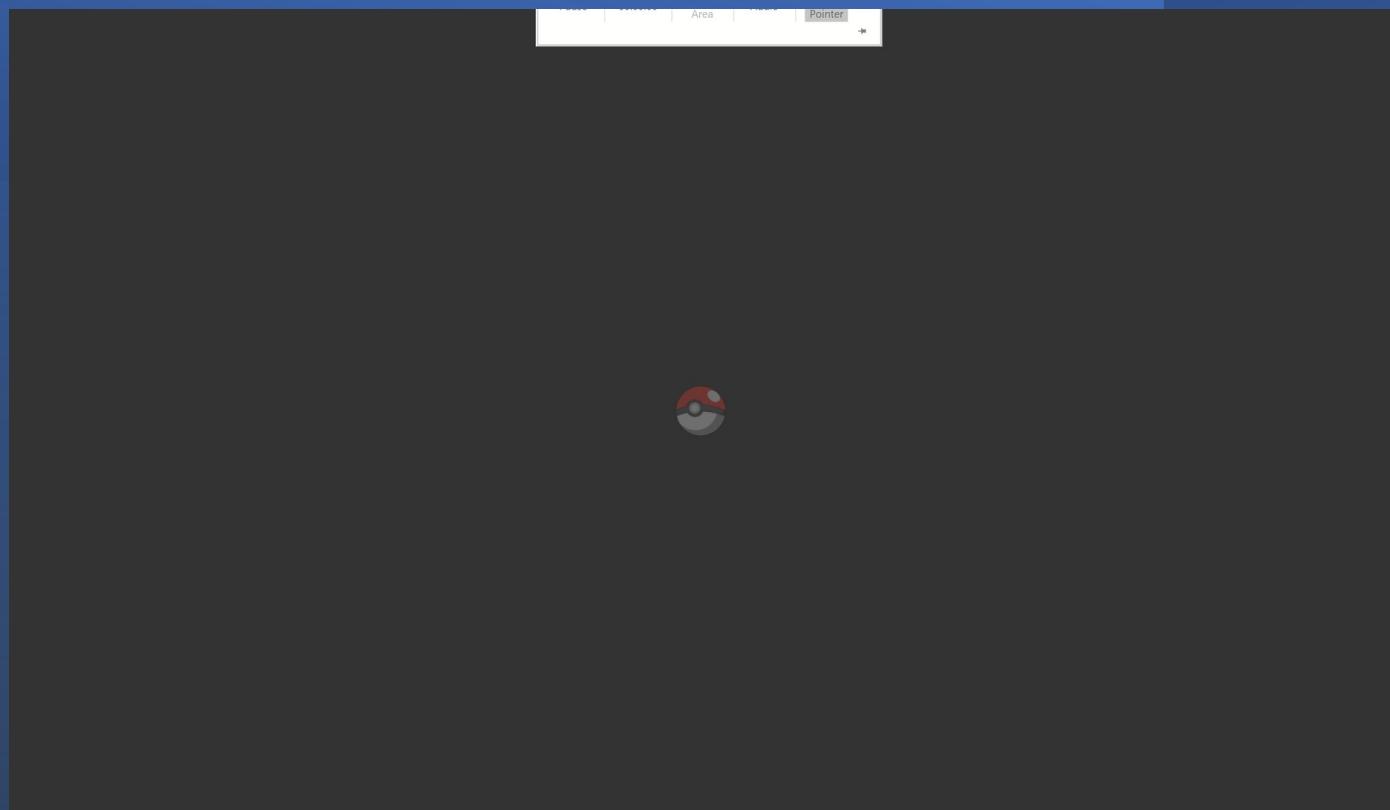
    #create db table to hold pokemon types
    c.execute('''CREATE TABLE pokemon_types(
        autoid integer PRIMARY KEY,
        id integer,
        poketype text
    )''')

    #create db table to hold pokemon base stats
    c.execute('''CREATE TABLE pokemon_base_stats(
        autoid integer PRIMARY KEY,
        id integer,
        pokeform text,
        base_attack integer,
        base_defense integer,
        base_stamina integer
    )''')

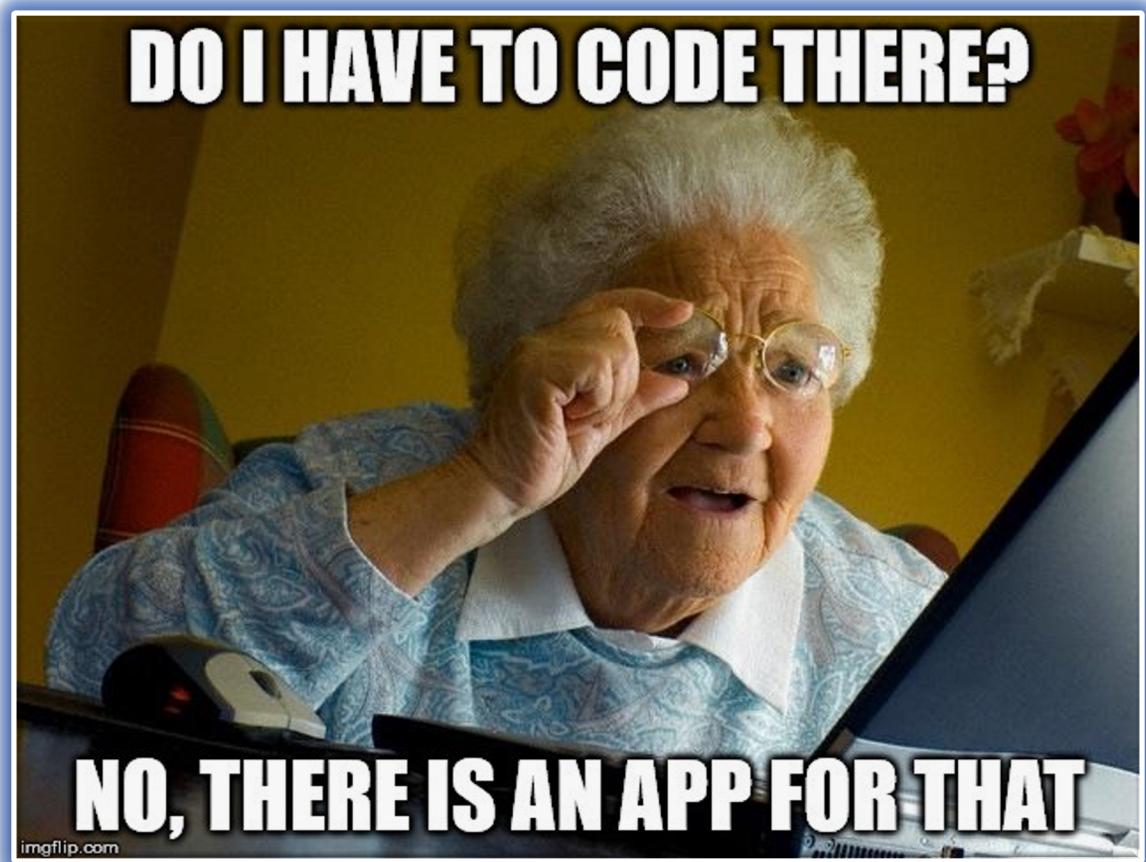
    #create db table to hold pokemon fast moves
    c.execute('''CREATE TABLE pokemon_fast_moves(
        autoid integer PRIMARY KEY,
        id integer,
        fast_moves text,
        form text
    )''')

    #create db table to hold pokemon charged moves
    c.execute('''CREATE TABLE pokemon_charged_moves(
        autoid integer PRIMARY KEY,
        id integer,
        charged_moves text,
        form text
    )''')

    #create db table to hold pokemon fast move stats
    c.execute('''CREATE TABLE pokemon_fast_move_stats(
        duration integer,
        attack float,
        defense float,
        speed float,
        special_attack float,
        special_defense float
    )''')
except:
    pass
```



# Creating the Full Stack App



```

Start of container -->
<div class="container-fluid">
  <section class="row">
    <div class="col-md-2">
      <aside class="filters">
        <div class="panel panel-default">
          <div class="panel-heading">Filter Search</div>
          <div class="panel-body">
            <div class="form-group">
              <ul class="list-group" id="filters">
                <li class="filter list-group-item">
                  <label for="characterid">Select a Character</label>
                  <select id="selDataset" onchange="init()">
                    <option value="">Select a Character</option>
                  </select>
                </li>
              </ul>
            </div>
          </div>
        </div>
      </aside>
    </div>
  </section>
</div>

```

```

//-----initial function to load the webpage, called at the very end-----
function init() {
  var queryUrl = "/api/v1/names";
  var selector = d3.select("#selDataset");

  selector.html("");

  d3.json(queryUrl).then((data) => {
    var pokeList = data;
    var pokeAlpha = pokeList.sort();
    var pokeNames = [];

    for (var i = 0; i < pokeAlpha.length; i++) {
      var pokeID = pokeAlpha[i][0];
      var pokeName = pokeAlpha[i][1];
      pokeNames.push(pokeName + "-" + pokeID);
    }

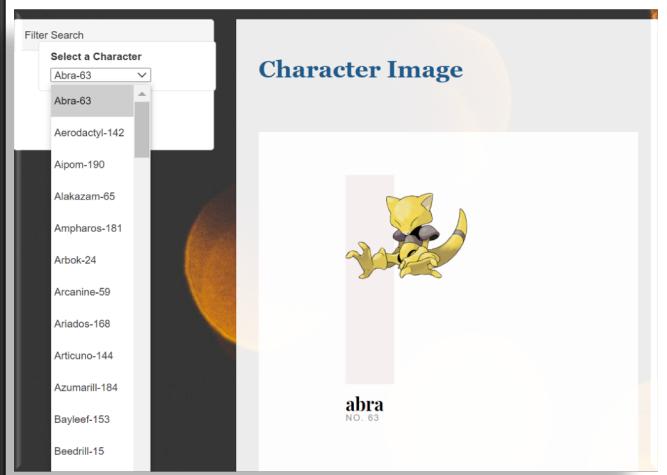
    pokeNames.sort();

    pokeNames.forEach((pokemon) => {
      selector.append("option").text(pokemon).property("value", pokemon);
    });

    var firstID = 460;
    getCharDesc(firstID);
    getCharImage(firstID);
    getCharType(firstID);
    getCharMoves(firstID);
    getBaseStats(firstID);
  });
}

init();

```



# The Dropdown

---

## Featured Pokémon

Once the user selects a Pokémon Go character, character-specific information will populate on the screen.

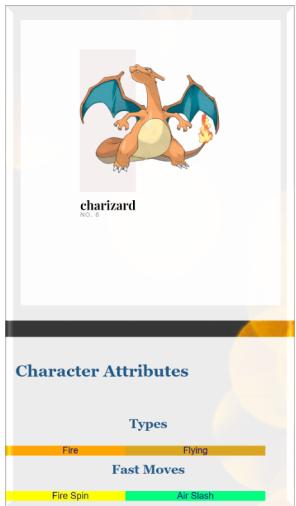


```
function getCharImage(pokecharID) {
  var charImageStubURL = "https://pokeres.bastionbot.org/images/pokemon/";
  var charImageURL = charImageStubURL + pokecharID + ".png";

  d3.select("#character-image")
    .append("img")
    .attr("src", charImageURL)
    .attr("width", "40%")
    .attr("id", "char-image");
}

//-----get character description from API-----
function getCharDesc(pokecharID) {
  var charDescStubURL = "https://pokeapi.co/api/v2/ability/";
  var charDescURL = charDescStubURL + pokecharID;

  d3.json(charDescURL)
    .then((data) => {
      var charDesc = data.effect_entries[1].effect;
      d3.select("#character-description")
        .append("div")
        .attr("id", "char-desc")
        .append("p")
        .text(charDesc);
    })
    .catch(function (error) {
      var charDesc = "No description found for this Pokémon character.";
      d3.select("#character-description")
        .append("div")
        .attr("id", "char-desc")
        .append("p")
        .text(charDesc);
    });
}
```



```
//-----get character type from API-----//
function getCharType(pokecharID) {
  var queryUrl = "/api/v1/types";
  var int_pokeCharID = parseInt(pokecharID);
  let filteredData = [];
  let typeList = [];
  typeList.length = 0;

  d3.json(queryUrl).then((data) => {
    filteredData.push(data.filter((type) => type[0] === int_pokeCharID));

    filteredData[0].forEach((pokemon) => {
      typeList.push(pokemon[1]);
    });

    // console.log(int_pokeCharID);
    // console.log(typeList);

    typeList.forEach((type) => {
      d3.select("#character-type")
        .append("div")
        .attr("id", "char-type")
        .append("h4")
        .attr("class", "col-md-6 row")
        .text(type);
    });
  });
}
```

```
//-----get character moves from API-----//
function getCharMoves(pokecharID) {
  var queryUrl = "/api/v1/fast_moves";
  var int_pokeCharID = parseInt(pokecharID);

  let filteredMoves = [];
  let moveList = [];
  moveList.length = 0;

  d3.json(queryUrl).then((data) => {
    filteredMoves.push(data.filter((move) => move[0] === int_pokeCharID));

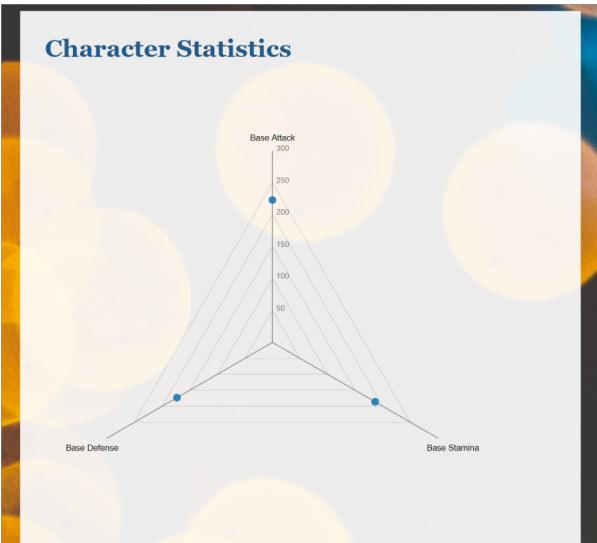
    filteredMoves[0].forEach((pokemon) => {
      moveList.push(pokemon[1]);
    });

    moveList.forEach((move) => {
      d3.select("#character-moves")
        .append("h4")
        .attr("id", "char-move")
        .attr("class", "col-md-6 row")
        .text(move);
    });
  });
}
```



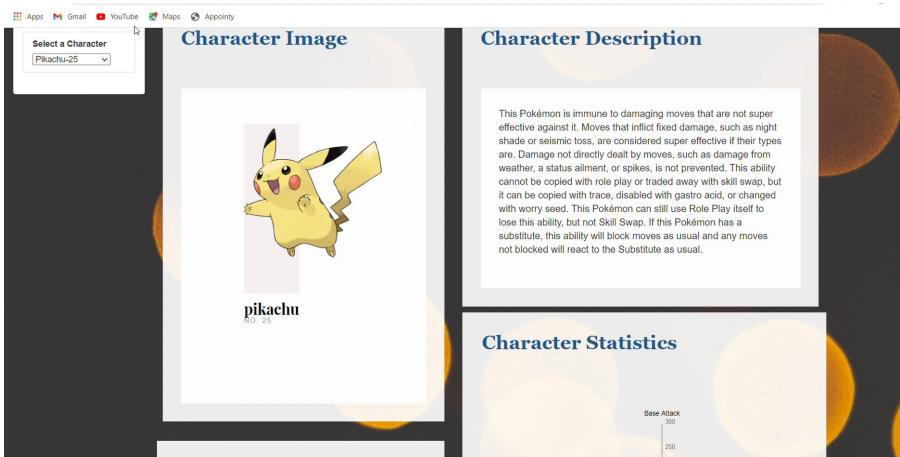
# Pokémon Attributes

# Pokémon Statistics



```
//-----build radar graph with character base stats-----//  
  
function getBaseStats(pokecharID) {  
  var queryUrl = "/api/v1/base_stats";  
  var int_pokeCharID = parseInt(pokecharID);  
  
  let filteredStats = [];  
  let statsList = [];  
  statsList.length = 0;  
  
  d3.json(queryUrl).then((data) => [  
    filteredStats.push(  
      data.filter((stat) => stat[0] === int_pokeCharID && stat[1] === "Normal")  
    );  
  
    var base_attack = filteredStats[0][0][2];  
    var base_defense = filteredStats[0][0][3];  
    var base_stamina = filteredStats[0][0][4];  
  
    var w = 500,  
        h = 500;  
  
    var colorscale = d3.scaleOrdinal(d3.schemeCategory10);  
  
    var LegendOptions = ["Pokemon Name"];  
  
    var d = [  
      [  
        { axis: "Base Attack", value: base_attack },  
        { axis: "Base Defense", value: base_defense },  
        { axis: "Base Stamina", value: base_stamina },  
      ],  
    ];  
  
    //Options for the Radar chart, other than default  
    var mycfg = {  
      w: w,  
      h: h,  
      maxValue: 300,  
      levels: 6,  
      ExtrawidthX: 300,  
    };  
  ]};  
  
function getBaseStats(pokecharID) {  
  var queryUrl = "/api/v1/base_stats";  
  var int_pokeCharID = parseInt(pokecharID);  
  
  let filteredStats = [];  
  let statsList = [];  
  statsList.length = 0;  
  
  d3.json(queryUrl).then((data) => [  
    filteredStats.push(  
      data.filter((stat) => stat[0] === int_pokeCharID && stat[1] === "Normal")  
    );  
  
    var base_attack = filteredStats[0][0][2];  
    var base_defense = filteredStats[0][0][3];  
    var base_stamina = filteredStats[0][0][4];  
  
    var w = 500,  
        h = 500;  
  
    var colorscale = d3.scaleOrdinal(d3.schemeCategory10);  
  
    var LegendOptions = ["Pokemon Name"];  
  
    var d = [  
      [  
        { axis: "Base Attack", value: base_attack },  
        { axis: "Base Defense", value: base_defense },  
        { axis: "Base Stamina", value: base_stamina },  
      ],  
    ];  
  
    //Options for the Radar chart, other than default  
    var mycfg = {  
      w: w,  
      h: h,  
      maxValue: 300,  
      levels: 6,  
      ExtrawidthX: 300,  
    };  
  ]};
```

# Anime.js



```
        }
        initEvents() {
            this.DOM.product.addEventListener('click', () => this.open());
        }
        open() {
            this.info.img = d3.select("#Pogo").attr('src')
            this.info.title = d3.select(".product__title").text()
            this.info.subtitle = d3.select(".product__subtitle").text()
            this.info.price = d3.select(".product__price").text()
            // this.info.description =d3.select(".product_description").text()
            DOM.details.fill(this.info);
            DOM.details.open([
                productBg: this.DOM.productBg,
                productImg: this.DOM.productImg
            ]);
        }
    }
```

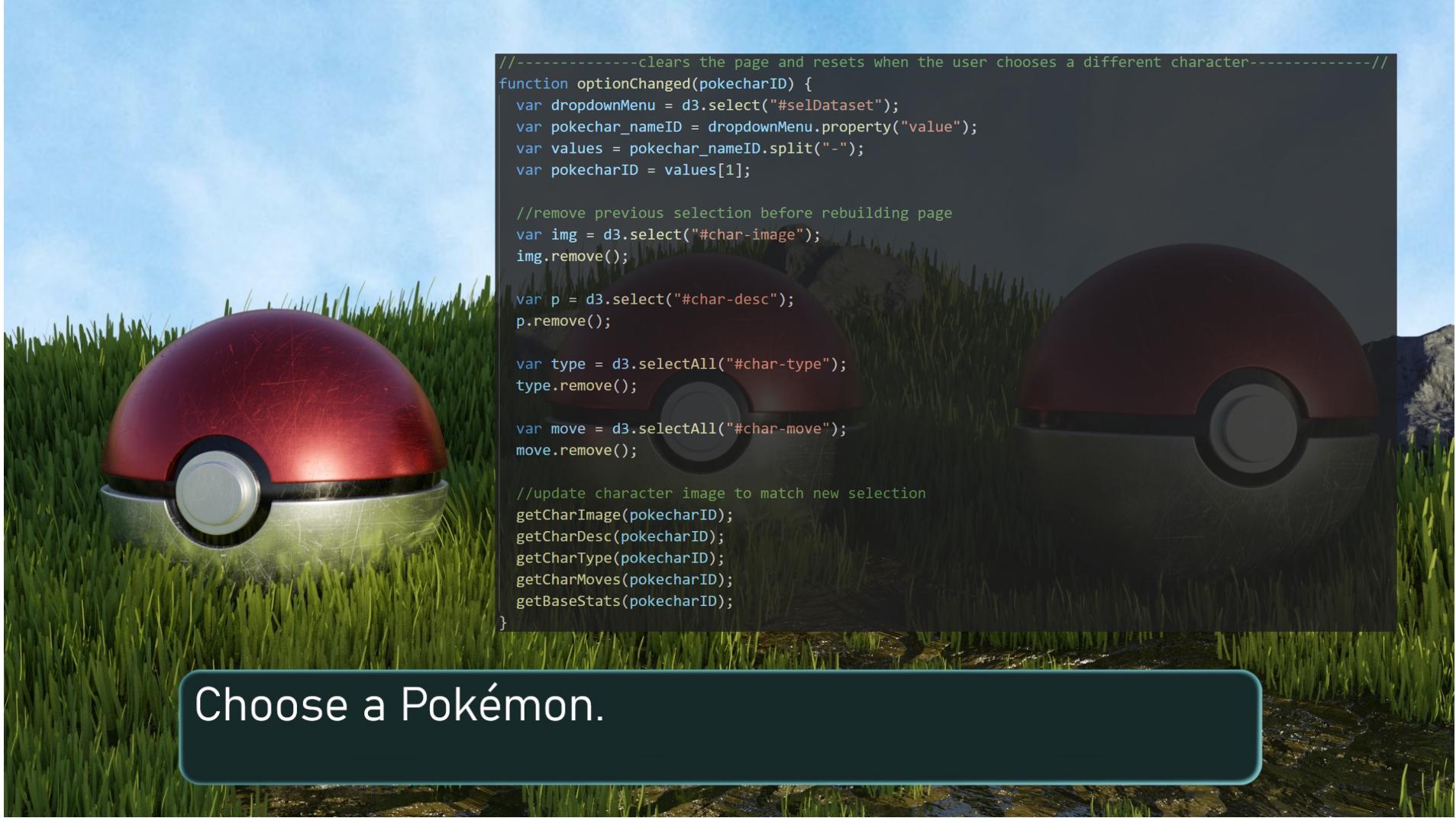
```
44     function getCharInfo(pokecharID) {
45         var charInfoStubURL = "https://pokeapi.co/api/v2/pokemon/";
46         var charInfoURL = charInfoStubURL + pokecharID;
47
48         d3.json(charInfoURL)
49             .then((data) => {
50                 console.log('getcharinfo data',data)
51                 var charDesc = data.name;
52                 d3.select('.product__title')
53                     // .append('div')
54                     .attr('id', 'title')
55                     // .append('p')
56                     .text(charDesc);
57                     // console.log('after pt')
58                 d3.select('.product__subtitle')
59                     .text("NO. " + pokecharID);
60
61                 var types = ""
62                 for (let type of data.types) {
63                     // console.log("data", data.types)
64                     let temp = "<li>"
65                     temp += type.type.name
66                     types += temp + "</li>"
67                 }
68                 d3.select('.product__price')
69                     .text("TYPE(s): <ul>" + types+"</ul>")
```



```
/* Page Loader */
.js .loading::before,
.js .loading::after {
    content: '';
    position: fixed;
    z-index: 100000;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: #333;
}

.js .loading::after {
    background: url(..static/pokeball.webp) no-repeat 50% 50%;
    background-size: 5em;
    pointer-events: none;
    opacity: 0.1;
    animation: loaderAnim 1s linear infinite alternate forwards;
}

@keyframes loaderAnim {
    to {
        opacity: 1;
    }
}
```

A photograph of two Poké Balls in a field of tall green grass under a blue sky. One Poké Ball is in the foreground on the left, and another is partially visible behind it on the right.

```
//-----clears the page and resets when the user chooses a different character-----//
function optionChanged(pokecharID) {
  var dropdownMenu = d3.select("#selDataset");
  var pokechar_nameID = dropdownMenu.property("value");
  var values = pokechar_nameID.split("-");
  var pokecharID = values[1];

  //remove previous selection before rebuilding page
  var img = d3.select("#char-image");
  img.remove();

  var p = d3.select("#char-desc");
  p.remove();

  var type = d3.selectAll("#char-type");
  type.remove();

  var move = d3.selectAll("#char-move");
  move.remove();

  //update character image to match new selection
  getCharImage(pokecharID);
  getCharDesc(pokecharID);
  getCharType(pokecharID);
  getCharMoves(pokecharID);
  getBaseStats(pokecharID);
}
```

Choose a Pokémon.



Thank you!