

code_to_pdf.py

```
from reportlab.platypus import SimpleDocTemplate, Paragraph, PageBreak, Preformatted
from reportlab.lib.styles import getSampleStyleSheet, ParagraphStyle
from reportlab.lib.pagesizes import LETTER
from reportlab.lib.units import inch
import os

def is_text_file(filepath):
    try:
        with open(filepath, 'r', encoding='utf-8') as f:
            f.read()
        return True
    except:
        return False

def generate_code_pdf(directory_path, output_path):
    styles = getSampleStyleSheet()
    code_style = ParagraphStyle(
        'Code',
        fontName='Courier',
        fontSize=8,
        leading=10,
    )

    doc = SimpleDocTemplate(
        output_path,
        pagesize=LETTER,
        leftMargin=0.75 * inch,
        rightMargin=0.75 * inch
    )
    elements = []

    exclude_dirs = {'venv', '.git', '__pycache__'}
    exclude_exts = {'.jpg', '.jpeg', '.png', '.pdf', '.pyc', '.DS_Store'}

    for root, dirs, files in os.walk(directory_path):
        dirs[:] = [d for d in dirs if d not in exclude_dirs]
        for file in sorted(files):
            ext = os.path.splitext(file)[1].lower()
            if ext in exclude_exts:
                continue

            file_path = os.path.join(root, file)
            if not is_text_file(file_path):
                print(f"Skipping binary file: {file_path}")
                continue

            rel_path = os.path.relpath(file_path, directory_path)
            try:
                with open(file_path, 'r', encoding='utf-8') as f:
                    content = f.read()
                    elements.append(Paragraph(f"<b>{rel_path}</b>", styles['Heading3']))
                    elements.append(Preformatted(content, code_style))
                    elements.append(PageBreak())
            except Exception as e:
                print(f"Skipping {file_path}: {e}")

    doc.build(elements)
    print(f"■ PDF generated: {output_path}")

# Run this in your script
generate_code_pdf(".", "output.pdf")
```

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Find Shoofly Pie • findshoofly.com</title>
  <!-- Load Rubik font -->
  <link href="https://fonts.googleapis.com/css2?family=Rubik:wght@400;700&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="assets/style.css" />
</head>
<body>
  <main>
    <div class="hero">
      
      <h1>Find Shoofly</h1>
      
    </div>

    <nav class="main-nav">
      <a href="map.html">■ Shoofly Map</a>
      <a href="list.html">■ Shoofly List</a>
      <a href="index.html">■ About Shoofly</a>
    </nav>

    <section class="section">
      <h2>What is Shoofly Pie?</h2>
      
      <p>Shoofly pie is a sweet molasses crumb pie rooted in Pennsylvania Dutch heritage. It comes in wet and dry b<
    </section>

    <section class="section">
      <h2>History of Shoofly Pie</h2>
      
      <p>The dessert dates back to the late 1800s, when it was often served for breakfast with coffee. Its name com<
    </section>

    <section class="section">
      <h2>Wet Bottom vs. Dry Bottom</h2>
      <div class="side-by-side">
        
        
      </div>
      <p>Wet-bottom pies have a gooey molasses layer above the crust. Dry-bottom pies are more crumbly and cake-lik<
    </section>

    <section class="section">
      <h2>About Find Shoofly</h2>
      <p>We collect and share places that serve this classic pie so lovers and the shoofly-curious can find a slice<
    </section>
  </main>

  <footer>
    &copy; 2025 findshoofly.com. All rights reserved.
  </footer>
</body>
</html>
```

list.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Shoofly List</title>
  <link href="https://fonts.googleapis.com/css2?family=Rubik:wght@400;700&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="assets/style.css" />
</head>
<body>
  <main>
    <div class="hero">
      
      <h1>Shoofly List</h1>
      
    </div>

    <nav class="main-nav">
      <a href="map.html">■ Shoofly Map</a>
      <a href="list.html">■ Shoofly List</a>
      <a href="index.html">■ About Shoofly</a>
    </nav>

    <section class="section">
      <table id="shoofly-table">
        <thead>
          <tr>
            <th>Name</th>
            <th>City</th>
            <th>State</th>
            <th>Style</th>
          </tr>
        </thead>
        <tbody></tbody>
      </table>
    </section>

    <section class="section">
      <a href="submit.html" class="button">Submit a Shoofly Spot</a>
    </section>
  </main>

  <footer>
    &copy; 2025 findshoofly.com. All rights reserved.
  </footer>

  <script src="assets/list.js"></script>
</body>
</html>
```

map.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Shoofly Map</title>
  <link href="https://fonts.googleapis.com/css2?family=Rubik:wght@400;700&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="assets/style.css" />
  <!-- Google Maps JS API -->
  <script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwvqpPxd8wHd5VjN7n04cD-IYWUQrH6xo"></script>
</head>
<body>
  <main>
    <div class="hero">
      
      <h1>Shoofly Map</h1>
      
    </div>

    <nav class="main-nav">
      <a href="map.html">■ Shoofly Map</a>
      <a href="list.html">■ Shoofly List</a>
      <a href="index.html">■ About Shoofly</a>
    </nav>

    <section class="section">
      <div id="map" style="height: 80vh; width: 100%;"></div>
    </section>

    <section class="section">
      <a href="submit.html" class="button">Submit a Shoofly Spot</a>
    </section>
  </main>

  <footer>
    &copy; 2025 findshoofly.com. All rights reserved.
  </footer>

  <script src="assets/map.js"></script>
</body>
</html>
```

package.json

```
{
  "name": "find-shoofly",
  "version": "0.1.0",
  "description": "A simple static website that lists places selling shoofly pie and shows them on an interactive map",
  "private": true,
  "scripts": {
    "start": "http-server -p 8080 -c-1 ."
  },
  "dependencies": {},
  "devDependencies": {}
}
```

submit.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Submit a Shoofly Spot</title>
  <link rel="stylesheet" href="assets/style.css" />
</head>
<body>
  <main>
    <div class="hero">
      <h1>
        
        Submit a Shoofly Spot
        
      </h1>
    </div>

    <nav class="main-nav">
      <a href="map.html">Shoofly Map</a>
      <a href="list.html">Shoofly List</a>
      <a href="index.html">About Shoofly</a>
    </nav>

    <section class="section">
      <h2>Submit a Shoofly Spot</h2>
      <p>Know a place that serves great shoofly pie? Let us know!</p>
      <form action="https://formspre.io/f/mwpqwejb" method="POST">
        <label>Name of Place:<br><input type="text" name="place" required /></label><br><br>
        <label>City, State:<br><input type="text" name="location" required /></label><br><br>
        <label>Style:<br>
          <select name="style">
            <option>Wet Bottom</option>
            <option>Dry Bottom</option>
            <option>Both</option>
          </select>
        </label><br><br>
        <label>Notes:<br><textarea name="notes" rows="4"></textarea></label><br><br>
        <button type="submit">Submit Spot</button>
      </form>
    </section>
  </main>

  <footer>
    &copy; 2025 findshoofly.com. All rights reserved.
  </footer>
</body>
</html>
```

assets/list.js

```
let shooflyData = [];

function populateTable(data) {
  const tbody = document.querySelector("#shoofly-table tbody");
  tbody.innerHTML = "";

  data.forEach(item => {
    const row = document.createElement("tr");
    row.innerHTML = `
      <td>${item.name} || "N/A"</td>
      <td>${item.city} || "N/A"</td>
      <td>${item.state} || "N/A"</td>
      <td>${item.type} || "Unknown"</td>
    `;
    tbody.appendChild(row);
  });
}

function populateFilters() {
  const states = [...new Set(shooflyData.map(item => item.state).filter(Boolean))].sort();
  const cities = [...new Set(shooflyData.map(item => item.city).filter(Boolean))].sort();
  const styles = [...new Set(shooflyData.map(item => item.type).filter(Boolean))].sort();

  const stateFilter = document.getElementById("state-filter");
  const cityFilter = document.getElementById("city-filter");
  const styleFilter = document.getElementById("style-filter");

  // Clear existing options except "All"
  [stateFilter, cityFilter, styleFilter].forEach(select => {
    select.innerHTML = `<option>All</option>`;
  });

  states.forEach(state => {
    const option = document.createElement("option");
    option.value = state;
    option.textContent = state;
    stateFilter.appendChild(option);
  });

  cities.forEach(city => {
    const option = document.createElement("option");
    option.value = city;
    option.textContent = city;
    cityFilter.appendChild(option);
  });

  styles.forEach(style => {
    const option = document.createElement("option");
    option.value = style;
    option.textContent = style;
    styleFilter.appendChild(option);
  });
}

function applyFilters() {
  const stateValue = document.getElementById("state-filter").value;
  const cityValue = document.getElementById("city-filter").value;
  const styleValue = document.getElementById("style-filter").value;

  let filteredData = shooflyData;
  if (stateValue !== "All") {
```

```

        filteredData = filteredData.filter(item => item.state === stateValue);
    }
    if (cityValue !== "All") {
        filteredData = filteredData.filter(item => item.city === cityValue);
    }
    if (styleValue !== "All") {
        filteredData = filteredData.filter(item => item.type === styleValue);
    }

    populateTable(filteredData);
}

document.addEventListener("DOMContentLoaded", () => {
    fetch("assets/locations.json")
        .then(response => response.json())
        .then(data => {
            shooflyData = data;
            populateTable(shooflyData);
            populateFilters();

            document.getElementById("state-filter").addEventListener("change", applyFilters);
            document.getElementById("city-filter").addEventListener("change", applyFilters);
            document.getElementById("style-filter").addEventListener("change", applyFilters);
        })
        .catch(error => console.error("Error loading locations.json:", error));
});

```


assets/locations.json

```
[
  {
    "id": 1,
    "name": "Dutch Haven",
    "address": "2857A Lincoln Highway E, Ronks, PA 17572",
    "city": "Ronks",
    "state": "PA",
    "latitude": 40.0158954,
    "longitude": -76.1584928,
    "type": "wet",
    "website": "https://www.dutchhaven.com",
    "description": "Landmark Lancaster County shop in a windmill■topped building, famous for wet bottom shoofly pie",
  },
  {
    "id": 2,
    "name": "Shady Maple Smorgasbord",
    "address": "129 Toddy Dr, East Earl, PA 17519",
    "city": "East Earl",
    "state": "PA",
    "latitude": 40.115952,
    "longitude": -76.032127,
    "type": "wet",
    "website": "https://www.shady-maple.com",
    "description": "Huge buffet restaurant where wet bottom shoofly pie is a permanent resident on the dessert bar.",
  },
  {
    "id": 3,
    "name": "Achenbach's Pastries",
    "address": "375 E Main St, Leola, PA 17540",
    "city": "Leola",
    "state": "PA",
    "latitude": 40.089,
    "longitude": -76.174,
    "type": "wet",
    "website": "https://achenbachs.com",
    "description": "Family bakery known for Long John donuts and other baked goods, including highly rated wet bott",
  },
  {
    "id": 4,
    "name": "Bird in Hand Bake Shop",
    "address": "542 Gibbons Rd, Bird in Hand, PA 17505",
    "city": "Bird in Hand",
    "state": "PA",
    "latitude": 40.042,
    "longitude": -76.182,
    "type": "both",
    "website": "https://bihbakeshop.com",
    "description": "Homemade Amish■style bake shop offering pies, whoopie pies and other treats; claims to make the",
  },
  {
    "id": 5,
    "name": "Granny's Discount Groceries",
    "address": "375 E Main St, Lancaster, PA 17602",
    "city": "Lancaster",
    "state": "PA",
    "latitude": 40.0379,
    "longitude": -76.3055,
    "type": "wet",
    "website": "https://lanclocal.com",
    "description": "Small Amish■owned grocery store with scratch■made baked goods, including wet bottom shoofly pie",
  },
]
```

```

{
  "id": 6,
  "name": "Lancaster Central Market",
  "address": "23 N Market St, Lancaster, PA 17603",
  "city": "Lancaster",
  "state": "PA",
  "latitude": 40.038,
  "longitude": -76.305,
  "type": "wet",
  "website": "https://centralmarketlancaster.com",
  "description": "Historic market where several vendors sell wet bottom shoofly pies, including Marley's Country
},
{
  "id": 7,
  "name": "Miller's Bakery",
  "address": "2811 Lincoln Hwy E, Ronks, PA 17572",
  "city": "Ronks",
  "state": "PA",
  "latitude": 40.017,
  "longitude": -76.175,
  "type": "wet",
  "website": "https://shoppesatmillers.com",
  "description": "Bakery attached to Miller's Smorgasbord serving the classic Pennsylvania Dutch wet bottom shoof
},
{
  "id": 8,
  "name": "Root's Country Market",
  "address": "705 Graystone Rd, Manheim, PA 17545",
  "city": "Manheim",
  "state": "PA",
  "latitude": 40.164,
  "longitude": -76.396,
  "type": "wet",
  "website": "https://rootsmarket.com",
  "description": "Tuesday only farmers market where numerous stands offer shoofly pies along with produce and cra
},
{
  "id": 9,
  "name": "S. Clyde Weaver",
  "address": "5253 Main St, East Petersburg, PA 17520",
  "city": "East Petersburg",
  "state": "PA",
  "latitude": 40.099,
  "longitude": -76.354,
  "type": "both",
  "website": "https://sclydeweaver.com",
  "description": "Specialty meat and cheese shop that also sells wet bottom shoofly pies in store and ships them
}
]

```

assets/map.js

```
function initMap() {
  const map = new google.maps.Map(document.getElementById("map"), {
    zoom: 7,
    center: { lat: 40.2732, lng: -76.8867 } // center on Pennsylvania
  });

  fetch("assets/locations.json")
    .then(response => response.json())
    .then(data => {
      data.forEach(loc => {
        new google.maps.Marker({
          position: { lat: loc.latitude, lng: loc.longitude },
          map,
          title: loc.name
        });
      });
    })
    .catch(err => console.error("Error loading locations:", err));
}
```

assets/style.css

```
/* General layout */
body {
  margin: 0;
  font-family: 'Rubik', sans-serif;
  background-color: #faf4ef;
  color: #3a2c27;
  line-height: 1.6;
}

/* Hero section */
.hero {
  display: flex;
  align-items: center;
  justify-content: center;
  gap: 1rem;
  padding: 1.5rem 1rem;
}

.hero h1 {
  font-size: 2.5rem;
  font-weight: 700;
}

.pie-icon {
  max-width: 60px;
  height: auto;
  border: none;
}

/* Navigation */
.main-nav {
  display: flex;
  justify-content: center;
  background-color: #fff8f1;
  border-bottom: 2px solid #e0c9a6;
  padding: 0.75rem;
  gap: 1.5rem;
}

.main-nav a {
  text-decoration: none;
  color: #5c3b26;
  font-weight: bold;
  font-size: 1.1rem;
}

.main-nav a:hover {
  color: #b84d26;
}

/* Sections */
.section {
  padding: 2rem;
  max-width: 960px;
  margin: 0 auto;
}

.section h2 {
  margin-bottom: 0.5rem;
}

.section p {
```

```

    margin-top: 0;
}

.section-img {
    max-width: 100%;
    height: auto;
    display: block;
    margin: 0.5rem auto 1rem;
}

.side-by-side {
    display: flex;
    gap: 1rem;
    flex-wrap: wrap;
    justify-content: center;
}

/* Table */
table {
    width: 100%;
    border-collapse: collapse;
    background-color: #fffd8;
}

th, td {
    padding: 0.75rem;
    border: 1px solid #d8c2a0;
    text-align: left;
}

th {
    background-color: #f2e6d9;
}

/* Buttons */
.button {
    display: inline-block;
    padding: 0.75rem 1.5rem;
    background-color: #8b5e3c;
    color: #fff;
    text-decoration: none;
    border-radius: 6px;
    font-weight: bold;
}

.button:hover {
    background-color: #b84d26;
}

/* Footer */
footer {
    background-color: #8b5e3c;
    color: #fff;
    text-align: center;
    padding: 1rem;
    margin-top: 2rem;
}

```