

CODING

```
import os

from fpdf import FPDF

# Define supported file extensions
SUPPORTED_EXTENSIONS = ['.py', '.js', '.html', '.css', '.json', '.txt', '.md']

# Create a PDF class
class PDF(FPDF):

    def header(self):

        self.set_font("Arial", "B", 14)

        self.cell(0, 10, "Project Source Code", ln=True, align="C")

        self.ln(5)

    def add_code(self, file_path, code):

        self.set_font("Courier", size=10)

        self.set_text_color(50, 50, 50)

        self.multi_cell(0, 5, f"\n\n--- {file_path} ---\n\n{code}")

    def is_supported(filename):

        return os.path.splitext(filename)[1] in SUPPORTED_EXTENSIONS

    def get_all_files(base_dir):

        for root, _, files in os.walk(base_dir):

            for file in files:

                if is_supported(file):

                    yield os.path.join(root, file)

    def create_pdf_from_project(project_path, output_pdf="project_code.pdf"):

        pdf = PDF()
```

```
pdf.add_page()
```

```
for file_path in get_all_files(project_path):
```

```
    try:
```

```
        with open(file_path, "r", encoding="utf-8") as f:
```

```
            code = f.read()
```

```
            pdf.add_code(file_path, code)
```

```
    except Exception as e:
```

```
        print(f"Skipping {file_path}: {e}")
```

```
pdf.output(output_pdf)
```

```
print(f"✅ PDF saved as: {output_pdf}")
```

```
# Example usage
```

```
if __name__ == "__main__":
```

```
    project_folder = "path/to/your/project" # Replace this
```

```
    create_pdf_from_project(project_folder)
```