1. Python Overview

Python is an interpreted, high-level language known for its simplicity and readability.

- **Syntax**: Python uses indentation instead of braces {}.
- **Variables**: No need to declare types; Python is dynamically typed.

python

```python
x = 5  # Integer
name = "John"  # String
```

- **Functions**:

python

```python
def add(a, b):
    return a + b
```

- **Data Structures**:
    - **Lists**:

python

```python
my_list = [1, 2, 3]
```

    - **Dictionaries**:

python

```python
my_dict = {"key": "value"}
```

- **Installing Packages**: Use pip.

bash

```bash
pip install flask
```

---

2. Flask Overview

Flask is a micro web framework for building web applications in Python.

- **Installation**:

bash

```bash
pip install flask
```

- **Basic Flask App**:

python

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def home():
    return "Hello, World!"

if __name__ == "__main__":
    app.run(debug=True)
```

- **Key Concepts**:
  - **Routes**: Define URL endpoints using the @app.route() decorator.
  - **Templates**: Serve HTML files using Flask's templating system, Jinja2.

python

```python
from flask import render_template

@app.route('/welcome')
def welcome():
    return render_template('welcome.html')
```

- **Running the App**: Run the Python script and Flask will start a web server. Visit http://127.0.0.1:5000/.

---

3. HTML Overview
HTML is used to structure content on the web.

- **Basic Structure**:

html

```html
<!DOCTYPE html>

<html>

<head>
```

```html
    <title>My App</title>
</head>
<body>
    <h1>Hello, World!</h1>
    <p>This is a paragraph.</p>
</body>
</html>
```

- **Common Tags**:
  - <h1> to <h6>: Headings.
  - <p>: Paragraph.
  - <a href="link">: Hyperlink.
  - <img src="image.jpg">: Image.
- **Forms**:

html

```html
<form action="/submit" method="POST">
    <input type="text" name="username">
    <input type="submit" value="Submit">
</form>
```

---

4. CSS Overview
CSS is used to style HTML elements.

- **Basic Syntax**:

css

```css
body {
    background-color: #f0f0f0;
    font-family: Arial, sans-serif;
}


h1 {
```

```
  color: blue;

  text-align: center;

}
```

- **Linking CSS to HTML**:

html

```html
<link rel="stylesheet" type="text/css" href="styles.css">
```

- **Box Model**:

css

```css
div {

  padding: 10px;

  border: 1px solid black;

  margin: 20px;

}
```

---

5. Putting It All Together

- **File Structure**:

arduino

```
my_app/
├── app.py
├── templates/
│   └── index.html
└── static/
    └── styles.css
```

- **Example app.py**:

python

```python
from flask import Flask, render_template


app = Flask(__name__)
```

```python
@app.route('/')
def index():
    return render_template('index.html')


if __name__ == "__main__":
    app.run(debug=True)
```

- **Example templates/index.html**:

html
```html
<!DOCTYPE html>
<html>
<head>
    <title>My Flask App</title>
    <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='styles.css') }}">
</head>
<body>
    <h1>Welcome to My Flask App</h1>
    <p>This is a sample app using Python, Flask, HTML, and CSS.</p>
</body>
</html>
```

- **Example static/styles.css**:

css
```css
body {
    font-family: Arial, sans-serif;
    background-color: #f9f9f9;
}


h1 {
    color: #333;
}
```

6. Flask with Forms and Data Handling

- **HTML Form**:

html

```html
<form action="/submit" method="POST">
    <input type="text" name="username" placeholder="Enter your name">
    <input type="submit" value="Submit">
</form>
```

- **Flask Route to Handle Form**:

python

```python
from flask import request


@app.route('/submit', methods=['POST'])
def submit():
    username = request.form['username']
    return f"Hello, {username}!"
```