Flask Madlibs Solution

Basic Version

basic/app.py

```python
from flask import Flask, render_template, request
from flask_debugtoolbar import DebugToolbarExtension
from stories import story

app = Flask(__name__)
app.config['SECRET_KEY'] = "secret"

debug = DebugToolbarExtension(app)


@app.route("/")
def ask_questions():
    """Generate and show form to ask words."""

    prompts = story.prompts

    return render_template("questions.html", prompts=prompts)


@app.route("/story")
def show_story():
    """Show story result."""

    text = story.generate(request.args)

    return render_template("story.html", text=text)
```

basic/templates/questions.html

```html
<!doctype html>
<html>
<head>
  <title>Madlibs</title>
</head>
<body>
  <h1>Madlibs</h1>
  <form action="/story">
    {% for prompt in prompts %}
      <p>{{ prompt }}:
        <input name="{{ prompt }}">
      </p>
    {% endfor %}
    <button>Submit</button>
  </form>
```

```
  </body>
</html>
```

basic/templates/story.html

```html
<!doctype html>
<html>
<head>
  <title>Madlibs</title>
</head>
<body>
  <h1>Your Story</h1>
  <p>{{ text }}</p>
</body>
</html>
```

Further Study

further-study/stories.py

```python
"""Madlibs Stories."""


class Story:
    """Madlibs story.

    To  make a story, pass a code, a title, a list of prompts,
    and the text of the template.

        >>> s = Story(
        ...      "simple",
        ...      "A Simple Tale",
        ...      ["noun", "verb"],
        ...      "I love to {verb} a good {noun}.")

    To generate text from a story, pass in a dictionary-like thing
    of {prompt: answer, promp:answer):

        >>> ans = {"verb": "eat", "noun": "mango"}
        >>> s.generate(ans)
        'I love to eat a good mango.'
    """

    def __init__(self, code, title, words, text):
        """Create story with words and template text."""

        self.code = code
        self.title = title
        self.prompts = words
        self.template = text

    def generate(self, answers):
```

```
        """Substitute answers into text."""

        text = self.template

        for (key, val) in answers.items():
            text = text.replace("{" + key + "}", val)

        return text


# Here's a story to get you started

story1 = Story(
    "history",
    "A History Tale",
    ["place", "noun", "verb", "adjective", "plural_noun"],
    """Once upon a time in a long-ago {place}, there lived a
        large {adjective} {noun}. It loved to {verb} {plural_noun}."""
)

story2 = Story(
    "omg",
    "An Excited Adventure",
    ["noun", "verb"],
    """OMG!! OMG!! I love to {verb} a {noun}!"""
)

# Make dict of {code:story, code:story, ...}
stories = {s.code: s for s in [story1, story2]}
```

further-study/app.py

```python
from flask import Flask, render_template, request
from flask_debugtoolbar import DebugToolbarExtension
from stories import stories

app = Flask(__name__)
app.config['SECRET_KEY'] = "secret"

debug = DebugToolbarExtension(app)


@app.route("/")
def ask_story():
    """Show list-of-stories form."""

    return render_template("select-story.html",
                           stories=stories.values())


@app.route("/questions")
def ask_questions():
    """Generate and show form to ask words."""

    story_id = request.args["story_id"]
    story = stories[story_id]
```

```python
        prompts = story.prompts

        return render_template("questions.html",
                               story_id=story_id,
                               title=story.title,
                               prompts=prompts)


@app.route("/story")
def show_story():
    """Show story result."""

    story_id = request.args["story_id"]
    story = stories[story_id]

    text = story.generate(request.args)

    return render_template("story.html",
                           title=story.title,
                           text=text)
```

further-study/templates/select-story.html

```html
<!doctype html>
<html>
<head>
  <title>Madlibs</title>
</head>
<body>
  <h1>Madlibs</h1>
  <form action="/questions">
    <select name="story_id">
      {% for story in stories %}
        <option value="{{ story.code }}">
          {{ story.title }}
        </option>
      {% endfor %}
    </select>
    <button>Go</button>
  </form>
</body>
</html>
```

further-study/templates/questions.html

```html
<!doctype html>
<html>
<head>
  <title>Madlibs</title>
</head>
<body>
  <h1>Questions for {{ title }}</h1>
  <form action="/story">
    {% for prompt in prompts %}
      <p>{{ prompt }}:
```

```
        <input name="{{ prompt }}">
      </p>
    {% endfor %}
    <input type="hidden" name="story_id" value="{{ story_id}}">
    <button>Submit</button>
  </form>
</body>
</html>
```

further-study/templates/story.html

```
<!doctype html>
<html>
<head>
  <title>Madlibs</title>
</head>
<body>
  <h1>Your Version of {{ title }}</h1>
  <p>{{ text }}</p>
</body>
</html>
```