

Blogly Solution: Part Two

The solution is provided in three parts. [Download solution.](#)

Part Two

```
part-two/models.py

"""SQLAlchemy models for blogly."""

import datetime
from flask_sqlalchemy import SQLAlchemy

db = SQLAlchemy()

DEFAULT_IMAGE_URL = "https://www.freeiconspng.com/uploads/icon-user-blue-symbol-people-person-generic--public-domain--21.png"

class User(db.Model):
    """Site user."""

    __tablename__ = "users"

    id = db.Column(db.Integer, primary_key=True)
    first_name = db.Column(db.Text, nullable=False)
    last_name = db.Column(db.Text, nullable=False)
    image_url = db.Column(db.Text, nullable=False, default=DEFAULT_IMAGE_URL)

    posts = db.relationship("Post", backref="user", cascade="all, delete-orphan")

    @property
    def full_name(self):
        """Return full name of user."""

        return f"{self.first_name} {self.last_name}"

class Post(db.Model):
    """Blog post."""

    __tablename__ = "posts"

    id = db.Column(db.Integer, primary_key=True)
    title = db.Column(db.Text, nullable=False)
    content = db.Column(db.Text, nullable=False)
    created_at = db.Column(
        db.DateTime,
        nullable=False,
        default=datetime.datetime.now)
    user_id = db.Column(db.Integer, db.ForeignKey('users.id'), nullable=False)

    @property
    def friendly_date(self):
        """Return nicely-formatted date."""

        return self.created_at.strftime("%a %b %-d %Y, %I:%M %p")

def connect_db(app):
    """Connect this database to provided Flask app.

    You should call this in your Flask app.
    """

    db.app = app
    db.init_app(app)
```

```
part-two/app.py

from flask import Flask, request, redirect, render_template, flash
from flask_debugtoolbar import DebugToolbarExtension
from models import db, connect_db, User, Post

app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = "postgresql:///blogly"
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
app.config['SECRET_KEY'] = 'ihaveasecret'

# Having the Debug Toolbar show redirects explicitly is often useful;
# however, if you want to turn it off, you can uncomment this line:
#
# app.config['DEBUG_TB_INTERCEPT_REDIRECTS'] = False

toolbar = DebugToolbarExtension(app)

connect_db(app)
db.create_all()

@app.route('/')
def root():
    """Show recent list of posts, most-recent first."""

    posts = Post.query.order_by(Post.created_at.desc()).limit(5).all()
    return render_template("posts/homepage.html", posts=posts)

@app.errorhandler(404)
def page_not_found(e):
    """Show 404 NOT FOUND page."""

    return render_template('404.html'), 404

#####
# User route

@app.route('/users')
def users_index():
    """Show a page with info on all users"""

    users = User.query.order_by(User.last_name, User.first_name).all()
    return render_template('users/index.html', users=users)

@app.route('/users/new', methods=["GET"])
def users_new_form():
    """Show a form to create a new user"""

    return render_template('users/new.html')

@app.route('/users/new', methods=["POST"])
def users_new():
    """Handle form submission for creating a new user"""

    new_user = User(
        first_name=request.form['first_name'],
        last_name=request.form['last_name'],
        image_url=request.form['image_url'] or None)

    db.session.add(new_user)
    db.session.commit()
    flash(f"User {new_user.full_name} added.")

    return redirect("/users")

@app.route('/users/<int:user_id>')
def users_show(user_id):
    """Show a page with info on a specific user"""

    user = User.query.get_or_404(user_id)
    return render_template('users/show.html', user=user)

@app.route('/users/<int:user_id>/edit')
def users_edit(user_id):
    """Show a form to edit an existing user"""

    user = User.query.get_or_404(user_id)
    return render_template('users/edit.html', user=user)

@app.route('/users/<int:user_id>/edit', methods=["POST"])
def users_update(user_id):
    """Handle form submission for updating an existing user"""

    user = User.query.get_or_404(user_id)
    user.first_name = request.form['first_name']
    user.last_name = request.form['last_name']
    user.image_url = request.form['image_url']

    db.session.add(user)
    db.session.commit()
    flash(f"User {user.full_name} edited.")

    return redirect("/users")

@app.route('/users/<int:user_id>/delete', methods=["POST"])
def users_destroy(user_id):
    """Handle form submission for deleting an existing user"""

    user = User.query.get_or_404(user_id)
    db.session.delete(user)
    db.session.commit()
    flash(f"User {user.full_name} deleted.")

    return redirect("/users")

#####
# Posts route

@app.route('/users/<int:user_id>/posts/new')
def posts_new_form(user_id):
    """Show a form to create a new post for a specific user"""

    user = User.query.get_or_404(user_id)
    return render_template('posts/new.html', user=user)

@app.route('/users/<int:user_id>/posts/new', methods=["POST"])
def posts_new(user_id):
    """Handle form submission for creating a new post for a specific user"""

    user = User.query.get_or_404(user_id)
    new_post = Post(title=request.form['title'],
                    content=request.form['content'],
                    user=user)

    db.session.add(new_post)
    db.session.commit()
    flash(f"Post '{new_post.title}' added.")

    return redirect(f"/users/{user_id}")

@app.route('/posts/<int:post_id>')
def posts_show(post_id):
    """Show a page with info on a specific post"""

    post = Post.query.get_or_404(post_id)
    return render_template('posts/show.html', post=post)

@app.route('/posts/<int:post_id>/edit')
def posts_edit(post_id):
    """Show a form to edit an existing post"""

    post = Post.query.get_or_404(post_id)
    return render_template('posts/edit.html', post=post)

@app.route('/posts/<int:post_id>/edit', methods=["POST"])
def posts_update(post_id):
    """Handle form submission for updating an existing post"""

    post = Post.query.get_or_404(post_id)
    post.title = request.form['title']
    post.content = request.form['content']

    db.session.add(post)
    db.session.commit()
    flash(f"Post '{post.title}' edited.")

    return redirect(f"/users/{post.user_id}")

@app.route('/posts/<int:post_id>/delete', methods=["POST"])
def posts_destroy(post_id):
    """Handle form submission for deleting an existing post"""

    post = Post.query.get_or_404(post_id)
    db.session.delete(post)
    db.session.commit()
    flash(f"Post '{post.title}' deleted.")

    return redirect(f"/users/{post.user_id}")
```

Templates: Users

```
part-two/templates/users/show.html

{% extends 'base.html' %}

{% block title %}{% user.full_name %}{% endblock %}

{% block content %}

<div class="row">

    {% if user.image_url %}
    <div class="col-sm-2 col-6">
        
    </div>
    {% endif %}

    <div class="col-sm-10 col-12">
        <h1>{{ user.full_name }}</h1>

        <form>
            <button class="btn btn-primary btn-sm"
                formaction="/users/{{ user.id }}/edit"
                formmethod="GET">Edit
            </button>
            <button class="btn btn-danger btn-sm"
                formaction="/users/{{ user.id }}/delete"
                formmethod="POST">Delete
            </button>
        </form>

        <h2 class="mt-4">Posts</h2>

        <ul>
            {% for post in user.posts %}
            <li>
                <a href="/posts/{{ post.id }}">{{ post.title }}</a>
                <small>{{ post.friendly_date }}</small>
            </li>
            {% endfor %}
        </ul>

        <p><a href="/users/{{ user.id }}/posts/new" class="btn btn-primary">Add Post</a></p>
    </div>
</div>

{% endblock %}
```

Templates: Posts

```
part-two/templates/posts/show.html

{% extends 'base.html' %}

{% block title %}{% post.title %}{% endblock %}

{% block content %}

<h1>{{ post.title }}</h1>

<p>{{ post.content }}</p>
<p><i>By {{ post.user.full_name }}
    on {{ post.friendly_date }}</i></p>

<form>
    <button class="btn btn-outline-primary"
        formmethod="GET"
        formaction="/users/{{ post.user_id }}">Cancel</button>
    <button class="btn btn-primary"
        formmethod="GET"
        formaction="/posts/{{ post.id }}/edit">Edit</button>
    <button class="btn btn-danger"
        formmethod="POST"
        formaction="/posts/{{ post.id }}/delete">Delete</button>
</form>

{% endblock %}
```

```
part-two/templates/posts/new.html

{% extends 'base.html' %}

{% block title %}Add Post{% endblock %}

{% block content %}

<h1>Add Post for {{ user.full_name }}</h1>

<form method="POST">

    <div class="form-group row">
        <label for="title" class="col-sm-2 col-form-label">Title</label>
        <div class="col-sm-10">
            <input class="form-control" id="title" name="title">
        </div>
    </div>

    <div class="form-group row">
        <label for="content" class="col-sm-2 col-form-label">Content</label>
        <div class="col-sm-10">
            <textarea class="form-control" id="content" name="content"></textarea>
        </div>
    </div>

    <div class="form-group row">
        <div class="ml-auto mr-3">
            <a href="/users/{{ user.id }}" class="btn btn-info">Cancel</a>
        </div>
        <button type="submit" class="btn btn-success">Add</button>
    </div>
</div>

{% endblock %}
```

```
part-two/templates/posts/edit.html

{% extends 'base.html' %}

{% block title %}Edit Post{% endblock %}

{% block content %}

<h1>Edit Post</h1>

<form method="POST" action="/posts/{{ post.id }}/edit">

    <div class="form-group row">
        <label for="title" class="col-sm-2 col-form-label">Title</label>
        <div class="col-sm-10">
            <input class="form-control" id="title" name="title" value="{{ post.title }}">
        </div>
    </div>

    <div class="form-group row">
        <label for="content" class="col-sm-2 col-form-label">Post Content</label>
        <div class="col-sm-10">
            <textarea class="form-control"
                id="content"
                name="content">{{ post.content }}</textarea>
        </div>
    </div>

    <div class="form-group row">
        <div class="ml-auto mr-3">
            <a href="/users/{{ post.user_id }}" class="btn btn-outline-info">Cancel
            </a>
            <button type="submit" class="btn btn-success">
                Edit
            </button>
        </div>
    </div>
</form>

{% endblock %}
```

```
part-two/templates/posts/homepage.html

{% extends 'base.html' %}

{% block title %}Blogly{% endblock %}

{% block content %}

<h1>Blogly Recent Posts</h1>

{% for post in posts %}
<h2 class="mt-4">
    <a href="/posts/{{ post.id }}">{{ post.title }}</a>
</h2>
<p>{{ post.content }}</p>
<p>
    <small>By {{ post.user.full_name }} on {{ post.friendly_date }}</small>
</p>
{% endfor %}

{% endblock %}
```

Part Three

[See solution for Part Three](#)