

# Overview



**Add and remove cards**

**Template: HTML forms**

**View**

- Accept POST requests
- Process user input
- Redirect after form submit

**Forms and Security**



# Demo



Adding cards

Jinja form template

View for processing form data



# Demo



Deleting cards

Jinja form template

View for processing form data



# Demo



**A word about forms and security**



# HTML Forms

```
<form method="post">
```

Question: `<input type="text" name="question">`

Answer: `<input type="text" name="answer">`

`<button type="submit">Create</button>`

```
</form>
```



# Handling POST Requests

```
@app.route("/add_card", methods=["GET", "POST"])
def add_card():
    if request.method == "POST":
        # Handle user input
    else:
        return render_template("add_card.html")
```



```
card = { "question": request.form["question"],  
         "answer": request.form["answer"] }  
  
... # Save the data to file  
  
return redirect(url_for(...))
```

---

## Processing Form Input Data

Don't forget to import `request`

`request` contains request info: headers, cookies, form data, etc.

`request` is a global object, but bound to the *current* request



# Summary



**Add and remove cards**

**Template: HTML forms**

**View**

- Accept POST requests
- Process user input
- Redirect after form submit

**Forms and Security**

