

# CS2021-Week 9 - Forms

Python Web Development

## HTML Forms

[https://www.udacity.com/wiki/  
cs253/unit-2HTML](https://www.udacity.com/wiki/cs253/unit-2HTML)

### Form for Submitting input:

```
<form> <input name="q"> <input type="submit"> </  
form>
```

### Web Application:

```
import webapp2  
form = """ <form action="http://www.google.com/  
search"> <input name="q"> <input type="submit">  
</form> """
```

```
class MainPage(webapp2.RequestHandler):  
    def get(self):  
        #self.response.headers['Content-Type'] = 'text/  
plain'    self.response.out.write(form)
```

```
app = webapp2.WSGIApplication([('/', MainPage)],
debug=True)
```

## More Handlers

```
import webapp2
```

```
form = """
```

```
<form action="/testform">
```

```
    <input name="q"> <input  
type="submit">
```

```
</form>
```

```
"""
```

```
class
```

```
MainPage(webapp2.RequestHan  
dler):
```

```
    def get(self):
```

```
#self.response.headers['Content-  
Type'] = 'text/plain'
```

```
self.response.out.write(form)
```

```
class
```

```
TestHandler(webapp2.RequestH  
andler):
```

```
    def get(self):
```

```
        q=self.request.get("q")
```

```
        self.response.out.write(q)
```

```
app =
```

```
webapp2.WSGIApplication([('/',  
MainPage),
```

```
                        ('/testform',
```

```
TestHandler)],
```

```
        debug=True)
```

## Validation of Input

```
import webapp2
```

```
form = """
```

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

```
<input name="q"><input
```

```
type="submit">
```

```
</form>
```

```
"""
```

```
class
```

```
MainPage(webapp2.Request
```

```
Handler):
```

```
    def get(self):
```

```
self.response.out.write(
form)
```

```
def post(self):
```

```
self.response.out.write(
"Thanks! Valid day!")
```

```
app =
webapp2.WSGIApplication(
[('/', MainPage)],
debug=True)
```

## Responding Based on Validation

```
def post(self):
    user_month =
valid_month(self.request.get('month'))
```

```
        user_day =
valid_day(self.request.get('day'))
        user_year =
valid_year(self.request.get('year'))

        if not(user_month and user_day and
user_year):
            self.response.out.write(form)
        else:

self.response.out.write("Thanks! That's a
totally valid day!")
```

**Exercise:** Write functions `valid_month()`, `valid_day()`, `valid_year()`. And then run your code to test your solutions.

```
# -----
# User Instructions
#
# Modify the valid_month() function to verify
# whether the data a user enters is a valid
# month. If the passed in parameter 'month'
# is not a valid month, return None.
# If 'month' is a valid month, then return
# the name of the month with the first letter
# capitalized.
#
months = ['January',
          'February',
          'March',
          'April',
          'May',
          'June',
```

```
'July',  
'August',  
'September',  
'October',  
'November',  
'December']
```

```
def valid_month(month):
```

# Redirection

Instead of rendering the result in a post, we send them to another page that says "Thanks!". If the user's post is successful, the server sends a redirect message that causes the browser to get the "success" page.

Why is it nice to redirect after a form submission?

So that reloading the page  
doesn't resubmit the form.  
So we can have distinct pages for  
forms and success pages.

## Implementing Redirection

Let's make the change to our  
application. We need to:

make a "thanks" handler.

add the /thanks URL.

redirect to the /thanks URL.

**handler for redirects**

```
class
```



```
ThanksHandler(webapp2.Re  
questHandler):
```

```
    def get(self)
```

```
        self.response.out.write(  
            "Thanks! That's a  
            totally valid day!")
```

```
app =  
webapp2.WSGIApplication(  
    [('/', MainPage),
```

```
    ('/thanks',  
     ThanksHandler)],
```

```
    debug=True)
```

```
class
MainPage(webapp2.Request
Handler):
    def post(self):
        user_month =
self.request.get('month'
)
        user_day =
self.request.get('day')
        user_year =
self.request.get('year')

        month =
valid_month(user_month)
        day =
```

```
valid_day(user_day)
        year =
valid_year(user_year)

        if not(month and
day and year):

self.write_form("That
doesn't look valid to
me, friend.",
user_month, user_day,
user_year)
        else:

self.redirect("/thanks")
```

Final Version of our  
Application using Validation

## and Redirect

```
import webapp2
```

```
import cgi
```

```
def escape_html(s):
```

```
    return cgi.escape(s, quote =  
True)
```

```
form = """
```

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

```
    What is your birthday?
```

```
    <br>
```

```
    <label>
```

```
        Month
```

```
        <input type="text"
```

```
name="month" value="%
```

```
(month)s">
  </label>
  <label>
    Day
    <input type="text"
name="day" value="% (day)s">
  </label>
  <label>
    Year
    <input type="text"
name="year" value="% (year)s">
  </label>
  <div style="color: red">%
(error)s</div>
  <br><br>
  <input type="submit">
</form>
```

|||||

```
months = ['January', 'February',  
'March', 'April', 'May', 'June',  
'July', 'August',  
          'September',  
'October', 'November',  
'December']
```

```
def valid_day(day):  
    if(day and day.isdigit()):  
        day = int(day)  
    if(day < 32 and day > 0):  
        return day
```

```
def valid_month(month):  
    if(month):
```

```
        month =  
month.capitalize()  
        if(month in months):  
            return month
```

```
def valid_year(year):  
    if(year and year.isdigit()):  
        year = int(year)  
        if(year < 2020 and year >  
1880):  
            return year
```

```
class  
MainPage(webapp2.RequestHan  
dler):  
    def write_form(self, error="",  
month="", day="", year=""):
```

```
        self.response.out.write(form
%{"error": error,
                                "month":
escape_html(month),
                                "day":
escape_html(day),
                                "year":
escape_html(year)})
```

```
def get(self):
    self.write_form()
```

```
def post(self):
    user_month =
self.request.get('month')
    user_day =
self.request.get('day')
```



```
    user_year =  
self.request.get('year')
```

```
    month =  
valid_month(user_month)  
    day = valid_day(user_day)  
    year = valid_year(user_year)
```

```
    if not(month and day and  
year):
```

```
        self.write_form("That  
doesn't look valid to me,  
friend.", user_month, user_day,  
user_year)
```

```
    else:
```

```
        self.redirect("/thanks")
```

```
class
ThanksHandler(webapp2.Reques
tHandler):
    def get(self):

self.response.out.write("Thanks!
That's a totally valid day!")

app =
webapp2.WSGIApplication([('/',
MainPage),
                        ('/thanks',
ThanksHandler)],
                        debug=True)
```

## Homework for Problem Set 2

[https://www.udacity.com/  
course/viewer#!/c-cs253/  
l-48756009/e-48538417/  
m-48696518](https://www.udacity.com/course/viewer#!/c-cs253/l-48756009/e-48538417/m-48696518)