

Python

- [pywebio](#)
- [Mapping](#)

pywebio

```
from pywebio.platform.flask import webio_view, FlaskHttpContext
from flask import Flask, request, send_from_directory, current_app, url_for
from pywebio.input import *
from pywebio.output import *
from pywebio.session import *

app = Flask(__name__)

def bmi():
    current_url = eval_js("window.location.href")
    from urllib.parse import urlsplit, parse_qs

    query = urlsplit(current_url).query
    params = parse_qs(query)

    # Convert the dictionary values from lists to single values
    flat_params = {k: v[0] for k, v in params.items()}

    put_text(flat_params)
    #put_text(current_url)
    height = input("Input your height(cm)", type=FLOAT)
    weight = input("Input your weight(kg)", type=FLOAT)

    BMI = weight / (height / 100) ** 2

    top_status = [(16, 'Severely underweight'), (18.5, 'Underweight'),
                  (25, 'Normal'), (30, 'Overweight'),
                  (35, 'Moderately obese'), (float('inf'), 'Severely obese')]

    for top, status in top_status:
        if BMI <= top:
            put_text('Your BMI: %.1f. Category: %s' % (BMI, status))
            break

app.add_url_rule('/tool', 'test', webio_view(bmi), methods=['GET', 'POST', 'OPTIONS'])
#, provide_automatic_options= True)
app.run(host='0.0.0.0', port=8888)
```

Mapping

```
import pywebio
from pywebio.platform.flask import webio_view
from flask import Flask, request, send_from_directory, current_app, url_for
from pywebio import STATIC_PATH
from pywebio.input import *
from pywebio.output import *
from pywebio.session import *
import logging
from datetime import date, timedelta
from functools import partial
import time
from pywebio.session import info as session_info
from pywebio import config
import threading

app = Flask(__name__)

@app.route('/')
def hello():
    msg = f'<br/><a href="/tool">Start a new session</a>'
    return msg

@config(theme='sketchy',css_style="""header,

    footer {

        background-color: #009932;

        color: #fff;

        padding: 10px 0;

        text-align: center;

    }

    nav ul {

        list-style: none;

        padding: 0;

    }

    nav li {

        display: inline-block;
```

```
    margin-right: 20px;

}

nav a {

    text-decoration: none;

    color: #fff;

}

nav a:hover {

    color: red;

}

main {

    padding: 20px;

}

section {

    background-color: #e6e6e6;

    padding: 20px;

    border-radius: 5px;

    box-shadow: 0px 0px 10px 0px rgba(0, 0, 0, 0.1);

}

footer {

    position: fixed;

    bottom: 0;

    width: 100%;

}

.sticky {

    position: sticky;

    top: 0;

    z-index: 100;
```

```

}

p {

    text-align: center;

}

.tatata {

    position: fixed;

    bottom: 0;
    left:0;
    width:100%;

    background-color: lightgray;

    text-align: center;
    box-sizing: border-box;

}

.footer {

    visibility: hidden;
}

#chuv_logo {
    background-color:transparent;
    float: left;
    height: 40px;
    margin-left: 25px;

}

""
def demo(test=""):
    put_html("""<header class="sticky">



<ul>

<li><a href="#html" style="color:white;">Mapping</a></li>
<li><a href="#css" style="color:white;">Validation</a></li>
<li><a href="#js" style="color:white;">Enrichissement</a></li>
<li><a href="#js" style="color:white;">Loinc</a></li>
</ul>

</nav>

```

</header>

<footer class="tatata">

<p>© 2024 Datascience - DSI All rights reserved.</p>

</footer>'')

```
age = input("What is your name? ")
put_text(f"Hello {age}")
#put_text("You are old")
while age != 2:
    age = input("Enter your age: ", type=NUMBER)
    with use_scope('scope2', clear=True): # enter the existing scope and clear the previous content
        put_text(f"you have selected {age}")
put_text("you win")
```

```
def bmi():
    height = input("Input your height(cm)", type=FLOAT)
    weight = input("Input your weight(kg)", type=FLOAT)

    BMI = weight / (height / 100) ** 2

    top_status = [(16, 'Severely underweight'), (18.5, 'Underweight'),
                  (25, 'Normal'), (30, 'Overweight'),
                  (35, 'Moderately obese'), (float('inf'), 'Severely obese')]
```

```
for top, status in top_status:
    if BMI <= top:
        put_text('Your BMI: %.1f. Category: %s' % (BMI, status))
        break
```

```
def start_flask_server():
    app.add_url_rule('/tool', 'webio_view', webio_view(demo), methods=['GET', 'POST', 'OPTIONS'])
    app.run(host='0.0.0.0', port=8888)
```

```
# if __name__ == '__main__':
#     t = threading.Thread(target=start_flask_server)
#     t.daemon = True
#     t.start()
#     webview.create_window('CEF Example', 'http://localhost:8888')
#     webview.start(gui='cef')
```

```
if __name__ == '__main__':
    start_flask_server()
#     app.add_url_rule('/tool', 'webio_view', webio_view(bmi), methods=['GET', 'POST', 'OPTIONS'])
#     webview.create_window('Flask example', app)
#     webview.start(gui='cef')
```