

Zaawansowane techniki bazodanowe

Zadanie 8

Tomasz Jakubowski

nr albumu 7043

```

@app.route('/wykaz_projektow')
def wykaz_projektow():
    columns = [
        (Projekt, 'id_projekt'),
        (Projekt, 'nr_projekt'),
        (Projekt, 'temat_projekt'),
        (Projekt, 'data_rozpoczecia'),
        (Projekt, 'data_zakonczenia'),
        (Projekt, 'kwota'),
        (Projekt, 'uwagi'),
        (Rodzaj, 'nazwa_rodzaj'),
        (Status, 'nazwa_status'),
    ]

    rows = db.session.query(Projekt,Rodzaj,Status)\
        .join(Rodzaj)\
        .join(Status)\
        .all()

    return render_template('wykaz_projektow.html', columns=columns, rows=rows)

```

Figure 1: Poprawka do wykazu projektów (Python)

```

1 {% extends "bazowa_templatka.html" %}
2
3 {% block title %}
4 Wykaz projektów
5 {% endblock %}
6
7 {% block content %}
8 <h1>Wykaz projektów</h1>
9
10 <table class="table table-scroll bg-dark">
11   <thead>
12     <tr>
13       {% for table_colname in columns %}
14         <th>{{ table_colname[1] }}</th>
15       {% endfor %}
16     </tr>
17   </thead>
18   <tbody>
19     {% for row in rows %}
20       <tr>
21         {% for table_colname in columns %}
22           <td>
23             <div>{{ row[table_colname[0]].__dict__[table_colname[1]] }}</div>
24           </td>
25         {% endfor %}
26       </tr>
27     {% endfor %}
28   </tbody>
29 </table>
30 {% endblock %}

```

Figure 2: Uporządkowany kod HTML wykazu projektów

Wykaz projektów								
id_projekt	nr_projekt	temat_projekt	data_rozpoczecia	data_zakonczenia	kwota	uwagi	nazwa_rodzaj	nazwa_status
6	1/2020	dom	2020-01-23	2022-01-05	500000.0000000000	brak cegieł	budowlany	nowy
7	02/2022	zakup rezonansu mag	2022-01-19	2022-04-21	900000.0000000000	brak	medyczny	nowy
8	ddxdd	temataaaaaaa	2022-02-24	2022-02-09	6669.0000000000	sdfsdf	budowlany	nowy
9	eeeeeeee	123213	2022-02-10	2022-03-01	3333.0000000000	3123123	budowlany	nowy

Figure 3: Naprawiona tabelka w wykazie projektów

```

@app.route('/projekty_ug_rodzaj', methods=['GET', 'POST'])
def projekty_ug_rodzaj():
    columns = [
        (Projekt, 'id_projekt'),
        (Projekt, 'nr_projekt'),
        (Projekt, 'temat_projekt'),
        (Projekt, 'data_rozpoczecia'),
        (Projekt, 'data_zakonczenia'),
        (Projekt, 'kwota'),
        (Projekt, 'uwagi'),
        (Rodzaj, 'nazwa_rodzaj'),
        (Status, 'nazwa_status'),
    ]

    if request.method == 'POST':
        rows = db.session.query(Projekt, Rodzaj, Status) \
            .join(Rodzaj) \
            .join(Status) \
            .filter(Rodzaj.nazwa_rodzaj == request.form['rodzaj']) \
            .all()
    else:
        rows = []

    return render_template('projekty_ug_rodzaj.html', db=db, Rodzaj=Rodzaj, columns=columns, rows=rows)

```

Figure 4: Projekty według rodzaju, kod Python

```

1 {% extends "bazowa_templatka.html" %}
2
3 {% block title %}
4 Projekty według rodzaju
5 {% endblock %}
6
7 {% block content %}
8 <h1>Projekty według rodzaju</h1>
9 <form action="/projekty_wg_rodzaj" method="POST">
10   <div class="form-group">
11     <label class="form-label" for="rodzaj-input">Rodzaj projektu</label>
12     <select id="rodzaj-input" type="text" class="form-select text-dark" name="rodzaj">
13       {% for row in db.session.query(Rodzaj).all() %}
14         <option>{{ row.__dict__['nazwa_rodzaj'] }}</option>
15       {% endfor %}
16     </select>
17   </div>
18   <br/>
19   <button type="submit" class="btn btn-primary">Filtruj</button>
20 </div>
21 </form>
22
23 <table class="table table-scroll bg-dark">
24   <thead>
25     <tr>
26       {% for table_colname in columns %}
27         <th>{{ table_colname[1] }}</th>
28       {% endfor %}
29     </tr>
30   </thead>
31   <tbody>
32     {% for row in rows %}
33       <tr>
34         {% for table_colname in columns %}
35           <td>
36             <div>{{ row[table_colname[0]].__dict__[table_colname[1]] }}</div>
37           </td>
38         {% endfor %}
39       </tr>
40     {% endfor %}
41   </tbody>
42 </table>
43 {% endblock %}

```

Figure 5: Projekty według rodzaju, kod HTML

Projekty według rodzaju

Rodzaj projektu

budowlany

Filtruj

id_projekt	nr_projekt	temat_projekt	data_roz poczenia	data_zakonczenia	kwota	uwagi
6	1/2020	dom	2020-01-23	2022-01-05	500000.0000000000	bra
8	ddxdd	temataaaaaaaa	2022-02-24	2022-02-09	6669.0000000000	sdf
9	eeeeeeee	123213	2022-02-10	2022-03-01	3333.0000000000	312

Figure 6: Projekty według rodzaju, widok strony po załadowaniu

Projekty według rodzaju

Rodzaj projektu

budowlany

Filtruj

id_projekt	nr_projekt	temat_projekt	data_rozpoczecia	data_zakończ
7	02/2022	zakup rezonansu mag	2022-01-19	2022-01-19

Figure 7: Projekty według rodzaju po przefiltrowaniu

```
{% if not rows %}
<h2>Brak projektów do wyświetlenia</h2>
{% else %}
<table class="table table-scroll bg-dark">
  <thead>
    <tr>
      {% for table_colname in columns %}
        <th>{{ table_colname[1] }}</th>
      {% endfor %}
    </tr>
  </thead>
  <tbody>
    {% for row in rows %}
      <tr>
        {% for table_colname in columns %}
          <td>
            <div>{{ row[table_colname[0]].__dict__[table_colname[1]] }}</div>
          </td>
        {% endfor %}
      </tr>
    {% endfor %}
  </tbody>
</table>
{% endif %}
```

Figure 8: Modyfikacja w celu wyświetlenia komunikatu gdy nie wybrano filtru

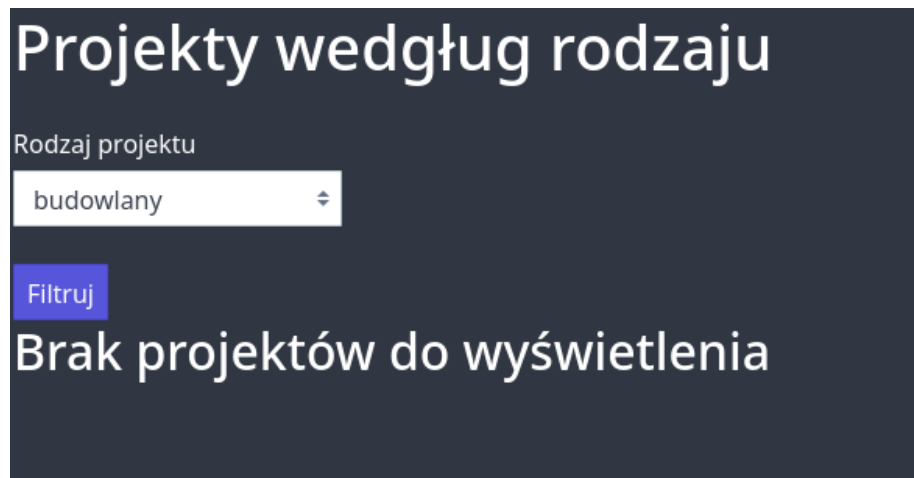


Figure 9: Komunikat wyświetlony na stronie zamiast tabelki

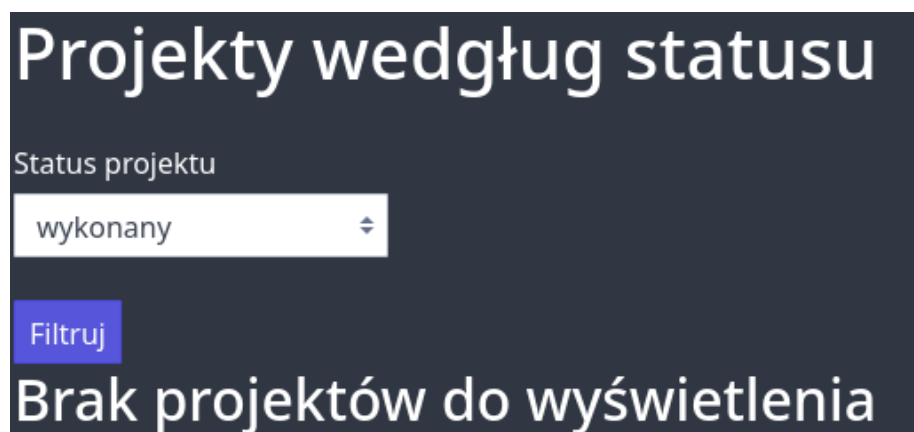


Figure 10: Wybrany filtr dla typu projektu, który nie pasuje do żadnego w bazie

Projekty według statusu

Status projektu

nowy

Filtruj

id_projekt	nr_projekt	temat_projekt	data_roz poczenia	data_zakonczenia	kwo
6	1/2020	dom	2020-01-23	2022-01-05	5000
7	02/2022	zakup rezonansu mag	2022-01-19	2022-04-21	9000
8	ddxdd	temataaaaaaaa	2022-02-24	2022-02-09	6669
9	eeeeeeee	123213	2022-02-10	2022-03-01	3333

Figure 11: Dla istniejącego typu tabelka wciąż wyświetla się prawidłowo

```

1 {% extends "bazowa_templatka.html" %}
2
3 {% block title %}
4 Projekty według statusu
5 {% endblock %}
6
7 {% block content %}
8 <h1>Projekty według statusu</h1>
9 <form action="/projekty_wg_status" method="POST">
10   <div class="form-group">
11     <div class="col-3 col-sm-12">
12       <label class="form-label" for="status-input">Status projektu</label>
13       <select id="status-input" type="text" class="form-select text-dark" name="status">
14         {% for row in db.session.query(Status).all() %}
15           <option>{{ row.__dict__['nazwa_status'] }}</option>
16         {% endfor %}
17       </select>
18     </div>
19   </div>
20   <button type="submit" class="btn btn-primary">Filtruj</button>
21 </form>
22
23 {% if not rows %}
24 <h2>Brak projektów do wyświetlenia</h2>
25 {% else %}
26 <table class="table table-scroll bg-dark">
27   <thead>
28     <tr>
29       {% for table_colname in columns %}
30         <th>{{ table_colname[1] }}</th>
31       {% endfor %}
32     </tr>
33   </thead>
34   <tbody>
35     {% for row in rows %}
36       <tr>
37         {% for table_colname in columns %}
38           <td>
39             <div>{{ row[table_colname[0]].__dict__[table_colname[1]] }}</div>
40           </td>
41         {% endfor %}
42       </tr>
43     {% endfor %}
44   </tbody>
45 </table>
46 {% endif %}
47 {% endblock %}

```

Figure 12: Projekty według statusu, kod HTML

```

@app.route('/projekty_wg_status', methods=['GET', 'POST'])
def projekty_wg_status():
    columns = [
        (Projekt, 'id_projekt'),
        (Projekt, 'nr_projekt'),
        (Projekt, 'temat_projekt'),
        (Projekt, 'data_roz poczenia'),
        (Projekt, 'data_zakonczenia'),
        (Projekt, 'kwota'),
        (Projekt, 'uwagi'),
        (Rodzaj, 'nazwa_rodzaj'),
        (Status, 'nazwa_status'),
    ]

    if request.method == 'POST':
        rows = db.session.query(Projekt, Rodzaj, Status) \
            .join(Rodzaj) \
            .join(Status) \
            .filter(Status.nazwa_status == request.form['status']) \
            .all()
    else:
        rows = []

    return render_template('projekty_wg_status.html', db=db, Status=Status, columns=columns, rows=rows)

```

Figure 13: Projekty według statusu, kod Python

Edytowanie projektu									
	id_projekt	nr_projekt	temat_projekt	data_rozpoczecia	data_zakonczenia	kwota	uwagi	nazwa_rodzaj	nazwa_status
Edytuj	7	02/2022	zakup rezonansu mag	2022-01-19	2022-04-21	900000.0000000000	brak	medyczny	nowy
Edytuj	14	333333	33333333	2022-02-10	2022-02-09	33333333.0000000000	333333333333	budowlany	nowy
Edytuj	15	3333331	11111111	2022-02-09	2022-02-10	11.0000000000	11	medyczny	nowy
Edytuj	16	dssdfsdf	sdfsdfsdf	2022-02-09	2022-02-13	3123123.0000000000	3333asdasd	budowlany	nowy

Figure 14: Tabelka również wyświetla prawidłowe dane

Edytowanie projektu									
	id_projekt	nr_projekt	temat_projekt	data_rozpoczecia	data_zakonczenia	kwota	uwagi	nazwa_rodzaj	nazwa_status
Edytuj	7	02/2022	zakup rezonansu mag	2022-01-19	2022-04-21	900000.0000000000	brak	medyczny	nowy
Anuluj Wyjść Usuń	14	333333	33333333	02/10/2022	02/09/2022	33333333.0000000000	333333333333	budowlany	nowy
Edytuj	15	3333331	11111111	2022-02-09	2022-02-10	11.0000000000	11	medyczny	nowy
Edytuj	16	dssdfsdf	sdfsdfsdf	2022-02-09	2022-02-13	3123123.0000000000	3333asdasd	budowlany	nowy

Figure 15: Tryb edycji, usuwanie rekordu

Edytowanie projektu									
	id_projekt	nr_projekt	temat_projekt	data_rozpoczecia	data_zakonczenia	kwota	uwagi	nazwa_rodzaj	nazwa_status
Edytuj	7	02/2022	zakup rezonansu mag	2022-01-19	2022-04-21	900000.0000000000	brak	medyczny	nowy
Edytuj	15	3333331	11111111	2022-02-09	2022-02-10	11.0000000000	11	medyczny	nowy
Edytuj	16	dssdfsdf	sdfsdfsdf	2022-02-09	2022-02-13	3123123.0000000000	3333asdasd	budowlany	nowy

Figure 16: Udało się usunąć rekord

uwagi	nazwa_rodzaj	nazwa_status
<div> <div></div> <div>zaktualizowana uwaga</div> </div>	<div> <div>medyczny</div> <div></div> </div>	<div> <div>zarchiwizowar</div> <div></div> </div>

Figure 21: Zmienione dane

Edytowanie projektu								
	id_projekt	nr_projekt	temat_projekt	data_rozpoczecia	data_zakonczenia	kwota	uwagi	nazwa_rodzaj nazwa_status
Edytuj	7	02/2022	zakup rezonansu mag123123213	2022-01-19	2022-04-21	1.0000000000	zaktualizowana uwaga	medyczny zarchiwizowany

Figure 22: Dane zaktualizowane z powodzeniem