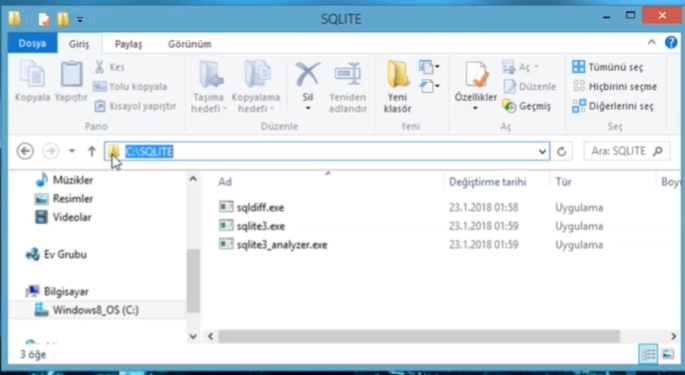
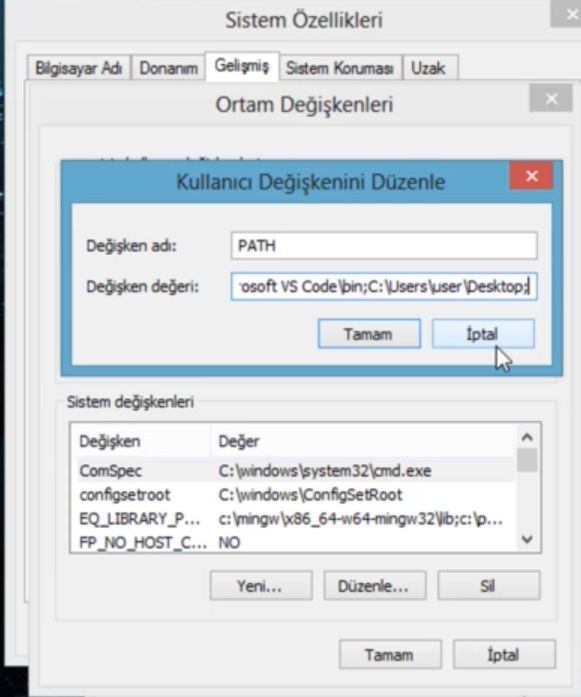
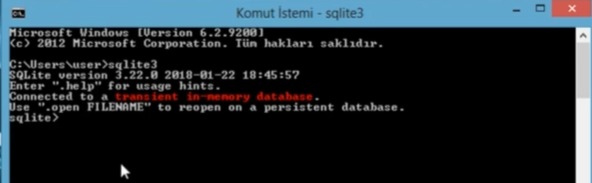
# Bölüm 20 Flask\_ORM\_ve\_SqlAlchemy\_ile\_TodoApp

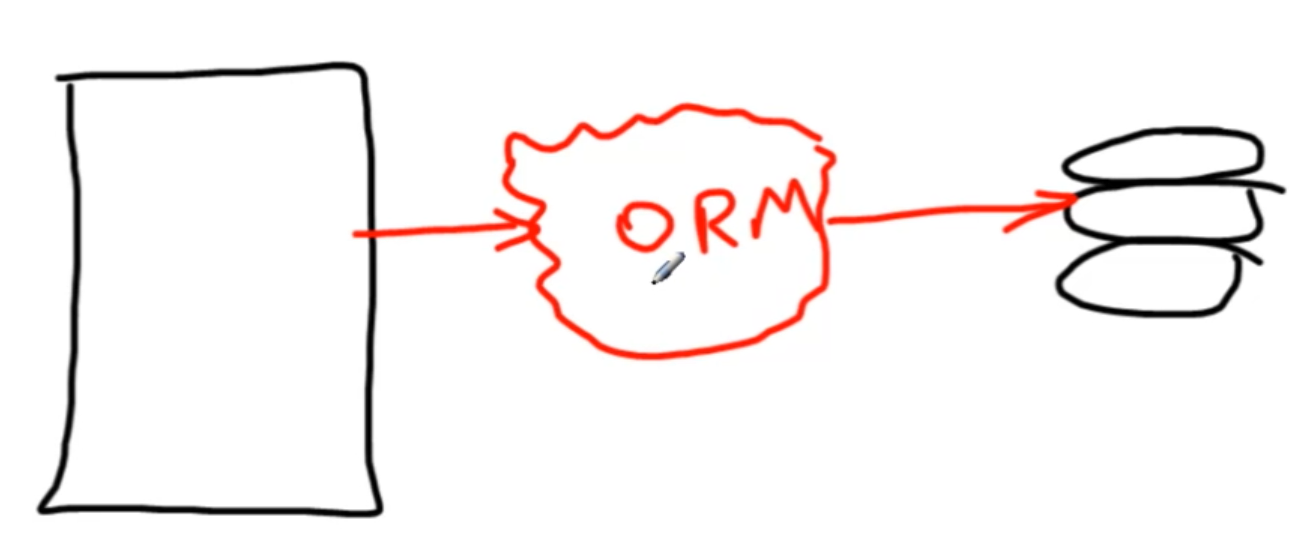
[SQLite Download Page](https://www.sqlite.org/download.html)





Noktalı virgülden sonra dosya konumunu kopyala

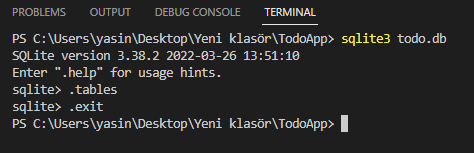




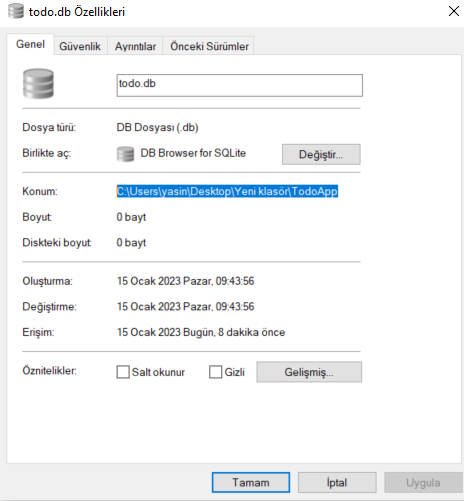
ORM: Object Relational Mapping anlamına gelmektedir. O/R Mapping olarakta yazılabilir. ORM, İlişkisel veritabanı ile nesneye yönelik programların arasında bir tür köprü özelliği gören ve ilişkisel veritabanındaki bilgilerimizi yönetmek için, nesne modellerimizi kullandığımız bir tekniktir.

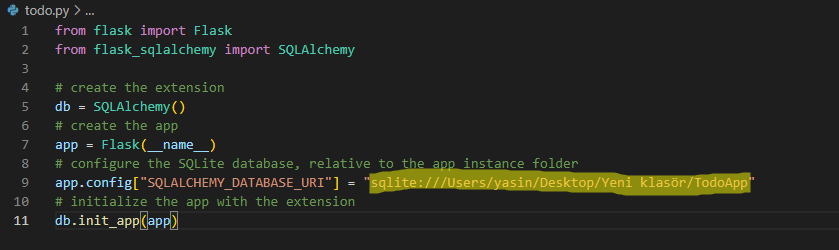
[Quick Start — Flask-SQLAlchemy Documentation (3.0.x) (palletsprojects.com)](https://flask-sqlalchemy.palletsprojects.com/en/3.0.x/quickstart/)

**SQLİTE, FLASK VE SQLALCHEMY İLİŞKİSİ**



Tables yazarak db oluşturduk. Tabloları gösterir ama tablo olmadığından db oluşturuyor.





from flask import Flask

from flask\_sqlalchemy import SQLAlchemy

# create the extension

db = SQLAlchemy()

# create the app

app = Flask(\_\_name\_\_)

# configure the SQLite database, relative to the app instance folder

app.config["SQLALCHEMY\_DATABASE\_URI"] = "sqlite:////Users/yasin/Desktop/Yeni klasör/TodoApp/todo.db"

# initialize the app with the extension

db.init\_app(app)

class Todo(db.Model):

    id = db.Column(db.Integer, primary\_key=True)

    title = db.Column(db.String(80))

    complete = db.Column(db.Boolean)

with app.app\_context():

    db.create\_all()

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)

todo.py

**TODOLARI SQLALCHEMY İLE TABLOYA EKLEME**

ORM kullanarak veri ekleyeceğiz.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Todo App</title>

</head>

<body>

</body>

</html>

İndex.html

from flask import Flask, render\_template, redirect,url\_for

from flask\_sqlalchemy import SQLAlchemy

# create the extension

db = SQLAlchemy()

# create the app

app = Flask(\_\_name\_\_)

# configure the SQLite database, relative to the app instance folder

app.config["SQLALCHEMY\_DATABASE\_URI"] = "sqlite:////Users/yasin/Desktop/Yeni klasör/TodoApp/todo.db"

# initialize the app with the extension

db.init\_app(app)

@app.route('/')

def index():

    return render\_template('index.html')

class Todo(db.Model):

    id = db.Column(db.Integer, primary\_key=True)

    title = db.Column(db.String(80))

    complete = db.Column(db.Boolean)

with app.app\_context():

    db.create\_all()

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)

todo.py

[Introduction · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/getting-started/introduction/)

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

    <title>Todo App</title>

</head>

<body>

    <div style = 'margin-top: 100px' class = 'container'>

        <h3>Todo List</h3>

        <hr>

        <form action="/add" method='post'>

        </form>

    </div>

    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl" crossorigin="anonymous"></script>

</body>

</html>

İndex.html

[Forms · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/components/forms/)

        <form action="/add" method='post'>

            <div class="form-group">

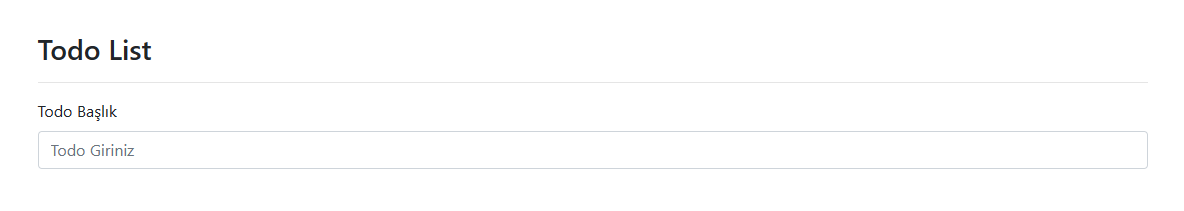
                <label for="formGroupExampleInput">Todo Başlık</label>

                <input type="text" class="form-control" name = 'title' id="formGroupExampleInput" placeholder="Todo Giriniz">

            </div>

        </form>

İndex.html



        <form action="/add" method='post'>

            <div class="form-group">

                <label for="formGroupExampleInput">Todo Başlık</label>

                <input type="text" class="form-control" name = 'title' id="formGroupExampleInput" placeholder="Todo Giriniz"><br>

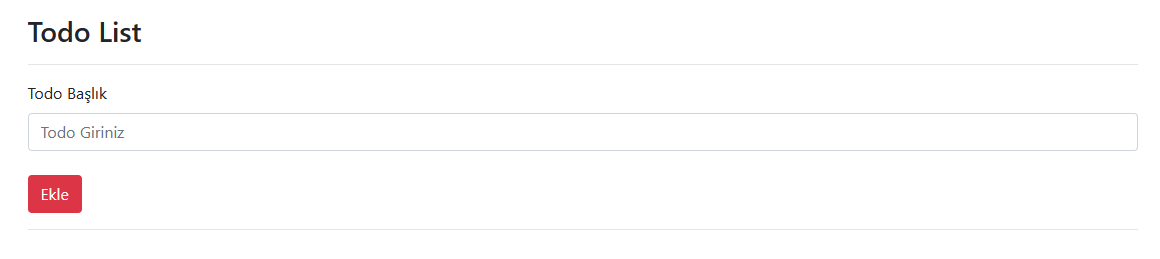
                <button type='submit' class='btn btn-danger'>Ekle</button>

                <hr>

            </div>

        </form>

İndex.html



Butona bastığımızda submit olacak. /add denen bir yere gidecek. Bu url’ye post yöntemiyle gideceğiz. Bu post yönteminden de name’si title olan alanı almam gerekiyor. Daha sonra veri tabanından eklememiz gerekiyor.

@app.route('/add',methods=[POST])

def addTodo():

    title = request.form.get('title')

    newTodo = Todo(title=title,complete=False)

todo.py

[Quick Start — Flask-SQLAlchemy Documentation (3.0.x) (palletsprojects.com)](https://flask-sqlalchemy.palletsprojects.com/en/3.0.x/quickstart/)

@app.route('/add',methods=[POST])

def addTodo():

    title = request.form.get('title')

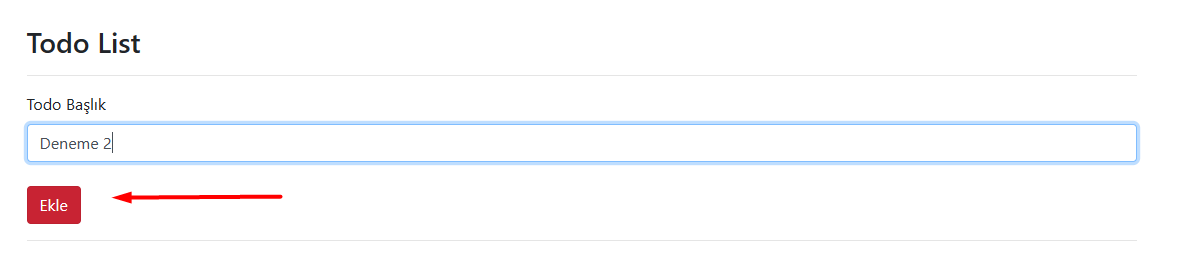
    newTodo = Todo(title=title,complete=False)

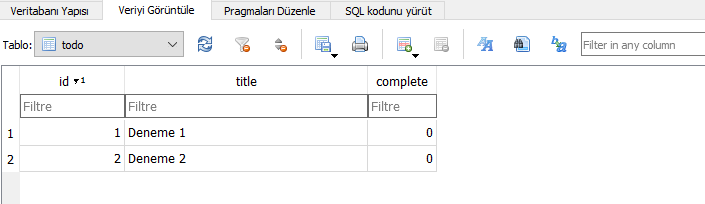
    db.session.add(newTodo)

    db.session.commit()

    return redirect(url\_for('index'))

todo.py





Complete değerleri sıfır çünkü complete=False olarak başlattık.

**TODOLARI ARAYÜZDE GÖSTERME**

[Tables · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/content/tables/)

       <table class="table">

            <thead>

                <tr>

                <th scope="col">id</th>

                <th scope="col">Todo Başlık</th>

                <th scope="col">Todo Durum</th>

                <th scope="col">Durum Güncelle</th>

                <th scope="col">Todo Sil</th>

                </tr>

            </thead>

            <tbody>

            </tbody>

        </table>

İndex.html

            <tbody>

                {% for todo in todos %}

                <tr>

                    <th scope="row">{{todo.id}}</th>

                    <td>{{todo.title}}</td>

                    <td>{{todo.complete}}</td>

                    <td><a href="/complete/{{todo.id}}" class='btn btn-danger'>Tamamla</a></td>

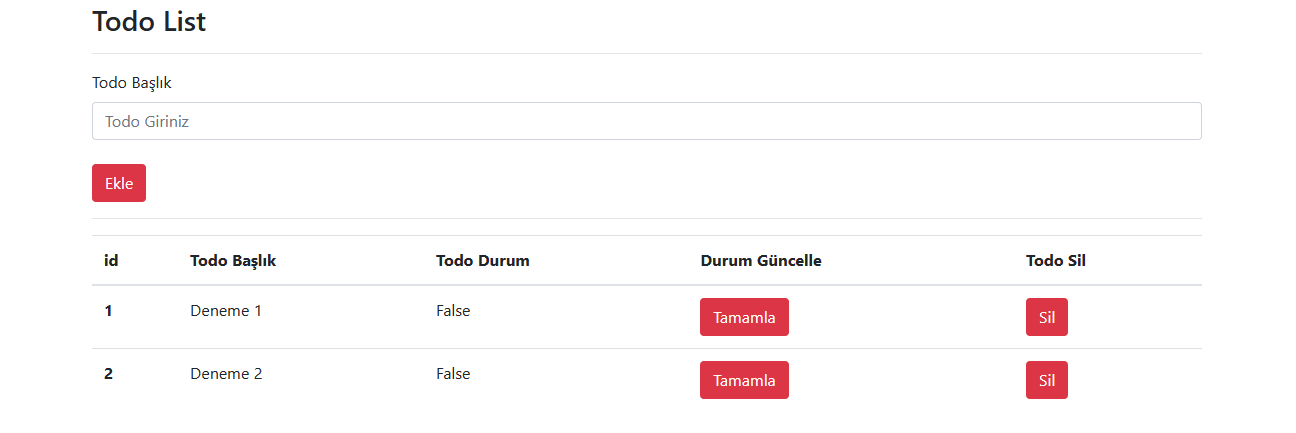
                    <td><a href="/delete/{{todo.id}}" class='btn btn-danger'>Sil</a></td>

                </tr>

                {% endfor %}

            </tbody>

İndex.html



            <tbody>

                {% for todo in todos %}

                <tr>

                    <th scope="row">{{todo.id}}</th>

                    <td>{{todo.title}}</td>

                    {% if todo.complete == False %}

                    <td>Tamamlanmadı</td>

                    {% else %}

                    <td>Tamamlandı</td>

                    {% endif %}

                    <td><a href="/complete/{{todo.id}}" class='btn btn-danger'>Tamamla</a></td>

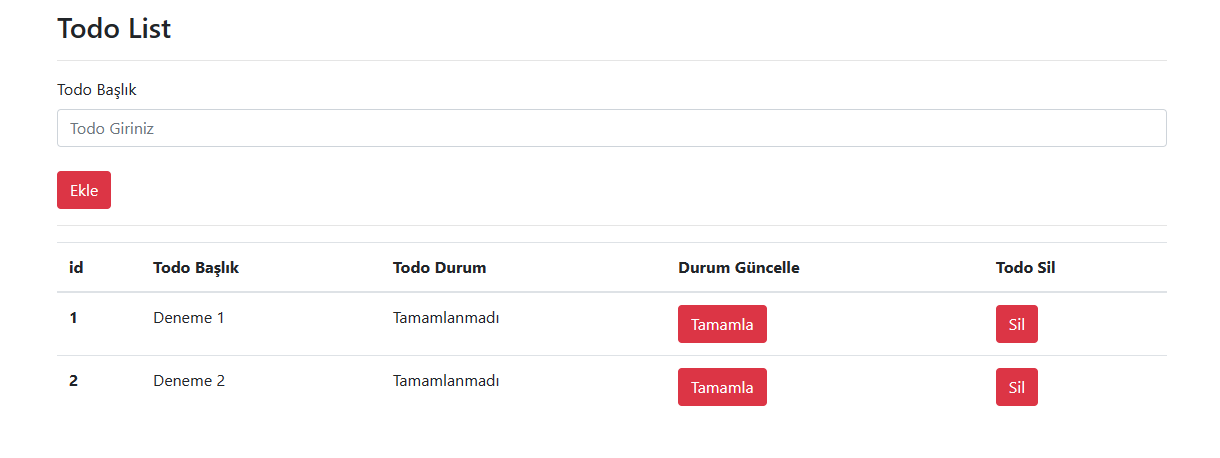
                    <td><a href="/delete/{{todo.id}}" class='btn btn-danger'>Sil</a></td>

                </tr>

                {% endfor %}

            </tbody>

İndex.html



**TODOLARI TAMAMLAMA**

@app.route('/complete/<string:id>')

def completeTodo(id):

    todo =Todo.query.filter\_by(id=id).first()

    """if todo.complete == True:

        todo.complete = False

    else:

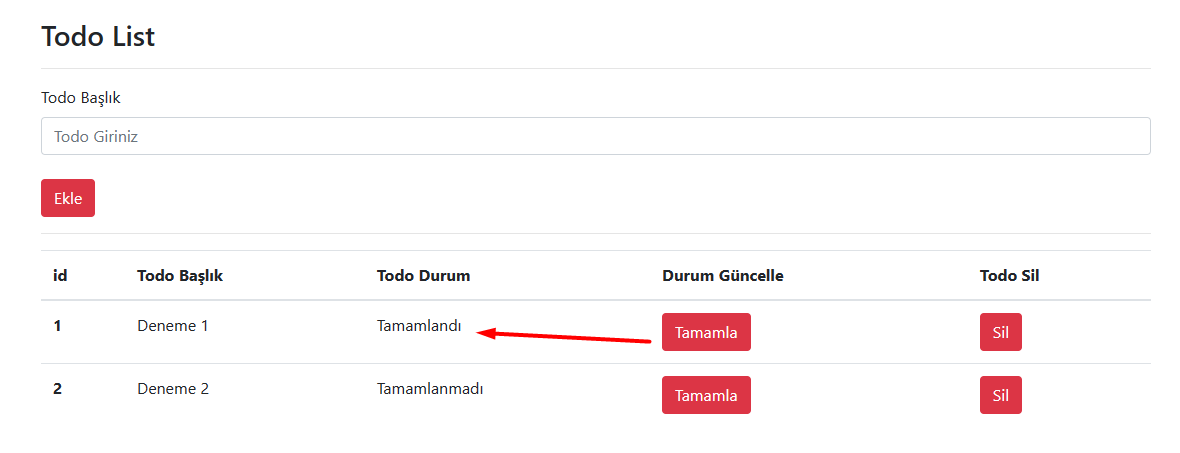
        todo.complete = True"""

    todo.complete = not todo.complete

    db.session.commit()

    return redirect(url\_for('index'))

todo.py



**Todoları Silme**

@app.route('/delete/<string:id>')

def deleteTodo(id):

    todo =Todo.query.filter\_by(id=id).first()

    db.session.delete(todo)

    db.session.commit()

    return redirect(url\_for('index'))

todo.py

        {% if todos==[] %}

        <div class="alert alert-warning" role="alert">

        Herhani bir todonuz bulunmuyor...

        </div>

        {% else %}

        <table class="table">

            <thead>

                <tr>

                <th scope="col">id</th>

                <th scope="col">Todo Başlık</th>

                <th scope="col">Todo Durum</th>

                <th scope="col">Durum Güncelle</th>

                <th scope="col">Todo Sil</th>

                </tr>

            </thead>

            <tbody>

                {% for todo in todos %}

                <tr>

                    <th scope="row">{{todo.id}}</th>

                    <td>{{todo.title}}</td>

                    {% if todo.complete == False %}

                    <td>Tamamlanmadı</td>

                    {% else %}

                    <td>Tamamlandı</td>

                    {% endif %}

                    <td><a href="/complete/{{todo.id}}" class='btn btn-danger'>Tamamla</a></td>

                    <td><a href="/delete/{{todo.id}}" class='btn btn-danger'>Sil</a></td>

                </tr>

                {% endfor %}

            </tbody>

        </table>

        {% endif %}

İndex.html. Todo bulunmazsa bir uyarı alıyoruz.