```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>2.6.4</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>
    <groupId>com.simplilearn</groupId>
    <artifactId>Phase4</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>Phase4</name>
    <description>Simplilearn Phase 4 Project</description>
    <properties>
        <java.version>11</java.version>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-data-jpa</artifactId>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-thymeleaf</artifactId>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-web</artifactId>
        </dependency>

        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-devtools</artifactId>
            <scope>runtime</scope>
            <optional>true</optional>
        </dependency>
        <dependency>
            <groupId>org.mariadb.jdbc</groupId>
            <artifactId>mariadb-java-client</artifactId>
            <scope>runtime</scope>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-test</artifactId>
            <scope>test</scope>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <plugin>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-maven-plugin</artifactId>
            </plugin>
        </plugins>
    </build>

</project>
```

```java
package com.simplilearn.Phase4.controller;

import com.simplilearn.Phase4.entity.Cart;
import com.simplilearn.Phase4.service.CartService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RestController;

import java.util.List;

@RestController
public class CartController {

    @Autowired
    CartService cs;

    @RequestMapping(value = "/addToCart", method = RequestMethod.POST, consumes = "application/json")
    public void addToCart(@RequestBody com.simplilearn.Phase4.entity.Cart cart){
        cs.saveCart(cart);
    }

    @RequestMapping(value = "/delCart", method = RequestMethod.GET)
    public void delCart(){
        cs.delCart();
    }

    @RequestMapping(value = "/getCart", method = RequestMethod.GET)
    public List<Cart> getCart(){
        List<Cart> ls = cs.getCart();
        return ls;
    }
}

package com.simplilearn.Phase4.controller;

import com.simplilearn.Phase4.entity.Product;
import com.simplilearn.Phase4.service.ProductService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RestController;

import java.util.List;

@RestController
public class Products {

    @Autowired
    ProductService ps;

    @RequestMapping(value = "/getProducts", method = RequestMethod.GET)
    public List<Product> getProducts(){

        List<Product> lp;
        lp = ps.getProducts();
        return lp;
    }
    @RequestMapping(value = "/postProducts", method = RequestMethod.POST, consumes = "application/json")
    public void postProducts(@RequestBody Product product){
        ps.saveProducts(product);
    }

    @RequestMapping(value = "/delProduct", method = RequestMethod.POST, consumes = "application/json")
    public void delProduct(@RequestBody Product product){
        ps.delProduct(product);
    }

    @RequestMapping(value = "/updateEnabled", method = RequestMethod.POST, consumes = "application/json")
    public void updateEnabled(@RequestBody Product product){
        ps.updateEnabled(product);
    }

}

package com.simplilearn.Phase4.controller;

import com.simplilearn.Phase4.entity.User;
import com.simplilearn.Phase4.repository.UserRepository;
import com.simplilearn.Phase4.service.UserService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.*;
import org.springframework.web.servlet.ModelAndView;

import java.util.Optional;

@Controller
public class Welcome {

    @Autowired
    UserService us;

    @RequestMapping("/")
    public String welcome(Model model) {
        return "index";
    }

    @GetMapping("/test")
    public String test(Model model){
        return "test";
    }

    @GetMapping("/userMain")
    public String userMain(Model model){
        return "UserMain";
    }

    @GetMapping("/userRegister")
    public String userRegister(Model model){
        return "UserRegister";
    }

    @GetMapping("/userLogin")
    public String userLogin(Model model){
        return "UserLogin";
```

```
}

@ResponseBody
@PostMapping("/userHome")
public String userHome(@ModelAttribute("user") User user, Model model)
{
    String email = user.getEmail();
    String pw = user.getPassword();

    boolean auth = us.authenticateUser(email, pw);
    System.out.println("Auth: " + auth);

    if(auth == true) {
        return "success";
    }else {
        return "error";
    }
}
@GetMapping("/userHome")
public String userHome(Model model){
    return "UserHome";
}

@GetMapping("/adminMain")
public String adminMain(Model model){
    return "AdminMain";
}

@GetMapping("/adminLogin")
public String adminLogin( Model model )
{
    return "AdminLogin";
}



@GetMapping("/showCart")
public String showCart( Model model )
{
    return "Cart";
}

@ResponseBody
@PostMapping("/adminHome")
public String adminHome(@ModelAttribute("user") User user, Model model){

    String email = user.getEmail();
    String pw = user.getPassword();

    boolean auth = us.authenticateUser(email, pw);

    if(auth == true) {
         return "success";
    }else {
        return "error";
    }
}

@GetMapping("/adminHome")
public String adminHome( Model model )
{
```

```java
        return "AdminHome";
    }

    @ResponseBody
    @PostMapping("/userRegister")
    public String userRegister(@ModelAttribute("user") User user, Model model){

        String email = user.getEmail();
        String pw = user.getPassword();
        String fname = user.getFname();
        String lname = user.getLname();

        boolean reg = us.registerUser(fname, lname, email, pw);

        if(reg) {
            return "success";
        }else {
            return "error";
        }
    }
}

package com.simplilearn.Phase4.entity;

import javax.persistence.*;
import java.util.ArrayList;
import java.util.List;

@Entity
@Table(name="cart_table")
public class Cart {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name="cart_id")
    private int cart_id;

    @Column(name="quantity")
    private int quantity;

    @Column(name="product_id")
    private int product_id;

    @Column(name="name")
    private String name;

    @Column(name="company")
    private String company;

    @Column(name="units")
    private int units;

    @Column(name="price", precision=10, scale=2)
    private float price;

    @Column(name="enabled")
    private boolean enabled;

    public float getTotal() {
        return this.price * this.quantity;
    }

    public int getProduct_id() {
        return product_id;
    }

    public void setProduct_id(int product_id) {
        this.product_id = product_id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getCompany() {
        return company;
    }

    public void setCompany(String company) {
        this.company = company;
    }

    public int getUnits() {
        return units;
    }

    public void setUnits(int units) {
        this.units = units;
    }

    public float getPrice() {
        return price;
    }

    public void setPrice(float price) {
        this.price = price;
    }

    public boolean isEnabled() {
        return enabled;
    }

    public void setEnabled(boolean enabled) {
        this.enabled = enabled;
    }

    public int getCart_id() {
        return cart_id;
    }

    public void setCart_id(int cart_id) {
        this.cart_id = cart_id;
    }

    public int getQuantity() {
```

```java
        return quantity;
    }

    public void setQuantity(int quantity) {
        this.quantity = quantity;
    }
}

package com.simplilearn.Phase4.entity;

import javax.persistence.*;

@Entity
@Table(name="product")
public class Product {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name="product_id")
    private int product_id;

    @Column(name="name")
    private String name;

    @Column(name="company")
    private String company;

    @Column(name="units")
    private int units;

    @Column(name="price", precision=10, scale=2)
    private float price;

    @Column(name="enabled")
    private boolean enabled;

    public int getProduct_id() {
        return product_id;
    }

    public void setProduct_id(int product_id) {
        this.product_id = product_id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getCompany() {
        return company;
    }

    public void setCompany(String company) {
        this.company = company;
    }
    public int getUnits() {
        return units;
    }

    public void setUnits(int units) {
        this.units = units;
    }

    public float getPrice() {
        return price;
    }

    public void setPrice(float price) {
        this.price = price;
    }

    public boolean isEnabled() {
        return enabled;
    }

    public void setEnabled(boolean enabled) {
        this.enabled = enabled;
    }
}

package com.simplilearn.Phase4.entity;

import javax.persistence.*;
import java.util.List;

@Entity
@Table(name="user")
public class User {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column(name="fname")
    private String fname;

    @Column(name="lname")
    private String lname;

    @Column(name="email")
    private String email;

    @Column(name="password")
    private String password;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }
```

```java
    public String getFname() {
        return fname;
    }

    public void setFname(String fname) {
        this.fname = fname;
    }

    public String getLname() {
        return lname;
    }

    public void setLname(String lname) {
        this.lname = lname;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
}

package com.simplilearn.Phase4.repository;

import com.simplilearn.Phase4.entity.Cart;
import com.simplilearn.Phase4.entity.Product;
import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Repository;

import java.util.List;

@Repository
public interface CartRepository extends CrudRepository<Cart, Integer> {



    //public User findByName(String name);
    List<Cart> findAll();
}
package com.simplilearn.Phase4.repository;

import com.simplilearn.Phase4.entity.Product;
import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Repository;

import java.util.List;

@Repository
public interface ProductRepository extends CrudRepository<Product, Integer> {
        //public User findByName(String name);
        List<Product> findAll();
}



package com.simplilearn.Phase4.repository;

import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Repository;
import com.simplilearn.Phase4.entity.User;

import java.util.List;

@Repository
public interface UserRepository extends CrudRepository<User, Integer> {



    //public User findByName(String name);
    List<User> findAll();

    User save(User user);
}

package com.simplilearn.Phase4.service;

import com.simplilearn.Phase4.entity.Cart;
import com.simplilearn.Phase4.repository.CartRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.List;

@Service
public class CartService {

    @Autowired
    CartRepository cr;

    public Cart saveCart(Cart cart){
        Cart c = cr.save(cart);
        return c;
    }

    public void delCart(){
        cr.deleteAll();
    }

    public List<Cart> getCart(){
        List<Cart> cl = cr.findAll();
        return cl;
    }
}

package com.simplilearn.Phase4.service;

import com.simplilearn.Phase4.entity.Cart;
import com.simplilearn.Phase4.entity.Product;
import com.simplilearn.Phase4.repository.CartRepository;
import com.simplilearn.Phase4.repository.ProductRepository;
import com.simplilearn.Phase4.repository.UserRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.ArrayList;
import java.util.List;
import java.util.Optional;

@Service
public class ProductService {

    @Autowired
    ProductRepository productRepository;



    public List<Product> getProducts(){
        List<Product> products = new ArrayList<>();
        products = productRepository.findAll();

        return products;
    }

    public Product saveProducts(Product product){
        //List<Product> products = new ArrayList<>();
        Product p = productRepository.save(product);

        return p;
    }

    public void delProduct(Product product){
        int id = product.getProduct_id();
        productRepository.deleteById(id);
    }

    public void updateEnabled(Product product){
        int id = product.getProduct_id();
        boolean enabled = product.isEnabled();

        Optional<Product> p = productRepository.findById(id);
        Product pr = p.get();

        System.out.println("enabled" + enabled);
        pr.setEnabled(enabled);
        productRepository.save(pr);
    }
}

package com.simplilearn.Phase4.service;

import com.simplilearn.Phase4.entity.User;
import com.simplilearn.Phase4.repository.UserRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.ArrayList;
import java.util.List;
```

```
@Service
public class UserService {

    @Autowired
    UserRepository userRepository;

    public boolean authenticateUser(String email, String pw) {

        List<User> users = new ArrayList<>();
        users = userRepository.findAll();

        for (User user : users) {
            if (user.getEmail().equals(email) && user.getPassword().equals(pw)) {
                return true;
            }
        }
        return false;
    }

    public boolean registerUser(String fname, String lname, String email, String password){

        User user = new User();
        user.setFname(fname);
        user.setLname(lname);
        user.setEmail(email);
        user.setPassword(password);

        List<User> users;
        users = userRepository.findAll();

        boolean exists = false;

        for(User us : users){
            if(us.getEmail() == user.getEmail()){
                exists = true;
            }
        }

        if(!exists) {
            User ret = userRepository.save(user);
        }

        if(!exists){
            return true;
        }else{
            return false;
        }
    }
}

function validateEmailWithRegex(email) {
    let emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

    return !!email && typeof email === "string" && email.match(emailRegex);
  }

function escapeHtml(unsafe) {
   return unsafe
      .replace(/&/g, "&amp;")
      .replace(/</g, "&lt;")
      .replace(/>/g, "&gt;")
      .replace(/"/g, "&quot;")
      .replace(/'/g, "&#039;");
  }
function validatePassword(pw){
    var strongRegex = new RegExp("^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\$%\^&\*])(?=.{8,})");

    return strongRegex.test(pw);
}

  function validateInputs() {
    let forward = false;

    let email = document.getElementById("email").value;

    // input sanitization
    email = email.trim();
    email = escapeHtml(email);

    let validEmail = validateEmailWithRegex(email);
    if(validEmail == false){
        document.getElementById('emailInvalid').className = "text-danger visible"
        forward = false;
    }else{
        document.getElementById('emailInvalid').className = "invisible"
        forward = true
    }

    let pw =document.getElementById("password").value;

    // input sanitization
    pw = pw.trim();
    pw = escapeHtml(pw);

    let validPW = validatePassword(pw);
    if(validPW){
        document.getElementById("pwInvalid").className = "invisible"
        forward = true;
    }else{
        document.getElementById('pwInvalid').className = "text-danger visible"
        forward=false;
    }
    if(forward){

    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
                if (xhttp.readyState == XMLHttpRequest.DONE) {
                    if("success" == xhttp.responseText){
                        window.location.href = "adminHome"
                    }else{
                        window.location.href = "adminLogin"
                    }


                }
            }

    xhttp.open("POST", "/adminHome", true);
    var user = new FormData();
    user.append("email", email);
    user.append("password", pw);
  xhttp.send(user);
  }
 }

$(document).ready(function() {
 let table = $('#table_id').DataTable({
  "ajax": {
         "url": "getProducts",
         "type": "GET",
         "contentType": "application/json",
         "dataSrc": "
        },

    "columns": [
                 {'data': 'product_id',
                 "defaultContent": null,
                 "visible": false},
                 {'data': 'name'},
                 {'data': 'company'},
                 {'data': 'units'},
                 {'data': 'price'},
                 {'data': 'enabled',
                 "defaultContent": true,
                 "visible": false,},
                 {
                 "targets": [-2],
                 "data": null,
                 "defaultContent": "<input id='enable' type='checkbox' checked data-toggle='toggle'> Enabled</input>"
                 },
                 {
                 "targets": [-1],
                 "data": null,
                 "defaultContent": "<button id='delRow'>Delete</button>"
                 }
               ],
  } );

  $('#addRow').on('click', function(){
  //Send Data from form to backend and reload table
         let name = document.getElementById("product").value;
         let company = document.getElementById("company").value;
         let units = document.getElementById("units").value;
         let price = document.getElementById("price").value;

         if(name == null || name == "" ||
            company == null || company == "" ||
            units == null || units == "" ||
            price == null || price == "")
         {
         alert("All values neccessary")
         return
         }

         let row = {product_id:null, name: name, company:company, units:units, price:price, enabled : true}
         $.ajax({
            "dataType":"json",
            'contentType': 'application/json',
            "type": "POST",
            "url": "postProducts",
            "data": JSON.stringify(row)
         });
         table.ajax.reload();
        });

$('#table_id tbody').on( 'click', '#delRow', function () {
    //Delete Row of Data from Backend and datatable
    let rowData = table.row( $(this).parents('tr') ).data();

    $.ajax({
        "dataType":"json",
        'contentType': 'application/json',
        "type": "POST",
        "url": "delProduct",
        "data": JSON.stringify(rowData)
     });
    table
    .row( $(this).parents('tr') )
    .remove()
    .draw();
});

$('#table_id tbody').on( 'click', '#enable', function () {
//todo update database with enabled/disabled
        //update boolean cell with true or false
        let rowData = table.row( $(this).parents('tr') ).data();
        let idx = table.row( $(this).parents('tr')).index()

        if($(this).prop("checked"))
        {
            table.cell(idx, 5).data(true);
            rowData.enabled = true
        }else{
            table.cell(idx, 5).data(false);
            rowData.enabled = false
        }

        $.ajax({
            "dataType":"json",
            'contentType': 'application/json',
            "type": "POST",
            "url": "updateEnabled",
            "data": JSON.stringify(rowData)
        });


    });
});
$(document).ready(function() {

 var table = $('#table_id').DataTable({
 //todo get enabled products from backend
  "ajax": {
                     "url": "getCart",
                     "type": "GET",
                     "contentType": "application/json",
                     "dataSrc": '',
                   },
     "columns": [
                  {'data': 'cart_id',
                   "visible": false},
                  {'data': 'product_id',
                  "visible": false},
                  {'data': 'name'},
                  {'data': 'company'},
                  {'data': 'units',
                  "visible": false},
                  {'data': 'price'},
                  {'data': 'enabled',
                  "visible": false},
                  {'data': 'quantity'},
                  {'data': 'total',
                  "defaultContent": 0}
                ]
    } );
});



function validateEmailWithRegex(email) {
    let emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

    return !!email && typeof email === "string" && email.match(emailRegex);
  }

function escapeHtml(unsafe) {
  return unsafe
     .replace(/&/g, "&amp;")
     .replace(/</g, "&lt;")
     .replace(/>/g, "&gt;")
     .replace(/"/g, "&quot;")
     .replace(/'/g, "&#039;");
  }
function validatePassword(pw){
    var strongRegex = new RegExp("^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\$%\^&\*])(?=.{8,})");

    return strongRegex.test(pw);
}

  function validateInputs() {
    //console.log("in validate function")
    let forward = false;

    let email = document.getElementById("email").value;

    // input sanitization
    email = email.trim();
    email = escapeHtml(email);

    let validEmail = validateEmailWithRegex(email);
    if(validEmail == false){
        document.getElementById('emailInvalid').className = "text-danger visible"
        forward = false;
    }else{
        document.getElementById('emailInvalid').className = "invisible"
        forward = true
    }

    let pw =document.getElementById("password").value;

    // input sanitization
    pw = pw.trim();
    pw = escapeHtml(pw);

    let validPW = validatePassword(pw);
    if(validPW){
        document.getElementById("pwInvalid").className = "invisible"
        forward = true;
    }else{
        document.getElementById('pwInvalid').className = "text-danger visible"
        forward=false;
    }
    if(forward){
      var xhttp = new XMLHttpRequest();
        xhttp.onreadystatechange = function() {
            if (xhttp.readyState == XMLHttpRequest.DONE) {
                if("success" == xhttp.responseText){
                    window.location.href = "userHome"
                }else{
                    window.location.href = "userLogin"
                }
            }
        }
        xhttp.open("POST", "/userHome", true);
        var user = new FormData();
        user.append("email", email);
        user.append("password", pw);
        xhttp.send(user);
    }
  }
function validateEmailWithRegex(email) {
  let emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

  return !email && typeof email === "string" && email.match(emailRegex);
}

function checkInputEmail(email) {
  let isValidEmail = validateEmail(email);

  if (isValidEmail) {
    document.getElementById("emailHelp").remove();
    alert("Valid email!");
  } else {
    document.getElementById("emailHelp").innerText = "Invalid email!";
  }
}

function validateName(fname, lname){

  if(null == fname || null == lname){
      return null;
  }

  if (""== fname || "" == lname){
      return false;
  }

  return true;
}

function escapeHtml(unsafe) {
  return unsafe
    .replace(/&/g, "&amp;")
    .replace(/</g, "&lt;")
    .replace(/>/g, "&gt;")
    .replace(/"/g, "&quot;")
    .replace(/'/g, "&#039;");
```

```
}

function validateInputs() {
 let email = document.getElementById("email").value;

 let forward = false
 // input sanitization
 email = email.trim();
 email = escapeHtml(email);

 let validEmail = validateEmailWithRegex(email);
 if(validEmail == false){
   document.getElementById('emailInvalid').className = "text-danger visible"
   forward = false
 }else{
   document.getElementById('emailInvalid').className = "invisible"
   forward = true
 }

 let fname = document.getElementById("fname").value
 let lname = document.getElementById("lname").value

 // input sanitization
 fname = fname.trim();
 fname = escapeHtml(fname);

 // input sanitization
 lname = lname.trim();
 lname = escapeHtml(lname);

 let validName = validateName(fname, lname);
 if (validName == false){
   document.getElementById("nameInvalid").className = "text-danger visible"
   forward = false
 }
 else{
   document.getElementById("nameInvalid").className = "invisible"
   forward = true
 }

 let pw = document.getElementById("password").value

 pw = pw.trim()
 pw = escapeHtml(pw)

 if(forward){
  //console.log("in submit form")
    var xhttp = new XMLHttpRequest();
     xhttp.onreadystatechange = function() {
       if (xhttp.readyState == XMLHttpRequest.DONE) {
         if("success" == xhttp.responseText){
           window.location.href = "userLogin"
         }else{
           window.location.href = "userRegister"
         }
       }
     }



     xhttp.open("POST", "/userRegister", true);
```

```
        var user = new FormData();
        user.append("email", email);
        user.append("password", pw);
        user.append("fname", fname);
        user.append("lname", lname);
        xhttp.send(user);
  }
}
$(document).ready(function() {
 let table = $('#table_id').DataTable({
 //todo get enabled products from backend
 //todo show units in table and that it's not possible to order more
  "ajax": {
                 "url": "getProducts",
                 "type": "GET",
                 "contentType": "application/json",
                 "dataSrc": ''
                },
    "columns": [
                {'data': 'cart_id',
                 "visible": false,
                 "defaultContent": "notSet"},
                {'data': 'product_id',
                "visible": false},
                {'data': 'name'},
                {'data': 'company'},
                {'data': 'units'},
                {'data': 'price'},
                {'data': 'enabled',
                "visible": false},
                {'data': 'quantity',
                "defaultContent": 0,
                "visible": false},
                {
                "targets": [-2],
                "data": null,
                "defaultContent": "<input id='quantity' type='number'></input>"
                },
                {"data:":"total",
                "defaultContent": 0},
                {
                "targets": [-1],
                "data": null,
                "defaultContent": "<button id='addToCart'>Add to Cart</button>"
                }
              ]
   } );

   $('#table_id tbody').on( 'click', 'tr', function () {
         if ( $(this).hasClass('selected') ) {
             $(this).removeClass('selected');
         }
         else {
             table.$('tr.selected').removeClass('selected');
             $(this).addClass('selected');
         }
     } );

   $('#table_id tbody').on( 'click', '#addToCart', function () {
     //Add Row to Database
     let rowData = table.row( $(this).parents('tr') ).data();
     rowData.quantity = $(this).parents('tr').find("input").val();
```

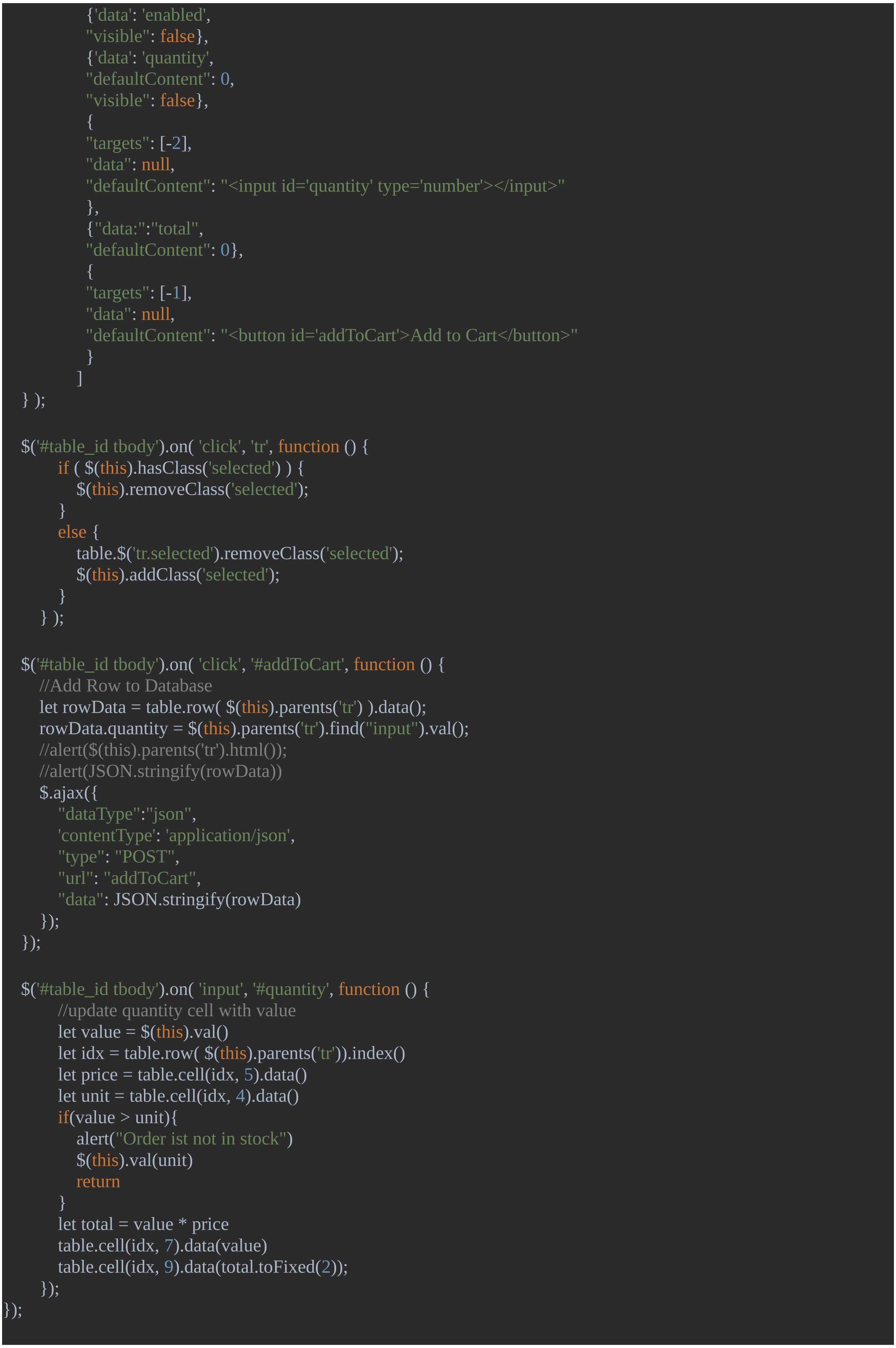
```
        //alert($(this).parents('tr').html());
        //alert(JSON.stringify(rowData))
        $.ajax({
            "dataType":"json",
            'contentType': 'application/json',
            "type": "POST",
            "url": "addToCart",
            "data": JSON.stringify(rowData)
        });
    });

    $('#table_id tbody').on( 'input', '#quantity', function () {
            //update quantity cell with value
            let value = $(this).val()
            let idx = table.row( $(this).parents('tr')).index()
            let price = table.cell(idx, 5).data()
            let unit = table.cell(idx, 4).data()
            if(value > unit){
                alert("Order ist not in stock")
                $(this).val(unit)
                return
            }
            let total = value * price
            table.cell(idx, 7).data(value)
            table.cell(idx, 9).data(total.toFixed(2));
        });
});

document.getElementById('delCart').addEventListener("click", function() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (xhttp.readyState == XMLHttpRequest.DONE) {
        }
    }
    xhttp.open("GET", "/delCart", true);
    xhttp.send(null);
});

document.getElementById('showCart').addEventListener("click", function() {
    location.href ="showCart"
});

$(document).ready(function() {
 let table = $('#table_id').DataTable({
 //todo get enabled products from backend
 //todo show units in table and that it's not possible to order more
  "ajax": {
                    "url": "getProducts",
                    "type": "GET",
                    "contentType": "application/json",
                    "dataSrc": ''
                  },
     "columns": [
                  {'data': 'cart_id',
                   "visible": false,
                   "defaultContent": "notSet"},
                  {'data': 'product_id',
                  "visible": false},
                  {'data': 'name'},
                  {'data': 'company'},
                  {'data': 'units'},
                  {'data': 'price'},
```



```
          {'data': 'enabled',
          "visible": false},
          {'data': 'quantity',
          "defaultContent": 0,
          "visible": false},
          {
          "targets": [-2],
          "data": null,
          "defaultContent": "<input id='quantity' type='number'></input>"
          },
          {"data:":"total",
          "defaultContent": 0},
          {
          "targets": [-1],
          "data": null,
          "defaultContent": "<button id='addToCart'>Add to Cart</button>"
          }
        ]
  } );

  $('#table_id tbody').on( 'click', 'tr', function () {
      if ( $(this).hasClass('selected') ) {
        $(this).removeClass('selected');
      }
      else {
        table.$('tr.selected').removeClass('selected');
        $(this).addClass('selected');
      }
    } );

  $('#table_id tbody').on( 'click', '#addToCart', function () {
    //Add Row to Database
    let rowData = table.row( $(this).parents('tr') ).data();
    rowData.quantity = $(this).parents('tr').find("input").val();
    //alert($(this).parents('tr').html());
    //alert(JSON.stringify(rowData))
    $.ajax({
      "dataType":"json",
      'contentType': 'application/json',
      "type": "POST",
      "url": "addToCart",
      "data": JSON.stringify(rowData)
    });
  });

  $('#table_id tbody').on( 'input', '#quantity', function () {
      //update quantity cell with value
      let value = $(this).val()
      let idx = table.row( $(this).parents('tr')).index()
      let price = table.cell(idx, 5).data()
      let unit = table.cell(idx, 4).data()
      if(value > unit){
        alert("Order ist not in stock")
        $(this).val(unit)
        return
      }
      let total = value * price
      table.cell(idx, 7).data(value)
      table.cell(idx, 9).data(total.toFixed(2));
    });
});
```

```
document.getElementById('delCart').addEventListener("click", function() {
   var xhttp = new XMLHttpRequest();
   xhttp.onreadystatechange = function() {
      if (xhttp.readyState == XMLHttpRequest.DONE) {
      }
   }
   xhttp.open("GET", "/delCart", true);
   xhttp.send(null);
});

document.getElementById('showCart').addEventListener("click", function() {
   location.href ="showCart"
});

<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
   <title>Document</title>
</head>
<body>
   <main class="row">
      <div class="col-5 my-5 mx-auto align-self-center">
         <nav class="navbar navbar-expand-lg navbar-light bg-light">
            <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
         </nav>
        <div class="card card-block">
         <header class="text-center">
           <h1 class="text-primary text-uppercase">Admin Login</h1>
           <h6 class="text-secondary">Enter your credentials</h6>
         </header>
         <hr />

         <section>
           <form th:action="@{/adminHome}" method="post" class="font-weight-bold">
             <!-- Email -->
             <div id="emailField" name="emailField" class="form-group row">
               <label for="email" class="col-sm-4 ml-4 col-form-label"
                 >Email:*</label
               >
               <div class="col-sm-6">
                 <input
                   type="email"
                   id="email"
                   name="email"
                   class="form-control"
                   placeholder="abc@xyz.com"
                   required
                 />
                 <small id="emailInvalid" class="invisible">Invalid Email</small>
               </div>
             </div>


             <!-- Password -->
             <div
               id="passwordField"
         name="passwordField"
         class="form-group row"
        >
         <label for="password" class="col-sm-4 ml-4 col-form-label"
          >Password:*</label
         >
         <div class="col-sm-6">
          <input
           type="password"
           id="password"
           name="password"
           class="form-control"
           required
          />
          <small id="pwInvalid" class="invisible">Invalid Password</small>
         </div>
        </div>

        <!-- Stay signed in checkbox -->
        <div
         id="staySignedInCheck"
         name="staySignedInCheck"
         class="form-group font-weight-normal row"
        >
         <div class="col-sm-4 ml-4"></div>
         <div class="col-sm-6">
          <div class="form-check form-check-inline">
           <input
            type="checkbox"
            name="staySignedIn"
            id="staySignedIn"
            class="form-check-input"
           />
           <label for="staySignedIn" class="form-check-label"
            >Keep me signed in</label
           >
          </div>
         </div>
        </div>

        <!-- Submit Button -->
        <div id="loginButton" class="row">
         <div class="col-sm-10 offset-sm-1">
          <input
           type="button"
           value="Login"
           class="btn btn-block btn-primary mb-2"
           onClick="validateInputs()"
           />
         </div>
        </div>
       </form>
      </section>
     </div>
    </div>
   </main>
  <script src="../js/AdminLoginValidation.js"></script>
  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
```

```
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <title>Document</title>
</head>
<body>
    <main class="col-md-8 mx-auto">
        <section>
            <nav class="navbar navbar-expand-lg navbar-light bg-light">
                <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
                <div class="collapse navbar-collapse flex-grow-1 text-right">
                    <ul class="navbar-nav ml-auto flex-nowrap">
                        <li class="nav-item"><a class="nav-link" th:href="@{adminLogin}">Login <span class="sr-
only">(current)</span></a></li>
                    </ul>
                </div>
            </nav>
        </section>
        <section>
            <header class="text-center">
                <h1 class="text-primary text-uppercase">Hello Admin</h1>
            </header>
        </section>
    </main>

    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.11.4/css/jquery.dataTables.css">
    <title>Document</title>
</head>
<body>
<main class="row">
    <div class="col-10 my-5 mx-auto align-self-center">
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
      <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
      <div class="collapse navbar-collapse flex-grow-1 text-left">
        <ul class="navbar-nav ml-auto flex-nowrap">
          <li class="nav-item">Order Cart</li>
        </ul>
      </div>
      <div class="collapse navbar-collapse flex-grow-1 text-right">
        <ul class="navbar-nav ml-auto flex-nowrap">
          <li class="nav-item"><a class="nav-link" th:href="@{/}">Logout <span class="sr-
only">(current)</span></a></li>
        </ul>
      </div>
    </nav>
    <div>
      <section>
        <table id="table_id" class="display">
          <thead>
          <tr>
            <th>Cart ID</th>
            <th>Product ID</th>
            <th>Name</th>
            <th>Company</th>
            <th>Unit</th>
            <th>Price</th>
            <th>enabled</th>
            <th>Quantity</th>
            <th>Total</th>
          </tr>
          </thead>
          <tbody>
          <tr>
            <td></td>
            <td></td>
            <td></td>
            <td></td>
            <td></td>
            <td></td>
            <td></td>
            <td></td>
            <td></td>
          </tr>
          </tbody>
        </table>
      </section>
    </div>
  </div>
</main>

<script src="../vendor/jquery-3.6.0.min.js"></script>
<script type="text/javascript" charset="utf8" src="https://cdn.datatables.net/1.11.4/js/jquery.dataTables.js"></script>
<script src="https://cdn.datatables.net/buttons/2.2.2/js/dataTables.buttons.min.js"></script>
<script src="../js/cart.js"></script>
<!--script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script-->
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
```

```
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <title>Document</title>
</head>
<body>
    <main class="col-md-8 mx-auto">
        <header>
            <nav class="navbar navbar-expand-lg navbar-light bg-light">
                <a class="navbar-brand" href="#">Kitchen Story</a>
                <div class="collapse navbar-collapse flex-grow-1 text-right">

                    <ul class="navbar-nav ml-auto flex-nowrap">
                        <li class="nav-item"><a class="nav-link" th:href="@{userMain}">User Portal<span class="sr-
only">(current)</span></a></li>
                        <li class="nav-item"><a class="nav-link" th:href="@{adminMain}">Admin Portal<span class="sr-
only">(current)</span></a></li>
                    </ul>
                </div>
            </nav>
        </header>

        <section>

        </section>
    </main>
    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.11.4/css/jquery.dataTables.css">
    <title>Document</title>
</head>
<body>
<main class="row">
    <div class="col-10 my-5 mx-auto align-self-center">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
            <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
            <div class="collapse navbar-collapse flex-grow-1 text-left">
```

```
                <ul class="navbar-nav ml-auto flex-nowrap">
                    <li class="nav-item">User</li>
                </ul>
            </div>
            <div class="collapse navbar-collapse flex-grow-1 text-right">
                <ul class="navbar-nav ml-auto flex-nowrap">
                    <li class="nav-item"><a class="nav-link" th:href="@{/}">Logout <span class="sr-
only">(current)</span></a></li>
                </ul>
            </div>
        </nav>
        <div>
        <section>
            <div>
                <input type="button" id="showCart" value="Show Cart" class="btn btn-block btn-primary mb-2"/>
                <input type="button" id="delCart" value="Delete Cart" class="btn btn-block btn-primary mb-2"/>
            </div>
            <table id="table_id" class="display">
                <thead>
                <tr>
                    <th>Cart ID</th>
                    <th>Product ID</th>
                    <th>Name</th>
                    <th>Company</th>
                    <th>Unit</th>
                    <th>Price</th>
                    <th>enabled</th>
                    <th>Quantity</th>
                    <th>Quantity</th>
                    <th>Total</th>
                    <th></th>
                </tr>
                </thead>
                <tbody>
                <tr>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                </tr>
                </tbody>
            </table>
        </section>
        </div>
    </div>
</main>

<script src="../vendor/jquery-3.6.0.min.js"></script>
<script type="text/javascript" charset="utf8" src="https://cdn.datatables.net/1.11.4/js/jquery.dataTables.js"></script>
<script src="https://cdn.datatables.net/buttons/2.2.2/js/dataTables.buttons.min.js"></script>
<script src="../js/UserTable.js"></script>
<!--script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script-->
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
```

```
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
  <title>Document</title>
</head>
<body>
  <main class="row">
    <div class="col-5 my-5 mx-auto align-self-center">
      <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
      </nav>
     <div class="card card-block">
      <header class="text-center">
       <h1 class="text-primary text-uppercase">User Login</h1>
       <h6 class="text-secondary">Enter your credentials</h6>
      </header>
      <hr />

      <section>
       <form th:action="@{/userHome}" method="POST" class="font-weight-bold">
        <!-- Email -->
        <div id="emailField" name="emailField" class="form-group row">
         <label for="email" class="col-sm-4 ml-4 col-form-label"
          >Email:*</label
         >
         <div class="col-sm-6">
          <input
           type="email"
           id="email"
           name="email"
           class="form-control"
           placeholder="abc@xyz.com"
           required
          />
          <small id="emailInvalid" class="invisible">Invalid Email</small>

         </div>
        </div>

        <!-- Password -->
        <div
         id="passwordField"
         name="passwordField"
         class="form-group row"
        >
         <label for="password" class="col-sm-4 ml-4 col-form-label"
          >Password:*</label
         >
         <div class="col-sm-6">
          <input
```

```
           type="password"
           id="password"
           name="password"
           class="form-control"
           required
          />
          <small id="pwInvalid" class="invisible">Invalid Password</small>
         </div>
        </div>


        <!-- Stay signed in checkbox -->
        <div
         id="staySignedInCheck"
         name="staySignedInCheck"
         class="form-group font-weight-normal row"
        >
         <div class="col-sm-4 ml-4"></div>
         <div class="col-sm-6">
          <div class="form-check form-check-inline">
           <input
            type="checkbox"
            name="staySignedIn"
            id="staySignedIn"
            class="form-check-input"
           />
           <label for="staySignedIn" class="form-check-label"
            >Keep me signed in</label
           >
          </div>
         </div>
        </div>

        <!-- Submit Button -->
        <div id="loginButton" class="row">
         <div class="col-sm-10 offset-sm-1">
          <input
           type="button"
           value="Login"
           class="btn btn-block btn-primary mb-2"
           onClick="validateInputs()"
          />
         </div>
        </div>
       </form>
      </section>
     </div>
    </div>
   </main>
  <script src="../js/UserLoginValidation.js"></script>
  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
```

```
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
    <title>Document</title>
</head>
<body>
    <main class="col-md-8 mx-auto">
        <section>
            <nav class="navbar navbar-expand-lg navbar-light bg-light">
                <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
                <div class="collapse navbar-collapse flex-grow-1 text-right">

                    <ul class="navbar-nav ml-auto flex-nowrap">
                        <li class="nav-item"><a class="nav-link" th:href="@{userRegister}">Register <span class="sr-only">(current)</span></a></li>
                        <li class="nav-item"><a class="nav-link" th:href="@{userLogin}">Login <span class="sr-only">(current)</span></a></li>
                    </ul>
                </div>
            </nav>
        </section>
        <section>
            <header class="text-center">
                <h1 class="text-primary text-uppercase">Hello User</h1>
            </header>
        </section>
    </main>
    <section>

    </section>


    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>
</body>
</html>
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
    <title>Document</title>
</head>
<body>
    <main class="col-md-8 mx-auto">
        <section>
            <nav class="navbar navbar-expand-lg navbar-light bg-light">
```

```
                <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
                <div class="collapse navbar-collapse flex-grow-1 text-right">

                    <ul class="navbar-nav ml-auto flex-nowrap">
                        <li class="nav-item"><a class="nav-link" th:href="@{userRegister}">Register <span class="sr-
only">(current)</span></a></li>
                        <li class="nav-item"><a class="nav-link" th:href="@{userLogin}">Login <span class="sr-
only">(current)</span></a></li>
                    </ul>
                </div>
            </nav>
        </section>
        <section>
            <header class="text-center">
                <h1 class="text-primary text-uppercase">Hello User</h1>
            </header>
        </section>
    </main>
    <section>

    </section>


    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http:/www.thymeleaf.org">
<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://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <title>Document</title>
</head>
<body>
    <main class="col-md-5 mx-auto">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
            <a class="navbar-brand" th:href="@{/}">Kitchen Story</a>
            <div class="collapse navbar-collapse flex-grow-1 text-right">
                <ul class="navbar-nav ml-auto flex-nowrap">
                    <li class="nav-item"><a class="nav-link" th:href="@{userLogin}">Login <span class="sr-
only">(current)</span></a></li>
                </ul>
            </div>
        </nav>
        <header class="text-center">
          <h1 class="text-primary text-uppercase">Register</h1>
          <h6 class="text-secondary">Provide your details</h6>
        </header>
        <hr />
```

```
<section>
  <form action="register-backend" method="GET" class="font-weight-bold">
    <!-- Name (firstname and lastname) -->
    <div id="name" name="name" class="row form-group">
      <div id="firstNameField" name="nameField" class="col">
        <label for="fname">First name:*</label><br />
        <input
          type="text"
          id="fname"
          name="fname"
          class="form-control"
          autocomplete="off"
          autofocus
          required
        />
      </div>
      <div id="lastNameField" name="nameField" class="col">
        <label for="lname">Last name:</label><br />
        <input type="text" id="lname" name="lname" class="form-control" />
      </div>
    </div>
    <small id="nameInvalid" class="invisible">Name invalid</small>

    <!-- Email -->
    <div id="emailField" name="emailField" class="form-group">
      <label for="email">Email:*</label><br />
      <input
        type="email"
        id="email"
        name="email"
        class="form-control"
        placeholder="abc@xyz.com"
        required
      />
      <small id="emailFieldHelp" class="form-text text-muted"
        >Your email is safe with us :)</small>
      <small id="emailInvalid" class="invisible">Invalid Email</small>
    </div>

    <!-- Password -->
    <div id="passwordField" name="passwordField" class="form-group">
      <label for="password">Password:*</label><br />
      <input
        type="password"
        id="password"
        name="password"
        class="form-control"
        required
      />
      <small id="passwordHelp" class="form-text text-muted"
        ><ul>
          <li>Password should be between 8-16 characters</li>
          <li>
            Password should contain both lowercase and uppercase alphabets
          </li>
          <li>Password should contain at least one special symbol</li>
        </ul></small
      >
    </div>



    <!-- Submit Button -->
```

```
      <div id="registrationButton">
       <input
        type="button"
        value="Register"
        class="btn btn-block btn-primary mb-2"
        onClick="validateInputs()"
       />
      </div>
     </form>
    </section>
   </main>
  <script src="../js/UserRegistrationValidation.js"></script>
  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>
```