# Luxury Watch Store Webpage in Visual Studio Code:

**1. (Watches.html):**

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Watch Store</title>
    <link rel="stylesheet" href="/css/styles.css">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
    <header class="bg-dark text-white p-4">
        <div class="container">
            <h1>Luxury Watch Store</h1>
            <p class="lead">Timeless elegance on your wrist</p>
        </div>
    </header>
    <main class="container my-5">
        <div class="row">
            <div th:each="watch : ${watches}" class="col-md-6 col-lg-3 mb-4">
                <div class="card h-100">
                    <img th:src="${watch.imageUrl}" class="card-img-top" alt="Watch image">
                    <div class="card-body">
                        <h5 class="card-title" th:text="${watch.name}"></h5>
                        <h6 class="card-subtitle mb-2 text-muted" th:text="${watch.brand}"></h6>
                        <p class="card-text" th:text="${watch.description}"></p>
                    </div>
                    <div class="card-footer bg-white">
                        <div class="d-flex justify-content-between align-items-center">
                            <span class="h5" th:text="'$' + ${watch.price}"></span>
                            <button class="btn btn-primary">Add to Cart</button>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </main>
    <footer class="bg-dark text-white py-4 mt-5">
        <div class="container text-center">
            <p>&copy; 2023 Luxury Watch Store. All rights reserved.</p>
        </div>
    </footer>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

**2. (Styles.css):**

```
/* General Styles */
body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #f8f9fa;
    color: #333;
}
```

```
header {

   background: linear-gradient(135deg, #1a1a1a 0%, #343a40 100%);

   padding: 2rem 0;

   margin-bottom: 2rem;

}

.card {

   border: none;

   border-radius: 10px;

   overflow: hidden;

   box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

   transition: transform 0.3s ease, box-shadow 0.3s ease;

}

.card:hover {

   transform: translateY(-5px);

   box-shadow: 0 6px 12px rgba(0, 0, 0, 0.15);

}

.card-img-top {

   height: 200px;

   object-fit: cover;

}

.btn-primary {

   background-color: #343a40;

   border-color: #343a40;

}

.btn-primary:hover {

   background-color: #1a1a1a;

   border-color: #1a1a1a;
}
/* Responsive adjustments */
@media (max-width: 768px) {
  .card {
    margin-bottom: 20px;
  }
}
```

**3. (Watch.java):**

```
package com.example.watchstore;
public class Watch {
  private Long id;
  private String name;
  private String brand;
  private String description;
  private double price;
  private String imageUrl;
  public Watch() {
  }
  public Watch(Long id, String name, String brand, String description, double price, String imageUrl) {
    this.id = id;
    this.name = name;
    this.brand = brand;
    this.description = description;
    this.price = price;
    this.imageUrl = imageUrl;
  }
  // Getters and Setters
```

```
public Long getId() {
   return id;
}
public void setId(Long id) {
   this.id = id;
}
public String getName() {
   return name;
}
public void setName(String name) {
   this.name = name;
}
public String getBrand() {
   return brand;
}
public void setBrand(String brand) {
   this.brand = brand;
}
public String getDescription() {
   return description;
}
public void setDescription(String description) {
   this.description = description;
}
public double getPrice() {
   return price;
}
```

```
    public void setPrice(double price) {
        this.price = price;
    }
    public String getImageUrl() {
        return imageUrl;
    }
    public void setImageUrl(String imageUrl) {
        this.imageUrl = imageUrl;
    }
}
```

**4. (WatchController.java):**

```
package com.example.watchstore;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
@Controller
public class WatchController {
    private final WatchService watchService;
    public WatchController(WatchService watchService) {
        this.watchService = watchService;
    }
    @GetMapping("/")
    public String showWatches(Model model) {
        model.addAttribute("watches", watchService.getAllWatches());
        return "watches";
    }
}
```

**5. (WatchService.java):**

```
package com.example.watchstore;

import org.springframework.stereotype.Service;

import java.util.Arrays;

import java.util.List;

@Service

public class WatchService {

    public List<Watch> getAllWatches() {

        return Arrays.asList(

            new Watch(1L, "Chronograph", "Rolex", "Luxury chronograph watch with automatic movement", 8999.99, "/images/watch1.jpg"),

            new Watch(2L, "Diver Pro", "Omega", "Professional diving watch with 300m water resistance", 5999.99, "/images/watch2.jpg"),

            new Watch(3L, "Classic Heritage", "Tissot", "Vintage inspired watch with leather strap", 1299.99, "/images/watch3.jpg"),

            new Watch(4L, "Smart Watch X", "Apple", "Latest smartwatch with health monitoring", 499.99, "/images/watch4.jpg")

        );

    }

}
```

**6. (WatchstoreApplication.java):**

```
package com.example.watchstore;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class WatchstoreApplication {

    public static void main(String[] args) {

        SpringApplication.run(WatchstoreApplication.class, args);

    }
}
```

**Then run: (Start Debugging)**

**Open your browser: (http://localhost:8080)**