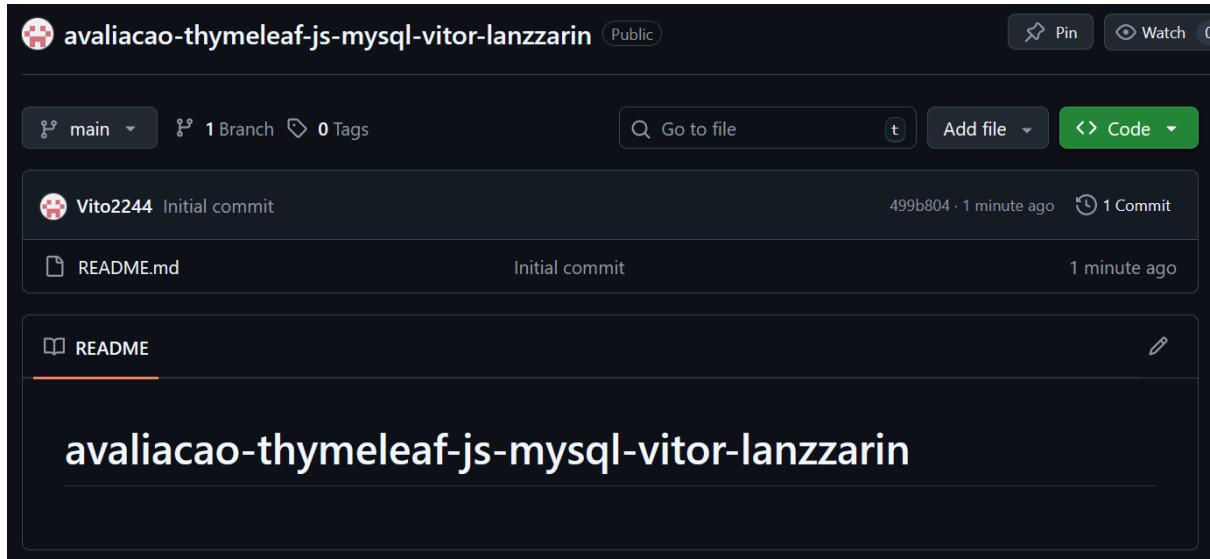
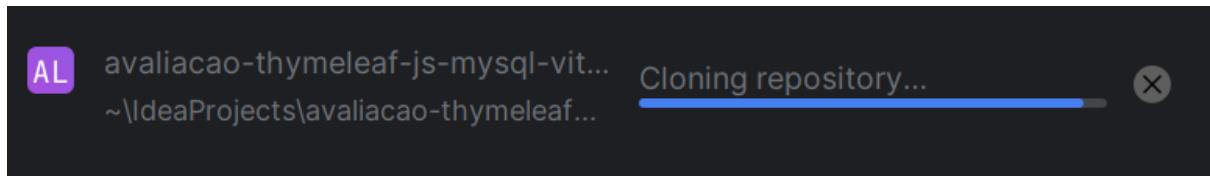


1- Criar o repositório:



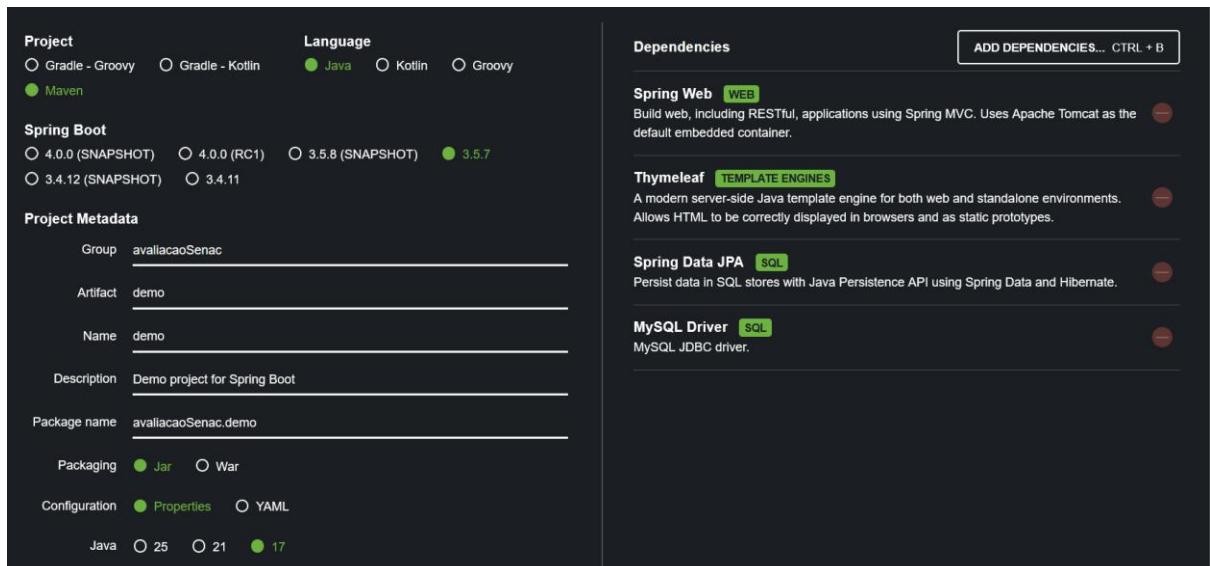
The screenshot shows a GitHub repository page. At the top, it says 'avaliacao-thymeleaf-js-mysql-vitor-lanzzarin' (Public). Below that, there are buttons for 'Pin' and 'Watch'. The repository has 1 branch and 0 tags. A search bar and a 'Go to file' button are also present. The main content area shows a commit from 'Vito2244' titled 'Initial commit' made 1 minute ago. The commit message is 'Initial commit'. Below the commit, there is a 'README' file with the same content: 'avaliacao-thymeleaf-js-mysql-vitor-lanzzarin'.

2- Clonar o repositório:



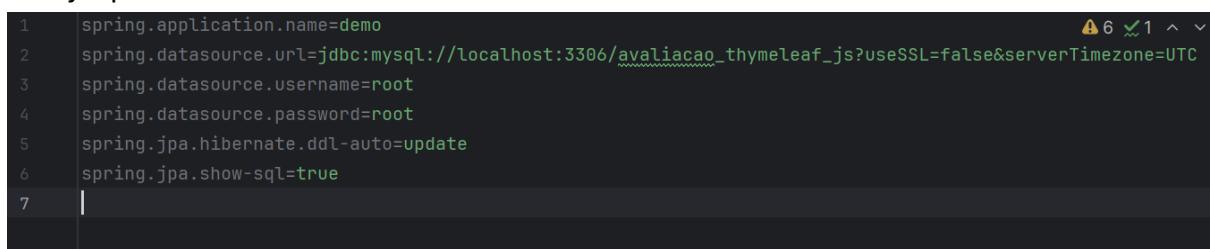
The screenshot shows an IDE interface with a progress bar indicating the 'Cloning repository...' process. The repository path is shown as '~\IdeaProjects\avaliacao-thymeleaf...'. The progress bar is nearly complete.

3-Spring Boot:



The screenshot shows the Spring Initializr interface. On the left, under 'Project', 'Maven' is selected. Under 'Language', 'Java' is selected. Under 'Spring Boot', '3.5.7' is selected. Under 'Project Metadata', the group is 'avaliacaoSenac', artifact is 'demo', name is 'demo', description is 'Demo project for Spring Boot', package name is 'avaliacaoSenac.demo', packaging is 'Jar', configuration is 'Properties', and Java version is '21'. On the right, under 'Dependencies', 'Spring Web' (WEB), 'Thymeleaf' (TEMPLATE ENGINES), 'Spring Data JPA' (SQL), and 'MySQL Driver' (SQL) are listed. There is a 'ADD DEPENDENCIES...' button at the top right.

4- MySql:



```
1 spring.application.name=demo
2 spring.datasource.url=jdbc:mysql://localhost:3306/avaliacao_thymeleaf_js?useSSL=false&serverTimezone=UTC
3 spring.datasource.username=root
4 spring.datasource.password=root
5 spring.jpa.hibernate.ddl-auto=update
6 spring.jpa.show-sql=true
7 |
```

The screenshot shows a code editor with a snippet of a configuration file. The code includes properties for the Spring application name ('demo'), database connection ('spring.datasource.url'), database credentials ('username' and 'password'), and Hibernate settings ('spring.jpa.hibernate.ddl-auto' and 'spring.jpa.show-sql'). The code is numbered from 1 to 7.

5- Entidade pessoa:

```
@Entity 5 usages
public class Pessoa {
    @Id 2 usages
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String nome; 2 usages
    private String email; 2 usages

    public Long getId() { no usages
        return id;
    }

    public void setId(Long id) { no usages
        this.id = id;
    }

    public String getNome() { no usages
        return nome;
    }

    public void setNome(String nome) { 1 usage
        this.nome = nome;
    }

    public String getEmail() { no usages
        return email;
    }
```

6- Repositorio e controlador:

```
1 package avaliacaoSenac.demo.repository;
2
3 import avaliacaoSenac.demo.model.Pessoa;
4 import org.springframework.data.jpa.repository.JpaRepository;
5
6 public interface PessoaRepository extends JpaRepository<Pessoa, Long> { 2 usages
7 }
8
9
10
11
12 @Controller no usages
13 public class PessoaController {
14     @Autowired 2 usages
15     private PessoaRepository pessoaRepository;
16
17     @GetMapping("/") no usages
18     @
19     public String index(Model model) {
20         model.addAttribute( attributeName: "mensagem", attributeValue: "Cadastro de Pessoas");
21         model.addAttribute( attributeName: "pessoas", pessoaRepository.findAll());
22         return "index";
23     }
24
25     @PostMapping("/salvar") no usages
26     public String salvar(@RequestParam String nome, @RequestParam String email) {
27         Pessoa p = new Pessoa();
28         p.setNome(nome);
29         p.setEmail(email);
30         pessoaRepository.save(p);
31         return "redirect:/";
32     }
33
34     @GetMapping("/sobre") no usages
35     public String sobre() {
36         return "sobre";
37     }
38 }
```

7- Thymeleaf:

```
2   <html lang="en" xmlns:th="http://www.thymeleaf.org">
8   <body>
9     <h1 th:text="${mensagem}"></h1>
10
11    <form action="/salvar" method="post">
12      <input type="text" name="nome" id="nome" placeholder="Nome">
13      <input type="email" name="email" id="email" placeholder="Email">
14      <button type="submit" onclick="salvarLocal()">Salvar</button>
15    </form>
16
17    <h2>Pessoas Cadastradas</h2>
18    <ul>
19      <li th:each="p : ${pessoas}" th:text="${p.nome + ' - ' + p.email}"></li>
20    </ul>
21
22    <hr>
23
24    <h3>Ultimo nome digitado (LocalStorage):</h3>
25    <p id="ultimoNome"></p>
26
27    <h3>Controlador de visitas (SessionStorage):</h3>
28    <p id="visitas"></p>
29
30    <a href="/sobre">Ir para Sobre</a>
31
32  </body>
33 </html>
```

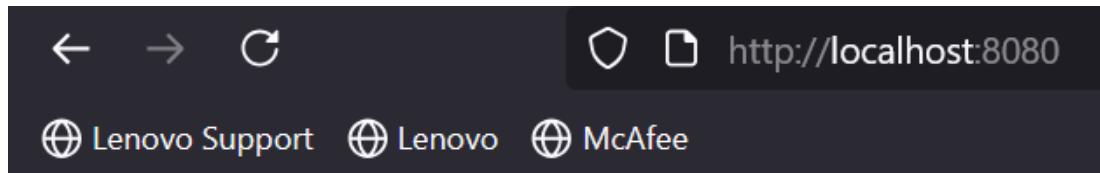


```
1  <!DOCTYPE html>
2  <html xmlns:th="http://www.thymeleaf.org">
3    <head>
4      <meta charset="UTF-8">
5      <title>Sobre</title>
6    </head>
7    <body>
8
9      <h1>Sobre o aluno</h1>
10     <p>Nome: Seu nome aqui</p>
11     <p>Descrição: Estudante de desenvolvimento web java com interesse em tecnologias modernas.</p>
12     <a href="/">Voltar</a>
13
14   </body>
15 </html>
```

8- JavaScript:

```
1  ↘ function salvarLocal(){
2      const nome = document.getElementById("nome").value
3      if (nome) {
4          localStorage.setItem("UltimoNome", nome);
5      }
6  }
7
8  ↘ window.onload = function(){
9      const ultimoNome = localStorage.getItem("ultimoNome");
10     if (ultimoNome) {
11         document.getElementById("ultimoNome").innerText = ultimoNome;
12     }
13
14     let visitas = sessionStorage.getItem("visitas");
15     visitas = visitas ? parseInt(visitas) + 1 : 1;
16     sessionStorage.setItem("visitas", visitas);
17     document.getElementById("visitas").innerText = visitas + " visita(s) nesta sessão.";
18 };
```

9- Teste:



Cadastro de Pessoas

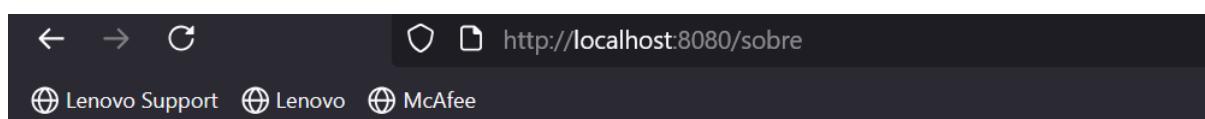
Pessoas Cadastradas

- Vitor - vitor@email.com
 - -
 - ryan - ryan@email
 - ryan - ryan123@email.com
-

Ultimo nome digitado (LocalStorage):

Controlador de visitas (SessionStorage):

[Ir para Sobre](#)



Sobre o aluno

Nome: Vitor

Descrição: Estudante de desenvolvimento web java com interesse em tecnologias modernas.

[Voltar](#)

10- GitHub:

The screenshot shows a GitHub repository page for a public repository named "avaliacao-thymeleaf-js-mysql-vitor-lanzarin". The repository has 1 branch and 0 tags. The main branch is selected. There are two commits listed:

- Vito2244** Add MySQL / JPA / Ent. Pessoa / Thymeleaf e JS · bf1adad · 8 minutes ago · 2 Commits
- demo** Add MySQL / JPA / Ent. Pessoa / Thymeleaf e JS · 8 minutes ago
- README.md** Initial commit · 2 hours ago