

Laboratório Prático: Configurando o Ambiente

Tempo estimado necessário: 15 minutos

O que você vai aprender

Neste laboratório, você configurará um repositório online para o projeto e aprenderá a criar pastas e arquivos no **Ambiente Skills Network**. Você também entenderá como visualizar a saída do seu código.

Objetivos de aprendizado

Após concluir este laboratório, você será capaz de:

- Criar um repositório online.
- Criar pastas e arquivos no **Ambiente Skills Network**.
- Executar o código para ver a saída.
- Realizar operações com Git.

Pré-requisitos

- Você deve ter concluído os cursos pré-requisitos, especialmente o curso **Introdução ao Git e GitHub**.
- Você deve ter uma conta no GitHub. Se você deseja criar uma conta no GitHub, clique [aqui](#) para os passos detalhados.

Passo 1: Crie um repositório online e crie um token de acesso pessoal

1. Crie um repositório público em branco no GitHub em sua conta, sem criar nenhum arquivo README.md para isso. Certifique-se de definir seu repositório como público e nomeá-lo adequadamente.
2. Você pode encontrar mais informações clicando neste [Create Github repository](#).
3. **Token de Acesso Pessoal** agora é uma parte obrigatória para enviar dados para repositórios do GitHub, para garantir que sua autenticação esteja relacionada à sua conta do GitHub.
4. Para criar um **Token de Acesso Pessoal**, vá para sua conta do GitHub e clique no ícone do seu perfil localizado no canto superior direito. Em seguida, clique em **Configurações**.
5. Em seguida, selecione **Configurações do desenvolvedor**. Esta opção geralmente está disponível na parte inferior da janela.
6. Navegue até **Tokens (clássico)** em **Tokens de acesso pessoal**.
7. Para gerar um token de acesso, clique em **Gerar um token de acesso pessoal**.
8. Na página **Gerar token**, preencha os detalhes necessários e marque a caixa de seleção **repo** para habilitar o acesso para comandos git.

9. Em seguida, clique em **Gerar token**.

10. Seu token de acesso pessoal será gerado. O token é válido por apenas **30 dias**. Você precisará gerar um novo token assim que o token atual expirar.

LEMBRE-SE: Certifique-se de copiar seu token de acesso pessoal agora. Você não poderá vê-lo novamente! Caso tenha esquecido de salvá-lo ou o tenha perdido, exclua o token já criado e gere um novo.

Nota: Este repositório será seu repositório principal para todos os laboratórios, exceto o laboratório do projeto de prática e o laboratório do projeto final.

Passo 2: Criar arquivos no Ambiente Skills Network

1. No painel da janela à direita, clique em Explorer, como mostrado na captura de tela abaixo (consulte o número 1). Em seguida, clique na pasta do projeto e depois clique no ícone destacado em vermelho (consulte o número 2) na captura de tela. Digite o nome da pasta como **SampleFolder**. Esta ação criará uma pasta para você. Em seguida, selecione a pasta **SampleFolder** mostrada no número 3, clique com o botão direito e selecione **New File**. Digite o nome do arquivo como **sample_folder.html** e clique em OK. Isso criará o arquivo HTML dentro da **SampleFolder**.

Nota: Não clique na pasta **.theia** antes de criar uma nova pasta. Certifique-se de estar no diretório raiz do projeto.

2. Crie a estrutura básica do template no arquivo HTML **sample_folder.html** copiando o conteúdo fornecido e colando no arquivo **sample_folder.html**.

```
<!DOCTYPE html>
<html>
<head>
  <title>Sample Folder</title>
</head>
<body>
  <h1>Sample Folder</h1>
  <button>SampleFolder</button>
</body>
</html>
```

Nota: Após colar o código, salve seu arquivo.

Passo 3: Execute comandos Git no terminal

1. Agora, clique na aba "Terminal" no canto superior direito da janela e, em seguida, clique em "Novo Terminal", conforme mostrado na captura de tela fornecida.

2. Em seguida, dentro do caminho do terminal, inicialize-o para criar um repositório Git para executar todos os comandos Git neste terminal. Use o comando fornecido para esse fim no terminal e pressione **Enter**.

```
git init
```

3. Depois, você precisa executar este comando no terminal para realizar os comandos necessários dentro do **Ambiente Skills Network** e pressione **Enter**.

```
git config --global --add safe.directory /home/project
```

Nota: O comando acima ajudará você a trabalhar dentro do ambiente da pasta do projeto com comandos git.

4. Então,

- o defina o `git config --global` para o endereço de e-mail executando o comando dado no terminal, fornecendo o endereço de e-mail da sua conta GitHub entre aspas duplas em vez de you@example.com e pressione **Enter**.

```
git config --global user.email "you@example.com"
```

- o Novamente, defina o `git config --global` para o nome de usuário executando o comando dado no terminal, fornecendo o nome de usuário da sua conta GitHub entre aspas duplas em vez de **Your Name** e pressione **Enter**.

```
git config --global user.name "Your Name"
```

5. Em seguida, execute `git add` e `git commit` para salvar as alterações no repositório GitHub, executando os seguintes comandos um após o outro:

```
git add --a
```

```
git commit -m "initial commit"
```

6. Execute os comandos `git push` para enviar os arquivos para o seu repositório GitHub.

- Adicione a URL do seu repositório GitHub na variável `origin2` escrevendo o comando dado no terminal.
- Também substitua `<git-repo-url>` pela URL do seu repositório GitHub, como `git remote add origin2 https://github.com/youraccountname/yourrepositoryname`, conforme mostrado na captura de tela fornecida, e clique em **Enter**.

```
git remote add origin2 <git-repo-url>
```

- Em seguida, execute o comando dado no terminal para enviar o conteúdo do seu arquivo para o repositório GitHub e clique em **Enter**.

```
git push origin2
```

7. Ao enviar os arquivos para o GitHub usando o comando `git push`, ele solicitará que você insira o nome de usuário da sua conta GitHub no terminal. Digite seu nome de usuário e pressione enter. Em seguida, ele também solicitará sua senha; aqui você precisará colar o **Token de Acesso Pessoal** que você gerou na etapa 1.

Nota: Ao colar seu **Token de Acesso Pessoal** no terminal, ele não aparecerá por motivos de segurança, mas já está lá. Basta pressionar enter, e ele enviará seus arquivos e pastas para o repositório GitHub.

8. Ele enviará todos os arquivos diretamente para o seu repositório GitHub.

Nota: Você também pode consultar [git commands](#) para mais instruções detalhadas.

Passo 4: Necessidade de realizar comandos Git

É crucial seguir alguns passos essenciais para garantir o gerenciamento adequado e a persistência dos seus dados em um repositório do GitHub:

- **Atualizações regulares:** Sempre que você fizer alterações ou adicionar novos componentes ao seu projeto, é essencial adicionar, commitar e enviar as atualizações para o seu repositório do GitHub. Esse processo garante que seu trabalho mais recente esteja armazenado com segurança e acessível aos colaboradores.
- **Persistência da sessão:** Durante uma sessão ativa, seus dados permanecem acessíveis. No entanto, é importante notar que, se sua sessão expirar ou você sair, precisará clonar o repositório novamente para retomar o trabalho.

Ao seguir essas diretrizes, você pode manter um repositório do GitHub bem organizado e eficiente, garantindo que seu trabalho esteja armazenado com segurança e facilmente acessível a você e seus colaboradores.

Passo 5: Verifique a saída

1. Para ver como sua página HTML será exibida no navegador, use a extensão Live Server embutida. Selecione o arquivo **sample_folder.html** na pasta **SampleFolder** e clique com o botão direito nesse arquivo e escolha ‘Abrir com Live Server’.
2. Uma notificação aparecerá no canto inferior direito, indicando que o servidor foi iniciado na porta 5500.
3. Em seguida, clique no botão Skills Network à esquerda (consulte o número 1). Essa ação abrirá a “Caixa de Ferramentas da Skills Network.” Em seguida, selecione “Iniciar Aplicação” (consulte o número 2). Uma vez lá, insira o número da porta **5500** em “Porta da Aplicação” (consulte o número 3) e clique neste botão .
4. Isso abrirá seu navegador padrão, onde você verá o nome da pasta **SampleFolder**.
3. Clique no nome dessa pasta. Isso mostrará os nomes dos arquivos disponíveis dentro dessa pasta.
4. Em seguida, clique no arquivo **sample_folder.html**, conforme mostrado abaixo.
5. A página inicial será aberta, permitindo que você veja a saída.

Nota: Lembre-se de salvar seus arquivos regularmente. Se você editar seu código, atualize seu navegador, que roda pela porta 5500. Dessa forma, não há necessidade de iniciar a aplicação novamente. Também, envie todas as últimas alterações de código no repositório do GitHub realizando os comandos git add, git commit e git push.

© IBM Corporation. Todos os direitos reservados.