

← → ↻

★

untitled_page × +

DSD - RMI -
BIBLIOTECA
MARIA GIOVANNA
E MATHEUS SILVA

<



...

>

×

😊

Hello
Nice to meet you



BIBLIOTECA RMI

- **Objetivo:** Implementar uma biblioteca virtual utilizando Pyro4 para comunicação RMI.
- **Componentes:**
 - Servidor (server.py)
 - Cliente (client.py)
 - Nameserver (start_ns.py)



CONCEITO DE RMI

REMOTE METHOD INVOCATION

- **RMI:** Técnica que permite a execução de métodos em objetos localizados em diferentes máquinas
- **Benefícios:** Facilita a criação de aplicações distribuídas, abstraindo a complexidade da comunicação entre processos.

ESTRUTURA DO PROJETO

- **Servidor (server.py):**
 - Gerencia livros, usuários e empréstimos.
- **Cliente (client.py):**
 - Interage com o usuário e realiza operações na biblioteca.



CONFIGURAÇÃO DO SERVIDOR

```
import Pyro4
from datetime import datetime
```

- **Definição da Classe Biblioteca:**
 - Métodos para gerenciamento de usuários, livros e sessões.
- **Execução do Servidor:**
 - Configura daemon e Nameserver do Pyro4.
 - Registra a classe Biblioteca.



CÓDIGO DO SERVIDOR



DEMONSTRAÇÃO:

```
@Pyro4.expose
@Pyro4.behavior(instance_mode="single")
class Biblioteca:
    def __init__(self):
        self.livros = ['O Senhor dos Anéis', 'Harry Potter']
        self.livrosEmprestados = {}
        self.usuarios = {}
        self.sesoes = {}
```

Métodos para gerenciar biblioteca...



* EXECUÇÃO DO SERVIDOR

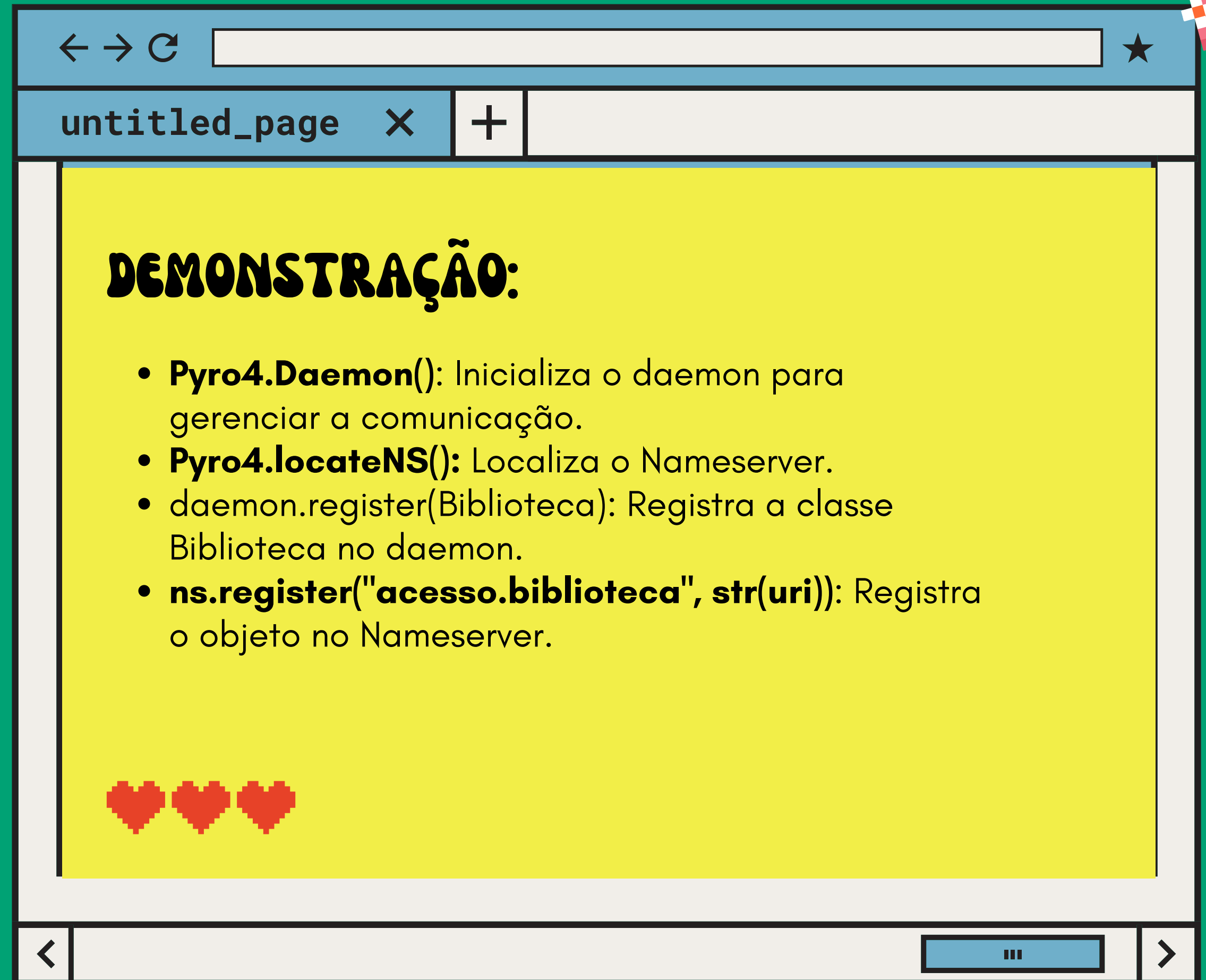
Chat

```
def main():  
    daemon = Pyro4.Daemon()  
    ns = Pyro4.locateNS()  
    uri = daemon.register(Biblioteca)  
    ns.register("acesso.biblioteca", str(uri))  
    print("Servidor da biblioteca está em  
execução.")  
    daemon.requestLoop()  
if __name__ == "__main__":  
    main()
```

Send



USO DO PYRO NO SERVIDOR





CONFIGURAÇÕES



Chat

NAMESERVER

```
import Pyro4.naming  
Pyro4.naming.startNSloop()
```

- Função: Gerencia a descoberta de serviços na rede.
- Configuração do Cliente

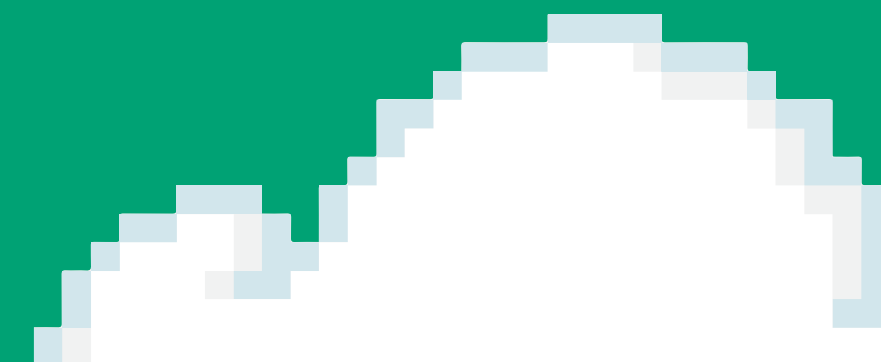
CLIENTE

```
import Pyro4
```

Menu Interativo:

- Menus para login, registro e operações na biblioteca.

Send



CONFIGURAÇÕES

Chat

CÓDIGO DO CLIENTE (EXEMPLO)

```
def main():  
    biblioteca =  
    Pyro4.Proxy("PYRONAME:acesso.biblioteca")  
    usuario_logado = None  
    # Loop para menu e operações...  
if __name__ == "__main__":  
    main()
```

FLUXO DE OPERAÇÕES DO CLIENTE

- Registro e Autenticação de Usuários.
- Gerenciamento de Livros:
 - Adicionar livros.
 - Emprestar e devolver livros.
 - Listar livros disponíveis e emprestados.

Send

