

• Persistência do estado do login

(usado para definir se o usuário pode estar logado em qualquer aba do navegador ou apenas em uma aba apenas)

Códigos iniciais vistos nos exemplos acima:

- Conectar o javascript com banco de dados do projeto e adicionar o script para utilizar a autenticação do projeto
- criar uma variável onde será atribuído o firebase e um método que retorna uma referência para o banco de dados [o método é o `.firestore()`]

```
let bancoDados = firebase.firestore();
```

- chamar um método de autenticação do firebase

```
let autenticacao = firebase.auth();
```

Para controlar o acesso do usuário definido regras de login. Se ele tem que se Logar todo tempo ou entra em uma única aba do navegador:

```
autenticacao.setPersistence (firebase.auth.Auth.Persistence.Nomes)
```

Colocar esse código dentro de uma função que Loga usuários ao sistema

Nomes:

- LOCAL (fica logado em qualquer aba do navegador)
- SESSION (fica logado apenas em uma aba do navegador)
- NONE (faz o usuário ficar deslogado quando atualizar uma ação no sistema. Usado quando temos sistemas que precise de muita segurança onde cada ação que o usuário realizar ele deve se Logar novamente para cada ação que ele for realizar)

Exemplo de um Código para Logar usuário e definir o tipo de controle do Login

- Conectar o javascript com banco de dados do projeto e adicionar o script para utilizar a autenticação do projeto
- criar uma variável onde será atribuído o firebase e um método que retorna uma referência para o banco de dados [o método é o `.firestore()`]

```
let bancoDados = firebase.firestore();
```

- chamar um método de autenticação do firebase

```
let autenticacao = firebase.auth();
```

usar duas variáveis email e senha de um usuário que já existe, para ter acesso ao sistema sem erro

```
let userEmail = "emailteste@gmail.com";
let userPassword = "senha12345";
```

```
autenticacao.setPersistence (firebase.auth.Auth.Persistence.Nomes).then (()=>{
```

chamar um método do firebase para logar usuários (colocar apenas o email/senha do usuário a ser logado)

```
autenticacao.signInWithEmailAndPassword(userEmail, userPassword)
    .then(loggedUser => {
```

```
        console.log(autenticacao.currentUser);
```

```
    }).catch(error => {
        caso tenha algum erro dever ser mostrado o erro
        console.log(error);
    })
```

```
}).catch(error => {
    caso tenha algum erro dever ser mostrado o erro
    console.log(error);
})
```