



# Estilizando com Estilo

Aula 05 - Hard Skills



Kam Idris (Unsplash)



Todos os direitos reservados  
©2023 Resilia Educapio

Códigos  
do Amanhã  
2023 EBANX RESILIA



## Objetivos de aprendizagem

- ⇒ Compreender o conceito de componentes base e variantes em Styled Components.
- ⇒ Aprender a criar um componente base com estilos padrão.
- ⇒ Entender como aplicar essas variantes em componentes JSX.



Todos os direitos reservados  
©2023 Resilia Educação



### Nesta aula, veremos:

- ⇒ Styled-components
- ⇒ Global Style
- ⇒ ThemeProvider

⇒ Variantes

---

### Por que aprender isso?

- ⇒ Styled Components oferece controle visual, consistência e flexibilidade na estilização de componentes React. Você pode criar estilos modulares e reutilizáveis, tornando o código mais organizado e facilitando a colaboração com a equipe de design.
- ⇒ Assim como o CSS, o Styled Components também promove a organização do código. No entanto, ele vai além, permitindo que você mantenha estilos em linha com os componentes que eles estilizam. Isso simplifica a manutenção, tornando as alterações nos estilos mais previsíveis e menos propensas a quebras em outras partes do aplicativo.
- ⇒ Styled Components permite a criação de temas personalizados, como temas claros e escuros. Isso é importante para a experiência do usuário, e você pode alternar entre temas facilmente, melhorando a compatibilidade e a portabilidade.



# CSS



Todos os direitos reservados  
©2023 Realiz Educação

# CSS

---

CSS (cascading style sheets), folha de estilo em cascata é o que usamos para dar vida às páginas web, alterando fontes, cores, layouts e muito mais.

Existem aproximadamente 10 formas de se escrever CSS.

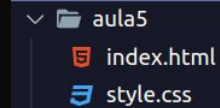


Todos os direitos reservados  
©2023 Realiza Educação

## CSS Puro(CSS Externo)

A forma mais comum é a que já conhecemos, chamada de **CSS Puro (CSS Externo)**, que é quando criamos um arquivo com a extensão `.css` e o importamos no arquivo `html` que vamos estilizar

```
1  body{  
2      width: 100%;  
3      height: 100vh;  
4      display: flex;  
5      align-items: center;  
6      justify-content: center;  
7  }
```



```
aula5  
├── index.html  
└── style.css
```



Todos os direitos reservados  
©2023 Realiza Educação

## CSS em Linha

O CSS em linha, nos permite aplicar estilos diretamente a elementos HTML usando o atributo `style`. Isso é útil para estilos específicos a serem aplicados a elementos individuais.

```
<body>
  <h1 style="color: blue;">Título em Azul</h1>
  <p style="font-size: 16px; background-color: #eee;">
    Este é um parágrafo
  </p>
</body>
```



Todos os direitos reservados  
©2023 Resilia Educação

## CSS-in-JS

O CSS-in-JS é uma maneira de escrever estilos CSS diretamente no seu código JavaScript. É uma abordagem que combina a escrita de estilos com o código JavaScript, tornando a estilização mais eficiente e melhorando a manutenção do seu código.

```
const StyledButton = styled.button`
  background-color: #007bff;
  color: #fff;
  padding: 10px 20px;
  cursor: pointer;

  &:hover {
    background-color: #0056b3;
  }
`;
```



Todos os direitos reservados  
©2023 Resilia Educação



## Vantagens CSS-in-JS

- ⇒ Encapsulamento de Estilos
- ⇒ Estilização Dinâmica
- ⇒ Reutilização de Componentes e Estilos
- ⇒ Performance Otimizada

- ⇒ Facilidade de Migração
- ⇒ Conveniência em Projetos JavaScript/React
- ⇒ Documentação Autodescritiva
- ⇒ Tematização Simples



Todos os direitos reservados  
©2023 Resilia Educação

### NOTA PARA O ESTUDANTE

Usar CSS-in-JS em vez de CSS puro oferece muitas vantagens, aqui estão as principais:

- **Encapsulamento de Estilos:** Com CSS-in-JS, os estilos são encapsulados em componentes individuais. Isso evita conflitos de estilos entre componentes e garante que os estilos sejam aplicados apenas onde são necessários.
- **Estilização Dinâmica:** É fácil estilizar componentes com base em props ou estado. Isso permite criar interfaces altamente dinâmicas e reativas.
- **Reutilização de Componentes e Estilos:** Você pode criar componentes estilizados reutilizáveis e compartilhá-los em todo o seu aplicativo.
- **Performance Otimizada:** Muitas bibliotecas CSS-in-JS geram estilos otimizados e apenas injetam o CSS necessário no DOM, o que pode melhorar o desempenho do aplicativo.
- **Facilidade de Migração:** Se você já possui um projeto com CSS puro, pode migrar gradualmente para CSS-in-JS, pois as duas abordagens podem coexistir.
- **Conveniência em Projetos JavaScript/React:** CSS-in-JS se integra bem com projetos JavaScript, como React, onde você já está escrevendo código JavaScript. Isso simplifica a criação de componentes visualmente coesos.
- **Documentação Autodescritiva:** Os estilos são parte integrante do código do componente, tornando a documentação mais autodescritiva, facilitando a colaboração entre desenvolvedores.

- **Tematização Simples:** CSS-in-JS facilita a criação de temas para seu aplicativo, permitindo que você altere a aparência global do aplicativo com facilidade.

A close-up photograph of a person's hands, focusing on the fingers and nails. The nails are painted a vibrant pink color. One finger, likely the ring finger, is adorned with a ring featuring a large, dark, oval-shaped stone set in a metal band. The background is softly blurred, showing more of the hands and possibly a keyboard, suggesting a professional or creative setting.

# STYLED-COMPONENTS



Todos os direitos reservados  
©2023 Realiz Educação

## O que é

Styled Components é uma biblioteca popular para estilizar componentes em React. Ela permite que você escreva estilos CSS diretamente no código dos componentes, mantendo-os isolados.



Todos os direitos reservados  
©2023 Realiza Educação

## Instalação

Para instalar o Styled Components em um projeto JavaScript ou React, você deve usar o seguinte código:

```
npm install styled-components
```

Para usufruir ao máximo da lib recomendamos que também instale a extensão do VSCode



### **NOTA PARA O ESTUDANTE**

Link da documentação: [styled-components](https://styled-components.com)

Link da extensão: [vscode-styled-components - Visual Studio Marketplace](https://marketplace.visualstudio.com/items?itemName=styled-components.vsc-styled-components)

Comando para instalar: `npm install styled-components`

## Criando o arquivo

O primeiro passo é criar um arquivo `nomeDoComponente.styles.js`

**Exemplo:** Vamos criar um componente **Button.jsx**

O seu arquivo de estilo deve se chamar **button.styles.js**



Todos os direitos reservados  
©2023 Resilia Educação

### **NOTA PARA O ESTUDANTE**

Usamos a extensão de arquivo **.styles.js** por dois motivos:

- 1 - O **.js** é porque estamos trabalhando com CSS-in-JS, ou seja, vamos escrever o CSS em um arquivo JS
- 2 - O **.styles** é para diferenciar o nome do arquivo do componente do seu arquivo de estilo

## Importando o styled-components

Dentro do arquivo `button.styles.js` precisamos importar o styled-components.

Para isso usamos a sintaxe

```
import styled from 'styled-components'
```



Todos os direitos reservados  
©2023 Resilia Educação

### NOTA PARA O ESTUDANTE

**Importação:** Em cada arquivo `styles.js` só é necessário importar o **styled** apenas uma vez

## Criando o componente de estilo

Com o componente styled importado, podemos criar os nossos componentes

Sintaxe padrão

```
export const NomeDoComponente = styled.nomeDaTag`  
  código css  
`
```

Sintaxe aplicada ao componente

```
export const StylesButton = styled.button`  
  código css  
`
```



Todos os direitos reservados  
©2023 Resilia Educação

### NOTA PARA O ESTUDANTE

**export const NomeDoComponente:** Cria uma constante chamada **NomeDoComponente** que será exportada para que você possa usá-la em outros arquivos do seu projeto. O nome que você escolher para **NomeDoComponente** é o nome do componente que você está criando. No exemplo do slide, criamos a constante **StyleButton**. Ela é, portanto, o nosso componente de estilo.

**styled.nomeDaTag:** O styled é um objeto exportado pela biblioteca Styled Components. A parte **.nomeDaTag** indica que você está criando um componente estilizado com base em uma tag HTML específica. No exemplo do slide, estamos criando um componente estilizado com base em um elemento `<button>`, e, por esta razão, usamos **styled.button**.

**`**: Isso é chamado de template literal do JavaScript (Famoso ostentação bigodinho). Permite que você insira código CSS diretamente dentro das crases (```). O código CSS inserido aqui será aplicado ao componente.



## Importando o componente

Com o componente de estilo pronto, agora precisamos importá-lo no nosso arquivo `.jsx`

Importando o componente `StylesButton`

```
import {StylesButton} from './button.styles.js'
```



Todos os direitos reservados  
©2023 Resilia Educação

## Atividade: Instalando e convertendo

### → O QUE É PRA FAZER?

- ⇒ Instalar a biblioteca Styled-components
- ⇒ Converter todos os arquivos `module.css` para `styles.js`



Todos os direitos reservados  
©2023 Resilia Educatio



## Global Style - O que é

---

O Global Style é uma forma de definir estilos que afetam toda sua aplicação, como cores e fontes, espaçamentos e até o `reset.css`. Isso garante uma aparência consistente em todas as páginas, economizando tempo e mantendo a identidade visual do seu projeto.



Todos os direitos reservados  
©2023 Resilia Educação

## Definindo o Global Styles

Para criar o global style o ideal é ter um arquivo chamado *GlobalStyles.js*.  
No arquivo nós importamos o *createGlobalStyles* da lib *styled-components*

```
import {createGlobalStyles} from 'styled-components'
```



Todos os direitos reservados  
©2023 Resilia Educação

## Definindo o Global Styles

Com o `createGlobalStyles` importado, podemos criar o componente com todo o css que será global na aplicação

```
export const GlobalStyle = createGlobalStyle`
  //Reset CC
  *{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
  }
  body{
    width: 100%;
    min-height: 100vh;
  }
`
```

 Todos os direitos reservados  
©2023 Resilia Educação

### NOTA PARA O ESTUDANTE

Usamos o global style para definir estilos que serão reutilizados na aplicação. Além disso, é também o melhor lugar para aplicar o `reset.css`. Como estamos trabalhando com um projeto grande, esse é o lugar para resetar algumas coisas que, por padrão, não existem no HTML. Veja alguns exemplos a seguir:

Exemplo 1 - Esse trecho define que toda aplicação terá 100% de largura, e a altura mínima será a altura da tela do usuário

```
body{
  width: 100%;
  min-height: 100vh;
}
```

Exemplo 2 - Aqui estamos declarando que essas tags, ao serem selecionadas, não terão o outline

```
button, input, textarea, select{
  outline: none;
```

```
}
```

As tags Button são tags de interação com o usuário. Por padrão do HTML, elas não vem com o cursor pointer. Por esta razão, já definimos isso aqui, para não precisar repetir esse código em outros lugares da aplicação.

```
button, a{  
  cursor: pointer;  
}
```

A tag **a** vem por padrão da cor azul e com um underline, já a tag **li** vem o o *list-style*(O pontinho antes do texto). Como na aplicação não aproveitamos esse estilo padrão, aqui nós devemos relatá-los.

```
a, li{  
  text-decoration: none;  
  list-style: none;  
  color: inherit;  
}
```

## Importando o Global Styles

Com o componente pronto, agora o que precisamos fazer é importá-lo. Para que funcione corretamente, o ideal é que fique sempre na raiz do projeto. A raiz do projeto nesse momento é o *main.js*

```
ReactDOM.createRoot(document.getElementById("root")).render(  
  <React.StrictMode>  
    <GlobalStyle />  
    <LandingPage/>  
  </React.StrictMode>  
);
```



Todos os direitos reservados  
©2023 Resilia Educação



# Theme Provider



Todos os direitos reservados  
© 2023 Realiz Educação

# Theme Provider

---

Simplifica o uso de diferentes temas de cores e fontes em uma aplicação. Ele cria variáveis para cores e estilos, tornando fácil aplicar e atualizar essas cores em todos os componentes e, consequentemente, simplifica a estilização e a personalização.



Todos os direitos reservados  
©2023 Resilia Educação

## Criando o tema

Para ter um tema, o primeiro passo é criar um.

Em um arquivo que idealmente é chamado de *theme.js* criamos o objeto do tema com as cores e os nomes das variáveis.

Como usaremos esse tema em outros arquivos é importante que ele seja exportado.

```
export const tema = {  
  blue900: "#263238",  
  white50: "#FFFFFF",  
  white200: "#E9F0F0",  
  red500: "#9A4A4D",  
  green500: "#34673E",  
};
```



Todos os direitos reservados  
©2023 Resilia Educação

## Importando o tema e o Theme provider

Com o tema criado, precisamos importar o ThemeProvider e o tema para a nossa aplicação.

Para que funcione o ideal é que fique sempre na raiz do projeto.  
A raiz do projeto nesse momento é o *main.js*

```
import {ThemeProvider} from 'styled-components'  
import {tema} from '../theme.js'
```



Todos os direitos reservados  
©2023 Resilia Educação

## Aplicando o Tema

Agora que temos o ThemeProvider e o nosso tema importados precisamos colocar na aplicação.

O ThemeProvider recebe uma props chamada **theme**, nesta propriedade nós passamos o tema que criamos.

Para que todos os locais da aplicação tenham acesso ao tema, é importante que o ThemeProvider esteja em torno de todos os componentes.

```
ReactDOM.createRoot(document.getElementById("root")).render(  
  <React.StrictMode>  
    <ThemeProvider theme={tema}>  
      <GlobalStyle />  
      <LandingPage />  
    </ThemeProvider>  
  </React.StrictMode>  
);
```



Todos os direitos reservados  
©2023 Resilia Educação

## Usando o tema

Com toda configuração pronta, agora podemos usufruir do nosso tema. Para isso vamos aplicar uma cor no componente Button criado anteriormente.

O ThemeProvider passa o tema como props para todos os componentes filhos. Dessa forma, para acessar o tema precisamos usar uma sintaxe js.

```
`  
background-color: ${(props) => props.tema.white50}  
`
```



Todos os direitos reservados  
©2023 Resilia Educação

## Atividade: GlobalStyle

### → O QUE É PRA FAZER?

- ⇒ Agora que já conhecemos o global style do styled-components, vamos criar o nosso, dentro da pasta **style**:



Todos os direitos reservados  
©2023 Resilia Educapto

## Atividade: GlobalStyle

### ⇒ COMO FAZER?

- Substitua o **GlobalStyle.css** por um arquivo **GlobalStyle.js**
- Crie o componente **GlobalStyle**
- Adicione o **reset.css** ao componente criado
- Importe o componente **GlobalStyle** no arquivo **main.js**
- Crie um arquivo **theme.js** exportando o objeto **theme**
- Adicione ao objeto, a paleta de cores do design system, respeitando os nomes
- Importe o objeto no arquivo **main.js** e passe-o como props para o **ThemeProvider**

### → FECHAMENTO

Compartilhar dúvidas e dificuldades na sala principal



Todos os direitos reservados  
©2023 Resilia Educatio



# Variantes

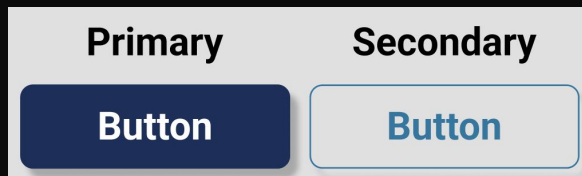


Todos os direitos reservados  
©2023 Realiz Educação

# Variantes

As variantes de um componente são estilos alternativos ou personalizações que você pode aplicar ao mesmo componente, com base em diferentes condições ou estados.

Essas variantes permitem que você reutilize um componente básico em várias situações, ajustando sua aparência de acordo com as necessidades do seu aplicativo.



## Componente base

O componente base é o ponto de partida para criar estilos compartilhados. É o componente que define os estilos comuns que outras variantes de componentes podem estender e personalizar.

```
const Button = styled.button`  
  border: 1px solid transparent;  
  border-radius: 8px;  
  width: fit-content;  
  text-align: center;  
  padding: 12px 24px;  
`;  
;
```

## Variantes

Para criar as variantes, vamos estender o componente base aplicando apenas os estilos únicos em cada componente variante.

```
export const PrimaryButton = styled(Button) `
background-color: ${(props) => props.tema.blue900};
color: ${(props) => props.tema.white50};
`;
```

```
export const SecondaryButton = styled(Button) `
background-color: transparent;
color: ${(props) => props.tema.blue500};
border: 1px solid ${(props) => props.tema.blue500};
`;
```

## Aplicando variantes

Com tudo criado, agora o que precisamos fazer é definir qual variante será retornada em nosso componente jsx de acordo com a variante criada

```
function Button({variant, label}) {  
  return (  
    <>  
    {variant == 'primary' && <PrimaryButton>{label}</PrimaryButton>}  
    {variant == 'secondary' && <SecondaryButton>{label}</SecondaryButton>}  
    </>  
  );  
}
```

## Escolhendo variantes

Agora , sempre que usarmos o componente é só passar a variante escolhida como props

```
<Button variant='primary' label='Botão primário' />  
<Button variant='secondary' label='Botão secundário' />
```

## Atividade: Criando componentes

### → O QUE É PRA FAZER?

Criar componentes no projeto react Finanças+

### → COMO FAZER?

- ⇒ Dentro da pasta componentes, crie uma pasta chamada **common**. Dentro dela, crie os componentes a seguir, passando as propriedades corretas e necessárias.
  - Logo (Com variação de cor e tamanho)
  - Button (Com o hover)
- ⇒ Dica: Para visualizar os componentes importe-os na página **LandingPage.jsx**
- ⇒ No componente **GlobalStyle** adicione a estilização correta para as tags de texto, baseado na tipografia apresentada no figma



# Dúvidas



Todos os direitos reservados  
©2023 Realiz Educação





# Estilizando com estilo

Aula 05 - Soft Skills



Kamilla Lins



Todos os direitos reservados  
©2023 Resilia Educapio

Códigos  
do Amanhã  
2023 EBANX RESILIA



## Objetivos de aprendizagem

- ⇒ Relacionar tipos de comportamentos com a prática da escuta ativa;
- ⇒ Conjecturar um diálogo entre empresa e cliente utilizando a escuta ativa.



Todos os direitos reservados  
©2023 Resilia Educação



### **Nesta aula, veremos:**

- ⇒ Dinâmica da Escuta ativa.

---

### **Por que aprender isso?**

- ⇒ A escuta ativa é importante porque melhora a comunicação, constrói relacionamentos mais sólidos, ajuda na resolução de problemas, promove o aprendizado e o desenvolvimento pessoal. Ao adotar essa habilidade, você se torna um comunicador mais eficaz e empático, capaz de estabelecer conexões significativas com os outros.



Todos os direitos reservados  
©2023 Realize Educação





O que é **escuta ativa**?

Qual a importância da escuta ativa aplicada  
à **relação com o cliente**?



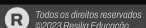
Todos os direitos reservados.  
© 2024 Brasil Educação

## Escuta ativa

Habilidade que pode ser adquirida com a prática e significa estar **totalmente concentrado no que está sendo dito**, em vez de apenas ouvir passivamente a mensagem de quem está falando.

Além de dar atenção total ao falante, é importante que o ouvinte ativo também consiga **mostrar que realmente está ouvindo**.

Na relação com o cliente é ouvi-lo sem pensar no que ele provavelmente está pensando, o que ele possivelmente quer resolver ou o que você pode oferecer para resolver o problema dele.



Todos os direitos reservados  
©2023 Realiza Educação

### **NOTA PARA O ESTUDANTE**

Escuta ativa é uma técnica de comunicação onde o ouvinte dedica total atenção ao falante, buscando compreender não apenas as palavras, mas também as emoções e o contexto por trás da mensagem. Nesse processo, o ouvinte demonstra interesse genuíno e empatia, permitindo que o falante se sinta valorizado e compreendido. Ela ainda desempenha um papel fundamental ao lidar com os clientes e é essencial por várias razões:

- **Compreensão das necessidades do cliente:** Através da escuta ativa, é possível entender melhor o que o cliente está procurando, suas expectativas e preocupações. Isso permite que você aceite soluções mais personalizadas e adaptadas às suas necessidades.
- **Construção de relacionamentos:** Ao demonstrar interesse genuíno pelo que o cliente tem a dizer, você cria um vínculo de confiança e respeito. Isso leva a relacionamentos mais sólidos e duradouros com os clientes.
- **Resolução eficiente de problemas:** Ao ouvir atentamente os problemas ou queixas do cliente, você pode identificar a causa raiz e encontrar soluções mais eficazes. Isso evita mal-entendidos e reduz a possibilidade de conflitos.

- **Evitar erros e retrabalho:** A escuta ativa ajuda a evitar mal-entendidos e interpretações equivocadas, o que reduz a ocorrência de erros e a necessidade de retrabalho.
- **Melhoria da experiência do cliente:** Clientes que se sentiram ouvidos e compreendidos tendem a ter uma experiência mais positiva. Isso pode levar à satisfação do cliente e ao boca a boca positivo, promovendo a confiança da empresa.
- **Identificação de oportunidades de venda:** Ao ouvir as necessidades do cliente, você pode identificar oportunidades de oferecer produtos ou serviços adicionais que podem agregar valor ao cliente.
- **Demonstração de profissionalismo:** A escuta ativa é uma habilidade valorizada no atendimento ao cliente, e ao praticá-la, você demonstra profissionalismo e respeito pela opinião do cliente.

A man with glasses and a beard, wearing a blue shirt, sits at a wooden desk with a laptop. He is looking towards a woman with long dark hair, who is also wearing a blue shirt and is seen from the side. They appear to be in a meeting or a collaborative work environment. The background is slightly blurred, showing a bookshelf and some office equipment.

# Momento de reflexão

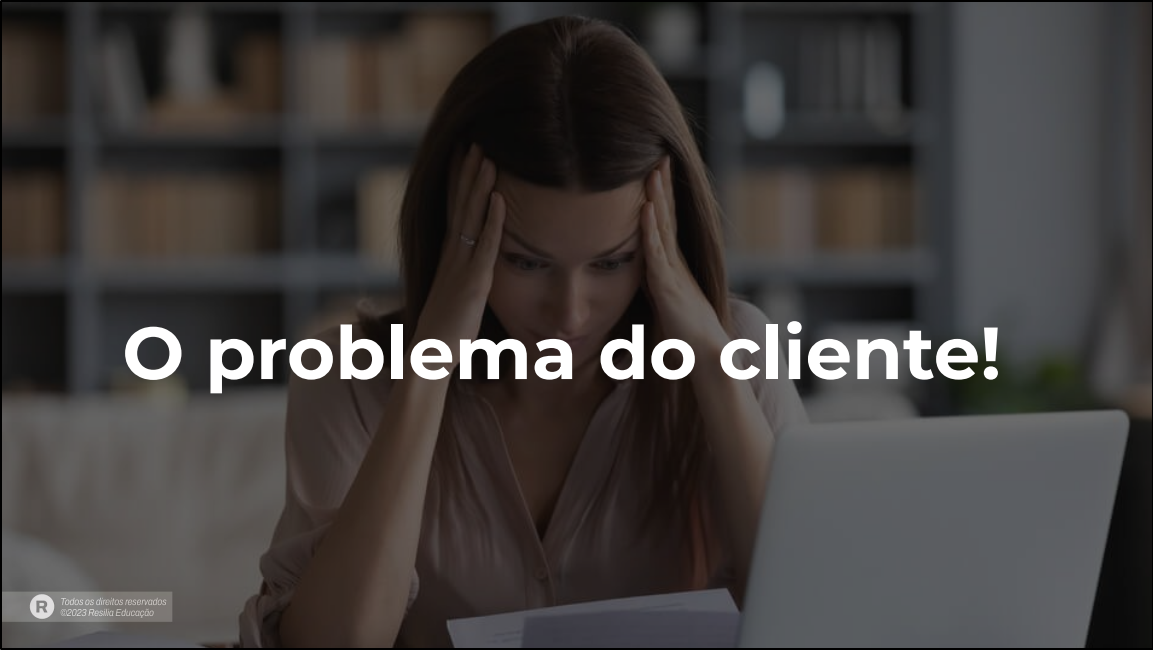
Quais **comportamentos** uma pessoa com **escuta ativa** demonstra?



## **Dicas para ter uma escuta ativa:**

1. Faça perguntas abertas, evite perguntas de sim ou não;
2. Resuma o que a pessoa falou;
3. Repita uma palavra ou frase;
4. Diga "Conte-me mais sobre isso";
5. Fale palavras de encorajamento.





**O problema do cliente!**



Todos os direitos reservados  
©2023 Realiza Educação

## **Mão na Massa: O problema do cliente!**

### **→ O QUE É PRA FAZER?**

Simular uma conversa com o cliente.

Uma pessoa do grupo fará o papel do cliente e receberá por chat as informações sobre a necessidade dele que precisa ser atendida.

O resto do grupo terá que trabalhar a **escuta ativa** para buscar a melhor solução para esse cliente.

Vocês devem escutar e fazer perguntas para entender melhor o caso.

## **Mão na Massa: O problema do cliente!**

### **→ COMO FAZER?**

Acessar o material que descreve o caso para a pessoa que será o cliente.

### **→ FECHAMENTO**

Compartilhar as ideias na sala principal



Todos os direitos reservados  
©2023 Resilia Educatio



## Para ir além

- ⇒ [Escuta ativa: o que é, como fazer, princípios e dicas.](#)
- ⇒ [Saiba o que é e como praticar a escuta ativa com estas 10 dicas.](#)



Todos os direitos reservados  
©2023 Resilia Educação





Até a próxima e  
**#confianoprocesso**



Todos os direitos reservados  
©2023 Resilia Educação

Códigos  
do Amanhã  
2023 EBANX RESILIA