* **TypeScript**: é um **superset de JavaScript**, ou seja, um conjunto de ferramentas e formas mais eficientes de escrever código JS, adicionando recursos que não estão presentes de maneira nativa na linguagem. Tem **tipagem estática** (os tipos das variáveis são definidos explicitamente no código) e **orientação a objeto**
* O TypeScript traz diversos **benefícios**, entre eles: **potencial de detecção de erros durante o desenvolvimento de projetos e a possibilidade de incluir a inteligência da IDE** que está sendo usada, refletindo num ambiente muito mais ágil e seguro durante o desenvolvimento do código

Conceitos Básicos

* **Tipagem estática**

Exemplo:

const nome: string = ‘Eduardo’;

const idade: number = 19;

const ativo: boolean = false;

* **Inferência de tipos**: quando definimos uma variável como string, o TS infere que essa variável sempre será uma string
* **Interfaces**: tipagem de um objeto, dizendo o que um objeto deve ter (obrigatório)

Exemplo:

**interface** IUsuario {

nome: string;

idade: number;

ativo: boolean;

}

const usuário: IUsuario = {

nome: ‘Eduardo’,

idade: 19,

ativo: true

}

* **Types**: muito semelhante a uma interface, com o mesmo papel

Exemplo:

**type** IUsuario {

nome: string;

idade: number;

ativo: boolean;

}

const usuário: IUsuario = {

nome: ‘Eduardo’,

idade: 19,

ativo: true

}

* **Enum**: usado para padronizar/centralizar algo, como por exemplo cargos de programação, ao invés de deixar em aberto para cada usuário se definir de forma diferente (programador, dev, software developer etc.)

Exemplo:

enum CARGO {

DESENVOLVEDOR = ‘desenvolvedor’

}

const usuário = {

nome: ‘Eduardo’,

cargo: CARGO.DESENVOLVEDOR

}