Documento Projeto Lista de Contatos

# Preparação do ambiente

No powershell;

Acessar o diretório do projeto;

 > 

Configurar o Create React App;



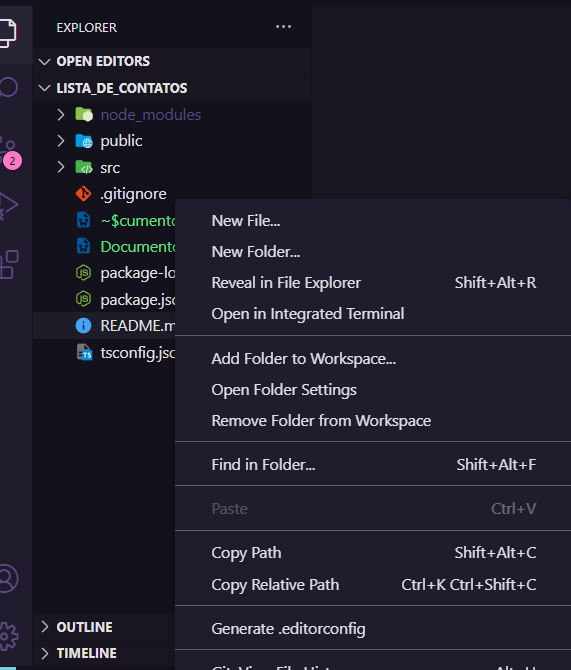
Criar pasta do projeto configurando template typescript;



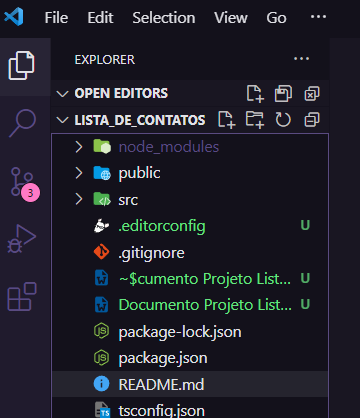
Abrir a pasta criada pelo vscode;

Clicar com direito na área de arquivos e selecionar Generate.editorconfig;

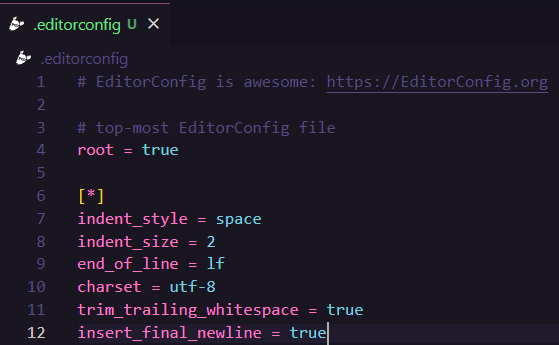
Obs: tem que estar com a extensão editorconfig do vscode instalada;



Clicar com o direito nesta área



Mudar para esta configuração;



Configurar o eslint;

No terminal vscode;



Se der erro:

se der erro:

npm cache clean --force

npm install @typescript-eslint/eslint-plugin@5.5.

npm install –force

Continuando…













se der erro:

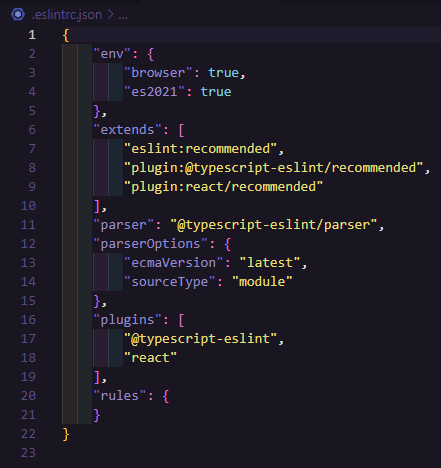


Configurando plugin react-hooks do eslint



Fazer as configurações no vscode;

Era...



Ficou...

{

    "env": {

        "browser": true,

        "es2021": true

    },

    "extends": [

        "eslint:recommended",

        "plugin:@typescript-eslint/recommended",

        "plugin:react/recommended"

    ],

    "parser": "@typescript-eslint/parser",

    "parserOptions": {

        "ecmaVersion": "latest",

        "sourceType": "module"

    },

    "plugins": [

        "@typescript-eslint",

        "react",

        "react-hooks"

    ],

    "rules": {

      "react-hooks/rules-of-hooks": "error",

      "react-hooks/exhaustive-deps": "warn",

      "react/prop-types": "off",

      "react/react-in-jsx-scope": "off",

      "@typescript-eslint/explicit-module-boundary-types": "off"

    },

    "settings": {

      "react": {

        "version": "detect"

      }

    }

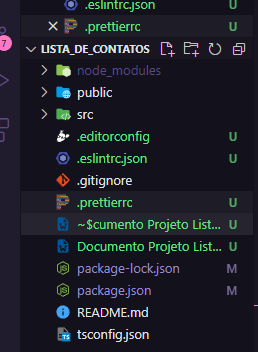
}

Configurar o prettier

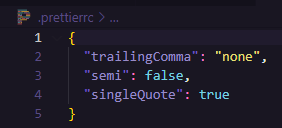




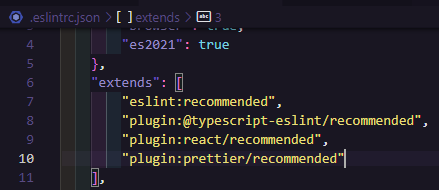
Na raiz dos arquivos no vscode, criar o arquivo ‘.prettierrc’ ;



Fazer a configuração neste arquivo;

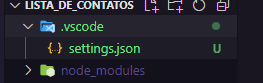


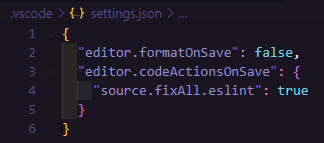
Fazer a integração do prettier com eslint:



Fazer com que o vscode conserte os erros encontrados pelo prettier automaticamente;

Criou a pasta ‘.vscode’ e o arquivo settings.json;





No terminal:



Para automatizar as correções;

Vamos inicializar nosso projeto no terminal:



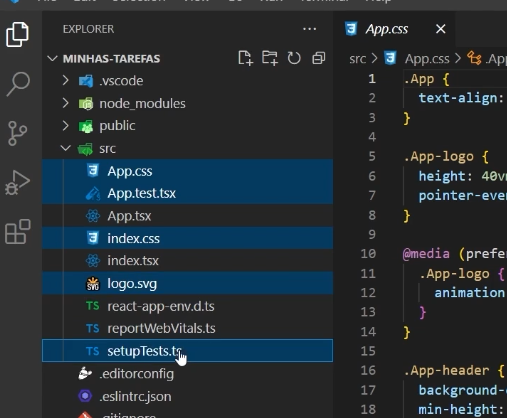
Na pasta Introdução ao CSS in JS com React tem explicação em detalhes, inclusive como adicionar extensão do React developer tools;



Instalar dependências

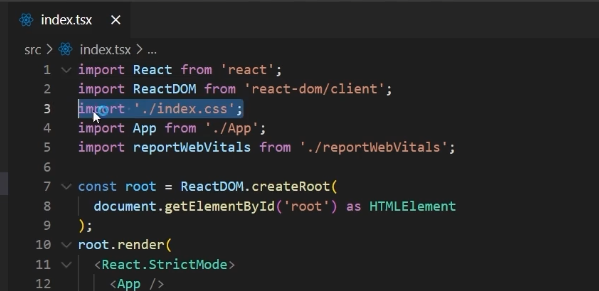


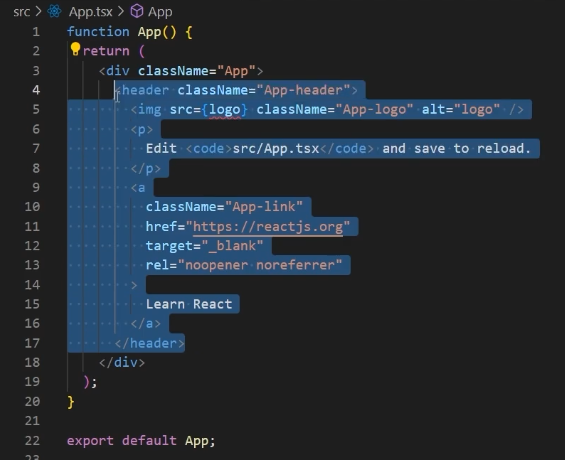
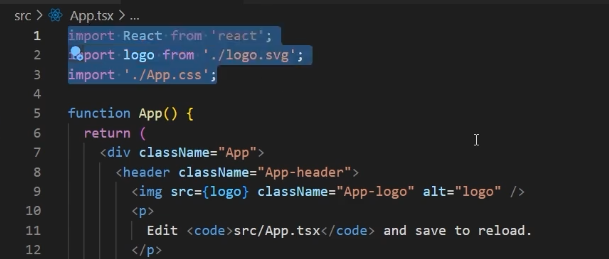
Apagando arquivos desnecessários



Apagando arquivos desnecessários

Remover as referências a esses arquivos deletados;





Deixando apenas a div;

Configurar o styled-components

Voltando ao terminal, fazendo a instalação do styled-components;



Instalando também o pacote de tipos para que o typescript também reconheça o styled componentes;

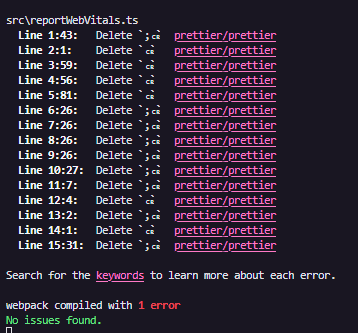


Fazer a instalação do redux;



Inicializar o servidor;





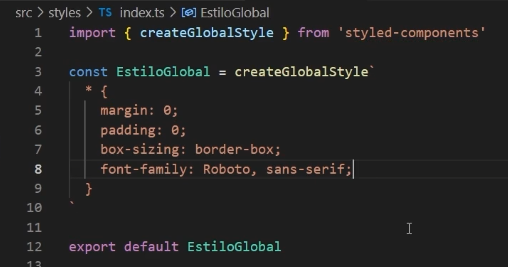
Resolvendo este erro do prettier:



Criar o estilo global

Em src > criou pasta styles > new file index.ts >

Neste arquivo criar o css globalStyle;



Criar estrutura do projeto em um container principal; um h1; um botão para novo contato; um container com os cards de contato;

No card além dos contatos, um botão editar e um remover;

Criar uma página para adicionar contato, com um h2, um container, e neste container, um input para o nome completo, um para email e um para telefone; e um botão adicionar;

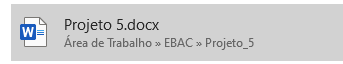
Criar a estrutura de pastas;

Pastas principais: components; containers; pages; store;

Os componentes são: BotaoNovoAdicionar; Contato; Formulario; ListaDeContatos;

Serão divididos BotaoNovoAdicionar e Contato em components; Formulario e ListaDeContatos em containers;

Seguindo...

 page5;

Após preparar o estilo para uma para a lista de contatos, começar a usar o reducers para preencher a lista;

Criar o RootReducer e sua tipagem;

Criar os reducers (slices) que vão determinar as actions;

Criar uma class que representa o contato, para preencher o initialState;

Por convenção será criada na pasta models (modelos de entidades; de abstrações);

Fazer o preenchimento do initialState com a class criada;

Consumir o reducer com Provider;

Fazer uso do reducer para preencher a listagem;

Remover item;

Chamar a action (reducer) criada e despachar uma action;

Fazer uso da class já criada para preencher argumentos passados para o reducer;

Construir a edição de tarefa;