

ESCOLA TÉCNICA 3D

RGI42

CRIANDO UM PROJETO DE SOFTWARE

Conversor de temperatura

RENAN

NOVA IGUAÇU-RJ

2024

Renan

CRIANDO UM PROJETO DE SOFTWARE

Conversor de temperatura

Trabalho feito em prol dos estudos e ganha de pontuação da matéria de LTP I fundamentos de PHP e Técnica de programação (Algoritmo), também para o ganho de experiencia em projetos reais.

Orientador(a): Gabriel Lyra

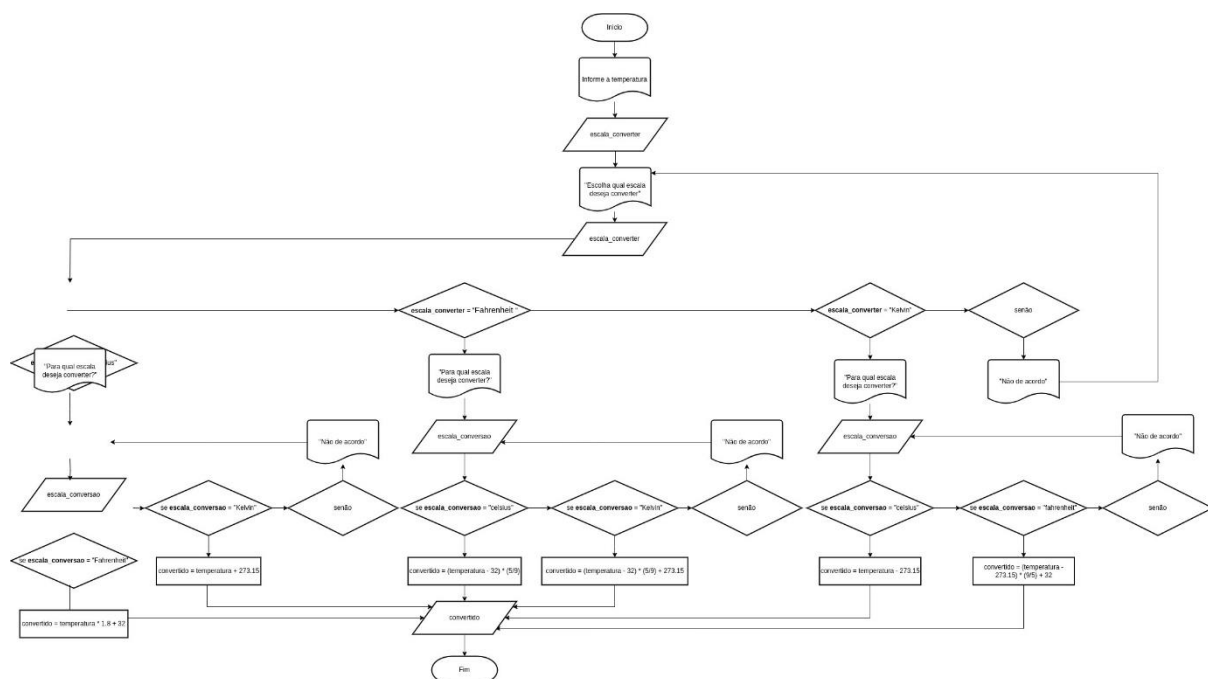
NOVA IGUAÇU-RJ

2024

1ª Etapa, introdução e a base

O programa que irei criar se baseia no tema; "conversor de temperatura", seja celsius, fahrenheit ou kelvin, de escala para escala que o cliente desejar. Usando métodos simples, como uso de <Se> e múltiplos cálculos (cálculos de conversão de cada escala). Então na 1º Etapa trata-se da criação do diagrama e algoritmo para se determinar como funcionará o programa, servindo como base para o código futuro como solicitado, usando HTML, CSS e PHP. Programa contará com 9* <Se>, sendo 3* <Se> para definir a escala da temperatura, ou seja, definir a variável para direcionar o programa ao caminho em específico, logo o restante dos <se> (6) serão usados para o direcionamento certo do restante do programa.

Diagrama criado como base a seguir:



Agora o algoritmo, para se ter como noção a lógica e como será a passagem de um diagrama à um algoritmo, de algoritmo para linguagem de programação PHP, veja a seguir, na próxima página, o algoritmo:

```

incio
| caracter: escala_converter, escala_conversao;
| inteiro: temperatura;
| real: valor_convertido;
|
| valor_convertido <- 0;
| escrever("Digite a temperatura: ");
| ler(temperatura);
| escrever("Digite a escala da temperatura a converter | celsius | fahrenheit | kelvin|: ");
| ler(escala_converter);
| escrever("Digite a escala para qual deseja converter | celsius | fahrenheit | kelvin|: ");
| ler(escala_conversao);
|
| se escala_converter == "celsius"
|     se escala_conversao == "fahrenheit"
|         valor_convertido <- temperatura * 1.8 + 32;
|         escrever("Temperatura convertida para: ", valor_convertido, "°F");
|     fim se
|     se escala_conversao == "kelvin"
|         valor_convertido <- temperatura + 273.15;
|         escrever("Temperatura convertida para: ", valor_convertido, "K");
|     fim se
| fim se
| se escala_converter == "fahrenheit"
|     se escala_conversao == "celsius"
|         valor_convertido <- (temperatura - 32) * (5/9);
|         escrever("Temperatura convertida para: ", valor_convertido, "°C");
|     fim se
|     se escala_conversao == "kelvin"
|         valor_convertido <- (temperatura - 32) * (5/9) + 273.15;
|         escrever("Temperatura convertida para: ", valor_convertido, "K");
|     fim se
| fim se
| se escala_converter == "kelvin"
|     se escala_conversao == "celsius"
|         valor_convertido <- temperatura - 273.15;
|         escrever("Temperatura convertida para: ", valor_convertido, "°C");
|     fim se
|     se escala_conversao == "fahrenheit"
|         valor_convertido <- (temperatura - 273.15) * (9/5) + 32;
|         escrever("Temperatura convertida para: ", valor_convertido, "°F");
|     fim se
| fim se
|
fim.

```

Como pode ver, o código é extenso em questão de linhas, mas a sua intuição é simples e permite uma interpretação fácil do que se trata o programa em si. Com o uso de muitos <Se>, e logo no final de cada, imprimindo o valor que o usuário deseja, agora o objetivo será transportar para uma linguagem de programação, no caso, PHP.

Utilizando também outras linguagens, não de programação, mas para o desenvolvimento do site, o HTML (a estrutura) e as CSS (o estilo).

2º Etapa, de algoritmo para linguagem de programação

Primeiramente, ao invés de estruturar o site, vamos criar o código PHP e o executar no terminal, para que o mais importante, o script, já esteja pronto, e assim só o adaptar ao site. Podemos chamar então o código a seguir, de “Teste1.php”:

```
<?php
$temperatura = readline("Informe o valor da temperatura: ");
$escala_converter = readline("Informe a escala a se converter (celsius | Fahrenheit | Kelvin): ");
$escala_conversao = readline("Informe a escala para qual se deseja converter (celsius | Fahrenheit | Kelvin): ");

echo "Temperatura convertida: ". conversao_temp($temperatura, $escala_converter, $escala_conversao);

function conversao_temp($temperatura, $escala_converter, $escala_conversao){
    if ($escala_converter == "celsius"){
        if ($escala_conversao == "fahrenheit"){
            $valor_convertido = $temperatura * 1.8 + 32;
            return "$valor_convertido °F \n";
        } else if ($escala_conversao == "kelvin"){
            $valor_convertido = $temperatura + 273.15;
            return "$valor_convertido K \n";
        }
    } else if ($escala_converter == "fahrenheit"){
        if ($escala_conversao == "celsius"){
            $valor_convertido = ($temperatura - 32) * (5/9);
            return "$valor_convertido °C \n";
        } else if ($escala_conversao == "kelvin"){
            $valor_convertido = ($temperatura - 32) * (5/9) + 273.15;
            return "$valor_convertido K \n";
        }
    } else if ($escala_converter == "kelvin"){
        if ($escala_conversao == "celsius"){
            $valor_convertido = $temperatura - 273.15;
            return "$valor_convertido °C \n";
        } else if ($escala_conversao == "fahrenheit"){
            $valor_convertido = ($temperatura - 273.15) * (9/5) + 32;
            return "$valor_convertido °F \n";
        }
    }
}
```

O código é simples, porém funcional, foi criado uma <function> para que possa ser solicitada sempre que desejada, será interessante para o site. Observando bem o código até mesmo continua “cru”, pois nem mesmo há a formatação dos números decimais convertidos. Agora, é necessário adaptar o script para HTML, utilizaremos o formulário e o método Post, isso para que possamos armazenar os dados através do formulário e processar no código os dados.

3º Etapa, adaptação

Seguiremos fazendo agora a adaptação, para que possamos executar o script no corpo do site, e não em um terminal. Veja a seguir, o arquivo index criado e formatado para que possamos enviar os dados e o código adaptado a receber esses dados:

HTML:

```
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Teste de site</title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10 <div>
11     <header>Conversor de temperatura</header>
12     <form action="script.php" method="post">
13         Valor da temperatura: <input type="number" name="temperatura" class="tempinput">
14         <select name="escala_converter" id="escala_converter">
15             <option value="celsius">celsius</option>
16             <option value="fahrenheit">fahrenheit</option>
17             <option value="kelvin">kelvin</option>
18         </select>
19         <select name="escala_conversao" id="escala_conversao">
20             <option value="celsius">celsius</option>
21             <option value="fahrenheit">fahrenheit</option>
22             <option value="kelvin">kelvin</option>
23         </select>
24         <input type="submit" value="converter">
25     </form>
26 </div>
27 </body>
28 </html>
```

PHP:

```
1 <?php
2 $temperatura = $_POST['temperatura'];
3 $escala_converter = $_POST['escala_converter'];
4 $escala_conversao = $_POST['escala_conversao'];
5
6 echo "Temperatura convertida: ". conversao_temp($temperatura, $escala_converter, $escala_conversao);
7
8 function conversao_temp($temperatura, $escala_converter, $escala_conversao){
9     if ($escala_converter == "celsius"){
10         if ($escala_conversao == "fahrenheit"){
11             $valor_convertido = $temperatura * 1.8 + 32;
12             return "$valor_convertido °F \n";
13         } else if ($escala_conversao == "kelvin"){
14             $valor_convertido = $temperatura + 273.15;
15             return "$valor_convertido K \n";
16         }
17     } else if ($escala_converter == "fahrenheit"){
18         if ($escala_conversao == "celsius"){
19             $valor_convertido = ($temperatura - 32) * (5/9);
20             return "$valor_convertido °C \n";
21         } else if ($escala_conversao == "kelvin"){
22             $valor_convertido = ($temperatura - 32) * (5/9) + 273.15;
23             return "$valor_convertido K \n";
24         }
25     } else if ($escala_converter == "kelvin"){
26         if ($escala_conversao == "celsius"){
27             $valor_convertido = $temperatura - 273.15;
28             return "$valor_convertido °C \n";
29         } else if ($escala_conversao == "fahrenheit"){
30             $valor_convertido = ($temperatura - 273.15) * (9/5) + 32;
31             return "$valor_convertido °F \n";
32         }
33     }
34 }
```

Como pode ser visto, as únicas alterações feitas no script foram as variáveis em si, para que possam receber os dados do formulário, de acordo com o <name=""> em específico, em html usamos também o <select> e <option> para criamos uma caixa de seleção, evitando até mesmo o erro do usuário caso fosse necessário escrever se escrevesse a escrita da escala que deseja de forma errada. A execução é simples, só acessar o terminal e digitar:

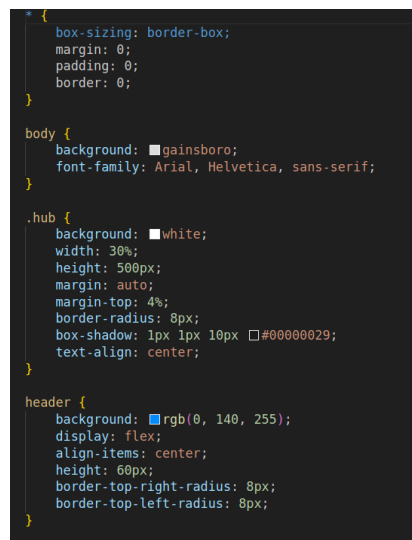
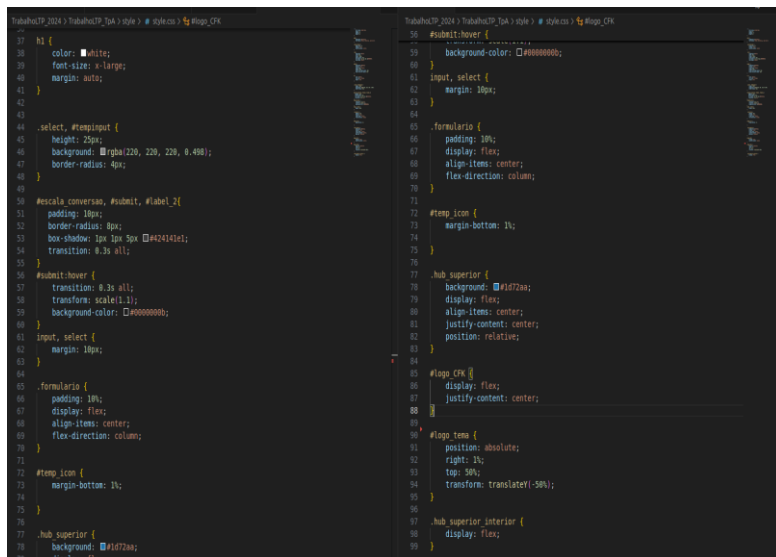
php -S localhost:8080 -t .

Para assim, inicializarmos um servidor “remoto” e conseguirmos ver o código em execução.

E assim, já temos o site praticamente em perfeita execução, mas falta uma coisa, a estética. Um site bom, é aquele que além de funcionar perfeitamente para o cliente, e não o trazer dores de cabeça durante o seu uso, é um site também atraente em seu design para atrair sua atenção e ser agradável aos olhos. Por isso, sendo declarado como projeto final, faremos a estética e o tornar mais atraente aos clientes, sendo também fácil de se compreender e mexer.

4º Etapa, projeto final e ajustes no código

As CSS são as responsáveis por esse trabalho, na estética, através delas poderemos fazer um site atraente aos olhos, meu objetivo é tornar o site minimalista, já porque não precisa ser complexo, farei de acordo com o nível do tema proposto, que é simples. O código em CSS pode se tornar extenso por conta de vários detalhes aqui e ali. Também, será demonstrado o “novo” código, porque estaremos tornando o arquivo “index.html” que solicitava o “script.php” em um “index.php”, mas será demonstrado mais a frente, por agora, cuidaremos do design. Veja a seguir:



Em si, é básico, porém torna o site minimalista, a ideia que eu propus. Veja a seguir o resultado:



Agora, trabalhando com o index.php, no qual eu disse anteriormente. Essa escolha foi tomada pois eu queria que o valor retornado fosse na mesma página e não redirecionado para outra página, sendo assim, foi embutido o script direto no arquivo index e sua extensão foi alterada, de html para php. Logo também fizemos algumas pequenas alterações no arquivo index, além de ter passado todo o script, fazendo com que o <action> seja redirecionado para o próprio index. Outras adições ao código também, como por exemplo no uso de <isset> que determina se a variável considerada é definida, ou seja, se foi declarada e se é diferente de null. Veja a seguir:

```
TrabalhoLTP_2024 > projeto final > index.php > html > head > link
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
7 <title>Teste de site</title>
8 <link rel="stylesheet" href="style.css">
9 </head>
10 <body>
11 <div class="hub">
12 <header><h1>Conversor de temperatura</h1></header>
13 <form action="index.php" method="post" class="formulario">
14 <label id="label_1" form="temperatura">Digite a temperatura: </label><input
15 <div>
16 <label><strong>De: </strong></label><select name="escala_converter">
17 <option value="celsius">°C celsius</option>
18 <option value="fahrenheit">°F fahrenheit</option>
19 <option value="kelvin">K kelvin</option>
20 </select>
21 </div>
22 <div>
23 <label><strong>Para: </strong></label><select name="escala_conversao">
24 <option value="celsius">°C celsius</option>
25 <option value="fahrenheit">°F fahrenheit</option>
26 <option value="kelvin">K kelvin</option>
27 </select>
28 </div>
29 <input type="submit" value="converter" id="submit">
30 </form>
31 <?isset($_POST['temperatura'])&isset($_POST['escala_converter'])&isset($_
32 </div>
33 </body>
34 </html>
35 </php>
36
37 function conversao_temp($temperatura, $escala_converter, $escala_conversao){
38     if ($escala_converter == "celsius"){
39         if ($escala_conversao == "fahrenheit"){
40             $valor_convertido = $temperatura * 1.8 + 32;
41             return "valor convertido °F \n";
42         } else if ($escala_conversao == "kelvin"){
43             $valor_convertido = $temperatura + 273.15;
44             return "valor convertido K \n";
45         }
46     } else if ($escala_converter == "fahrenheit"){
47         if ($escala_conversao == "celsius"){
48             $valor_convertido = ($temperatura - 32) * (5/9);
49             return "valor convertido °C \n";
50         } else if ($escala_conversao == "kelvin"){
51             $valor_convertido = ($temperatura - 32) * (5/9) + 273.15;
52             return "valor convertido K \n";
53         }
54     } else if ($escala_converter == "kelvin"){
55         if ($escala_conversao == "celsius"){
56             $valor_convertido = $temperatura - 273.15;
57             return "valor convertido °C \n";
58         } else if ($escala_conversao == "fahrenheit"){
59             $valor_convertido = ($temperatura - 273.15) * (9/5) + 32;
60             return "valor convertido °F \n";
61         }
62     }
63 }
```

O Resultado é que, ao invés do código retornar o valor que desejamos em outra página, agora será retornada na mesma página e imprimida na tela, sendo mais otimizado para o cliente, sem a necessidade de retornar à mesma página diversas vezes para executar o script novamente. Sendo esse o resultado:



Sendo assim, é finalizado o projeto. Provavelmente, quando chegar em mãos para avaliação talvez haja algumas mudanças em partes estéticas, mas o produto final em si, é isso.

Datas em que me dediquei para fazer o trabalho:

01/11/2024 - Criação do fluxograma e criação do algoritmo em folha de caderno.

02/11/2024 - Criação do script em PHP e estrutura do site HTML, sem focar no design, mas no funcional do site.

07/11/2024 - Criação do arquivo <style.css>, foco na estética do site, buscando tornar mais atraente aos olhos do cliente.

08-09/11/2024 - Atualização do arquivo <index.html> para <index.script>, e modificações no design do site, sendo a versão final tanto do código adaptado à HTML quanto a estética.

Aproximadamente, o meu projeto foi feito em 4 dias. Ao meu ver, o tema foi simples, fácil de se compreender e passar para um fluxograma, algoritmo e linguagem de programa. Só foi necessário a ajuda do Orientador, Gabriel Lyra, em certos momentos, como no uso do <display flex> e no uso do PHP embutido no próprio arquivo index. Porém, em uma visão geral, o trabalho foi fácil e agradável de se fazer, tendo diversos benefícios como experiência na criação de um projeto e a compreensão sobre a importância de se fazer um site em passos, começando pela base e gradualmente fazendo aos poucos. Agradeço se leu até aqui, até.