**ESCOLA SENAI “PROFESSOR VICENTE AMATO”**

Curso Técnico de Informática

FELIPE MARQUES STEFANI

**DESCRIÇÃO NARRATIVA – PROJETO FAST PARKING**

JANDIRA

2019

Criação do arquivo main.js onde serão armazenadas as funções JavaScript.

Função abrirModal(el):

* Adicionar à classList do elemento a classe desejada para abrir o modal.

Função fecharModal(el):

* Remover da classList do elemento a classe desejada para abrir o modal.

Função novoCadastro( ):

* Ela chama a função abrirModal usando o elemento desejado.

Função fechar( ):

* Ela chama a função fecharModal usando o elemento desejado.

Função limitarCaracteres( ):

* Limita os caracteres dos inputs.

Função horaAtualFormatada( ):

* É criada uma variável do tipo Date chamada data e a partir dela são capturadas informações da hora como horas e minutos, então essas horas e minutos são formatadas para o padrão do usuário.

Função dataAtualFormatada( ):

* É criada uma variável do tipo Date chamada data e a partir dela são capturadas informações da hora como data, mês e dia; então essas informações são formatadas para o padrão do usuário.

Função cadastrarVeículo( ):

* O botão que é usado como listener para essa função tem duas funcionalidades então começamos com uma condicional: se o texto do botão for “Adicionar” então ele chama a função. São criadas três variáveis data, do tipo Date, modeloCarro e placaCarro. É criado um objeto carro onde ele recebe modelo, placa, data, data\_saída, milissegundosEntrada e hora. Há outra condicional para verificar se os campos estão vazios, depois mais uma condicional para validar a placa, se a placa estiver digitada corretamente, então a função prosseguirá, senão, alertará o usuário para inserir a placa corretamente. Se ela prosseguir então ela entrará em mais uma condicional para verificar se o banco é nulo, caso for ele criará o banco, caso não for nulo a função irá inserir o objeto no banco.

Função abrirComprovante(placa,modelo,data,hora):

* Ela pega os elementos comp\_modelo, comp\_placa, comp\_data e comp\_hora no HTML e insere neles as informações que lhe foram atribuídas no momento de criação.

Função mostraRelatorio( ):

* Essa função lê o banco onde os elementos tenham data de saída diferente de nulo, lista eles na tabela do relatório.

Função abrirRelatorio( ):

* Chama a função mostraRelatorio( ) e abre o modal do relatório.

Função editarCarro(data):

* Lê o banco atrás do elemento que tenha a data igual à data que lhe foi passada como parâmetro, troca o texto dos inputs para os valores do modelo e placa e atribui a placa a variável carroEditar que será usada mais tarde e troca o texto do botão adicionar para “Atualizar”

Função cadastrarVeiculo( ):

* Se o texto do botão adicionar for “Atualizar” ele chama a função atualizarRegistro( ) passando a variável carroEditar como parâmetro.

Função atualizarRegistro(placa):

* Lê o banco e descobre o índice de qual elemento queremos, cria um objeto registro com as mesmas informações que esse objeto já tinha mas mudando a placa e o modelo. Depois disso ele substitui essas informações, grava no banco novamente, muda o texto do botão adicionar para “Adicionar” e zera os inputs.

Função excluirCarro(placa):

* Lê o banco dos preços para confirmar se o usuário inseriu um preço, senão a função irá alertar o usuário a colocar um preço. É criada uma variável data, do tipo Date, a função lê o banco, procura qual elemento queremos excluir, é criado um novo registro que muda a data\_saida, os milissegundosSaida e a hora\_saida do objeto. Ele faz a mudança no banco e atualiza a tabela.

Função calcularTempoEValor(placa):

* Lê o banco, encontra o índice, a data de saída e a data de entrada em milissegundos, encontra a diferença das duas, transforma isso para horas e minutos e faz o cálculo da quantidade de horas o carro em questão ficou estacionado com a tolerância de 0 minutos. Lê o banco dos preços para saber quanto custa a primeira e demais horas, faz o cálculo do quanto deve ser pago baseado na quantidade de horas que o carro ficou estacionado e retorna o valor a pagar.

Função mostraPatio( ):

* Lê o banco, faz um filter dele e cria a tabela baseada nos dados dos objetos que estão armazenados no banco.

Função cadastrarPreco( ):

* Pega o valor digitado no input e armazena numa variável, essa variável é armazenada num objeto no WebStorage; possui uma condicional para checar se os inputs estão vazios e possui mais uma condicional para checar se o banco está vazio, se estiver, a função o cria, senão ela substitui o último valor pelo novo valor digitado.

Função validarNumeros:

* Verifica se o texto inserido no input são números

Função validarTextos:

* Verifica se o texto inserido no input são caracteres de texto

Função filtroPlaca:

* Verifica se o texto inserido no input são números e caracteres de texto

Função adicionarHifen:

* Adiciona o Hífen na placa.

Função trocarParaMaiuscula:

* Troca o texto da placa para maiúscula.

Função adicionarVirgula:

* Adiciona a virgula do preço.

Função mascaraPreco:

* Cria a máscara para o preço com as funções acima.

Função mascaraPlaca:

* Cria a máscara para a placa com as funções acima.

Keyup listeners: quando uma tecla for pressionada, chama a função x.

Click listeners: quando um elemento for clicado, chama a função x.