

# DOCUMENTO DO CODIGO FONTE

## DA

## DIGITAL MARK

### CLASSE PROJETO

Na classe projeto tivermos que torná-la abstrata para prover uma base de dados de atributos pra que outra classe herdem os seus atributos.

Iniciamos o método **CONSTRUCTOR** na classe, para podemos determinar uma ordem de como cada membro é inicializado, seja automaticamente ou manualmente, repetindo os atributos e logo após inicializando com os comandos **THIS.ATRIBUTO=ATRIBUTO**.

Inserindo o **MÉTODO ENCAPSULAMENTO** dos atributos **CUSTO, PRAZO E DESCRIÇÃO** permitindo que o código tenha acesso a características de uma classe que esteja protegida.

Iniciamos o método **MOSTRA DADOS** para podemos imprime os dados inseridos em uma tabela e para podemos analisa o código.

Estanciamos com uma **VARIAVEL CONST** os atributos da classe para ver os resultados colocados.

### CLASSE TAREFA

Na classe tarefa tivermos que adicionar método **EXTENDS** de projeto para poder herda os atributos da classe pai.

Iniciamos o método **CONSTRUCTOR** na classe, para podemos determinar uma ordem de como cada membro é inicializado, seja automaticamente ou manualmente, repetindo os atributos e logo após inicializando com os comandos **THIS.ATRIBUTO=ATRIBUTO**.

Inserindo o **MÉTODO ENCAPSULAMENTO** do atributo **MATRICULA** permitindo que o código tenha acesso a características de uma classe que esteja protegida.

Iniciamos o método **HERANÇA** para podemos inicia os atributos na classe pai com o comando **SUPER ()**.

Iniciamos o método **MOSTRA DADOS** para podemos imprime os dados inseridos em uma tabela e para podemos analisa o código.

Estanciamos com uma **VARIAVEL CONST** os atributos da classe para ver os resultados colocados.

### CLASSE EQUIPE

Na classe equipe tivermos de criar uma **ARRAY DE FUNCIONARIO** para podemos adicionar vários funcionários na classe equipe.

Iniciamos o método **CONSTRUCTOR** na classe, para podemos determinar uma ordem de como cada membro é inicializado, seja automaticamente ou manualmente, repetindo os atributos e logo após inicializando com os comandos **THIS.ATRIBUTO=ATRIBUTO**.

Iniciamos o método **MOSTRA DADOS** para podemos imprime os dados inseridos em uma tabela e para podemos analisa o código.

Estanciamos com uma **VARIAVEL CONST** os atributos da classe para ver os resultados colocados.

## **ARRAY DE FUNCIONARIO**

Iniciamos a array para podemos usa os dados dos atributos dessa array na classe **EQUIPE**.

Iniciamos o método **CONSTRUCTOR** na array direto nos atributos da array, para podemos determinar uma ordem de como cada membro é inicializado, seja automaticamente ou manualmente,

Inserindo o **MÉTODO ENCAPSULAMENTO** do atributo **MATRICULA** permitindo que o código tenha acesso a características de uma classe que esteja protegida.

Iniciamos o método **MOSTRA DADOS** para podemos imprime os dados inseridos em uma tabela e para podemos analisa o código.

Estanciamos com uma **VARIAVEL CONST** os atributos da classe e da array para ver os resultados colocados.

I finalizando o comando **CONSOLE.LOG** com as estancia da array, para imprime o resultado do código .