# Teste Quero Education

O processo é simples e preferi usar código PHP puro, acho que fica mais claro.

Há a possibilidade de tratar de modo que chamei de procedural ou com banco de dados (que preferi usar) que permitiria ajustes futuros como “só relatar” ou outros que se imaginar.

Uso um funcoes\_padrao.php que contem algumas corriqueiras de meu dia a dia de trabalho.

Criei o banco de dados descrito mais abaixo em meu site, onde também subi a aplicação.

Para rodar a API criei o chama.php que passa os dados para o índex.php que poderá ser chamado por outros processos de front end.

A API pode ser acessada por em <https://www.deapoio.com.br/cursos/chama.php>.

A resposta tem a formatação de saída solicitada e ainda 2 objetos um JSON e outro Matriz.

# Respostas ao questionamento do email

* Quantos anos de desenvolvimento de software (backend, principalmente) você possui? Fale resumidamente sobre sua experiência mais desafiadora e suas responsabilidades nela.

Trabalho com TI desde os tempos da Faculdade em 1970 com um IBM/360.

Usava na engenharia, passando por Clipper e C desde o DR-Dos.

De 1989 em diante passo a me dedicar somente à TI, onde fui gerenciar sistemas de engenharia e produção na Bardella S.A. usando Cobol e um mainframe. Lá criei a primeira rede de PC´s.

No breve currículo que também disponibilizo no git há mais informação de minha longa carreira.

Com desenvolvimento WEB estou há cerca de 6 anos no desenvolvimento.

* Grandes projetos com ótimas práticas de desenvolvimento que lhe permitem escalar o produto fazem toda a diferença. Qual o maior projeto que você participou? Em relação a quantidade de usuários, time de desenvolvimento e tempo de vida do projeto? Comente falando de práticas utilizadas que permitem o produto a crescer e escalar de maneira sustentável.

Na Bardella coordenei equipe de desenvolvimento com 28 colaboradores (7 equipes) montando análise para todos tanto para os processos de Engenharia como de Produção e chão de Fábrica, integrando vários devices.

Desenvolvi projeto de coleta de dados para a Voith (Siemens) e muitos outros.

Usei frameworks como Progress e Magic mas voltei-me para a WEB para trabalhar com código aberto.

Sempre estudando, procuro oportunidade de voltar a trabalhar em equipe, onde a dinâmica nos permite aprender mais e mais rápido.

Estudei um pouco Angular e NodeJS.

No site [www.deapoio.com.br](http://www.deapoio.com.br) há alguns trabalhos realizados.

Desenvolvi o [www.sexoamigo.com.br](http://www.sexoamigo.com.br) que esta em faze de desenvolvimento de marketing.

* Qual sua linguagem de programação favorita (principal)? Fale um pouco sobre as características dela, pontos fortes e fracos. E em Ruby on Rails ou Node, quanto tempo de desenvolvimento?

Como disse passei por alguns frameworks. Não posso dizer que desenvolvi com algum desses citados, mas tenho muita facilidade de apreendê-los.

* Comente algumas boas práticas de desenvolvimento que você conhece e utiliza.

Desde sempre tive que otimizar desenvolvimento criando padronizações, tendo mesmo desenvolvido uma frameword em Clipper e C para desenvolver pequenas aplicações.

Faz parte de minha cultura usar códigos padronizados onde quer que esteja.

O Magic, para quem não conhece é um primor em padronização.

* Como você considera o seu nível (de Júnior a Sênior)? Por que?

Tenho conhecimentos de processos e cases que me coloca como Senior, mas como desenvolvedor WEB ainda me considero somente Pleno.

# Processo I - Usando BD

Ler Entrada até o fim

Sim

Sim

Ǝ 'lightning'

Ǝ 'min'

Mensagem: Palestra 'Entrada[n]' sem tempo definido. Não será agendada.

Não

Quebrar string nesse espaço em 2 campos: $palestra e $duracao=5

Buscar primeiro espaço anterior a 'min'

Buscar primeiro espaço anterior a 'lightning'

Quebrar string nesse espaço em 2 campos: $palestra e $duracao

Terminou Entrada?

Não

CarregarTabelaBD()

Ler palestras

order by duracao DESC

Sim

Busca periodo com tempo\_restante <= duracao

CriaPeriodo()

Achei ?

Não

Sim

AjustaTempoRestante

Mensagem: Palestra 'Entrada[n]' Sem Espaço na Agenda de **$max\_t** etapas

AnotaPeriodo($trilha, $periodo) com $ultimo\_horario e $tempo\_restante

AnotaPalestraLida

com $trilha, $periodo, hora\_inicio

AjustaFimPeriodoV()

## Parametrização

a) **max\_t :** número máximo de trilhas se informado, senão o default = 100

## Banco de Dados

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET AUTOCOMMIT = 0;

START TRANSACTION;

SET time\_zone = "+00:00";

-- Banco de dados: `\_cursos`

## -- Estrutura da tabela `palestras`

## CREATE TABLE `palestras` (

## `trilha` int(2) NOT NULL,

## `periodo` varchar(1) NOT NULL COMMENT 'Matutino,Vespertino',

## `palestra` varchar(150) NOT NULL,

## `duracao` varchar(2) NOT NULL,

## `hora\_inicio` time NOT NULL

## ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

## -- Estrutura da tabela `parametros`

## CREATE TABLE `parametros` (

## `max\_t` int(2) NOT NULL,

## `ultima\_trilha` int(2) NOT NULL,

## `ultimo\_periodo` varchar(1) NOT NULL,

## `minimo\_horario` time NOT NULL

## ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

## -- Estrutura da tabela `periodos`

## CREATE TABLE `periodos` (

## `trilha` int(2) NOT NULL,

## `periodo` varchar(1) NOT NULL,

## `ultimo\_horario` time NOT NULL COMMENT 'Matutino,Vespertino',

## `tempo\_restante` time NOT NULL,

## `maximo\_horario` time NOT NULL,

## `minimo\_horario` time NOT NULL

## ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

## -- Índices para tabela `palestras`

## ALTER TABLE `palestras`

## ADD PRIMARY KEY (`trilha`,`periodo`,`hora\_inicio`) USING BTREE,

## ADD UNIQUE KEY `palestra` (`palestra`),

## ADD KEY `duracao` (`duracao`);

## -- Índices para tabela `periodos`

## ALTER TABLE `periodos`

## ADD PRIMARY KEY (`trilha`,`periodo`) USING BTREE;

## COMMIT;

## CarregarTabelaBD()

* se **não** encontrar **$Entrada[n]=>palestra** em **palestras** (que também é chave única )

Adiciona à Tabela **palestras** no BD

* senão

Mensagem de erro: 'Palestra duplicada na matriz de entrada'.

## CriaPeriodo()

Lê parametros: `**max\_t**`, `**ultima\_trilha**`, `**ultimo\_periodo**`.

* se **parametros:ultimo\_periodo = 'V' =>**

**$ULTIMA\_TRILHA** = **parametros:ultima\_trilha** + 1

**$ULTIMO\_PERIODO** = '**M**'

* senão

**$ULTIMO\_PERIODO** = '**V**'

* se

**parametros:ultima\_trilha > parametros:max\_t =>** MOSTRA ERRO

* senão
* se **$ULTIMO\_PERIODO** = '**M**'

**$MAXIMO\_HORARIO** = 12:00h

* se **$ULTIMO\_PERIODO** = '**V**'

**$MAXIMO\_HORARIO** = 17:00h

(nunca terminar antes das 16:00h)

**$ULTIMO\_HORARIO** = 0

Regravar novos parâmetros;

Criar Novo Periodo;

## AjustaTempoRestante

## $HORA\_DE\_INICIO = $linha\_periodo['ultimo\_horario']

## $HORA\_FINAL = $HORA\_DE\_INICIO + $duracao

## $TEMPO\_RESTANTE = $MAXIMO\_HORARIO - $HORA\_FINAL

## AnotaSaida($trilha, $periodo,$Temp[n,palestra],$hora\_inicio)

Adiciona na matriz **Saida** (inicializada vazia)

## AjustaFimPeriodoV()

Caso algum $período(V) final seja menor que 16:00h teremos que remanejar palestas que estejam em outros períodos de modo que o todos os vespertinhos terminem depois das 16h.

**Não completei essa funcionalidade, pois pelo que entendi no e-mail só se os dados apresentassem configuração ela seria necessária.**

# Processo II - Só Procedural

Ler Entrada até o fim

Sim

Sim

Ǝ 'lightning'

Ǝ 'min'

Mensagem: Palestra 'Entrada[n]' sem tempo definido. Não será agendada.

Não

Quebrar string nesse espaço em 2 campos: $palestra e $duracao=5

Buscar primeiro espaço anterior a 'min'

Buscar primeiro espaço anterior a 'lightning'

Quebrar string nesse espaço em 2 campos: $palestra e $duracao

Terminou Entrada?

para $trilha =1 a **$max\_t**

$gravei= **false**

Ler $Temp até o fim

SortearTemp()

Não

CarregarTemp()

Sim

$periodo = 'V'

$periodo = 'M'

AnotaSaida ($trilha, $periodo, Temp[n])

$gravei= **true**

Sim

AjustaTempoRest ($trilha,$periodo, $Temp[duracao])

$TRILHA[ $trilha, $periodo, temporestante] < $Temp[n,duracao] ?

Não

Sim

$gravei?

Não

Mensagem: Palestra 'Entrada[n]' Sem Espaço na Agenda de **$max\_t** etapas

$i++

AjustaFimPeriodoV()

## Parametrização

a) **max\_t :** número máximo de trilhas se informado, senão o default = 100

## CarregarTemp()

se não encontrar **$Entrada[n]=>palestra** em **$Temp**

Adiciona à matriz **$Temp[$palestra,$duracao]** declarada no início como vazia.

senão

Mensagem de erro: 'Palestra duplicada na matriz de entrada'.

## SortearTemp()

Ordena a matriz **$Temp** ordem **decrescente de $duracao**.

## AjustaTempoRest($trilha,$periodo,$duracao)

global $hora\_de\_inicio

Regras:

Tantas Trilhas quanto parametrizado em **$max\_t**

Cada trilha:

se **$periodo** = 'M'

**$fim\_período** = sessão matinal de 09:00-12:00h no máximo

se $periodo = 'V'

**$fim\_período** = sessão vespertina da 13:00-16:00 prorrogável até 17:00h

(nunca terminar antes das 16:00h)

Matriz **$TRILHA**[periodo,numero\_trilha,ultimo\_final,tempo\_restante]

$hora\_de\_inicio = $TRILHA[$trilha,$periodo, ultimo\_final]

$hora\_final = $hora\_de\_inicio + $duracao

$tempo\_restante = **$fim\_período** - $hora\_final

$TRILHA[periodo,numero\_trilha,tempo\_restante] = $tempo\_restante

## AnotaSaida($trilha, $periodo,$Temp[n,palestra],$hora\_inicio)

Adiciona na matriz **Saida** (inicializada vazia)

## AjustaFimPeriodoV()

Caso algum $período(V) final seja menor que 16:00h teremos que remanejar palestas que estejam em outros períodos de modo que o todos os vespertinhos terminem depois das 16h.