

CPAPI020 - Balanceamento do Item Veículo CP

 Tempo aproximado para leitura: 6 minutos

Documento de API

Produto:	Datasul
Ocorrência:	Documentação de API
Nome Físico	cpp/cpapi020.p

Objetivo: Realizar o balanceamento do veículo, e funções de conversão.

Considerações Gerais:

- Devem ser passadas 2 (duas) temp-tables como parâmetro, sendo as duas de entrada e saída, e mais um parâmetro lógico (yes/no) que define se os erros já existentes na temp-table de erros serão eliminados no início da execução.

- Exemplo de execução da API:

```
run cpp/cpapi020.p (input-output table tt-balanceia,  
input-output table tt-erro,  
input yes).
```

- A API funciona da seguinte forma: Recebe como parâmetros as temp-tables, faz as críticas das mesmas, caso não tenha ocorrido nenhum problema retorna a quantidade resultante do balanceamento de um item concentrado com o veículo, que pode implicar em uma quantidade requisitada maior ou menor do veículo.

- Se o programa chamador necessita executar a api para vários balanceamentos pode-se criar um registro na temp-table tt-balanceia com cada requisição a ser balanceada, para depois executar a API.

- O campo quant-veiculo irá retornar o resultado do balanceamento este somado a quantidade da reserva do veículo implicará na quantidade a ser requisitada para o veículo.

Parâmetros de Entrada

Temp-table tt-balanceia: nesta temp-table deverá ser armazenado o item da reserva da ordem que será balanceado. O campo quant-requis deve levar a quantidade a ser requisitada ao fator da reserva, o campo quant-movto deve levar a quantidade a ser requisitada ao fator do lote, para encontrar essa quantidade pode-se utilizar a função f-conv-qtde descrita mais abaixo.

TEMP-TABLE TT-BALANCEIA				
Atributo	Tipo	Formato	Valor Inicial	Obrigatório
nr-ord-produ	Integer	">>>,>>>,>>9"	?	Sim
lt-codigo	Character	"x(16)"	?	Sim
quant-requis	Decimal	">>>>,>>9.9999"	?	Sim
quant-movto	Decimal	">>>>,>>9.9999"	?	Sim
quant-veiculo	Decimal	">>>>,>>9.9999"	?	Não
cod-Versão-Integracao	Integer	"999"	0	Sim

Parâmetros de Saída

Temp-table tt-erro: Para cada balanceamento serão executadas validações para consistir os registros, os erros encontrados serão gravados na temp-table tt-erro.

TEMP-TABLE TT-ERRO			
Atributo	Tipo	Formato	Valor Inicial
i-sequen	Integer		0
cd-erro	Integer		0
Mensagem	Character	“x(255)”	?

Obs: O campo cd-erro possui o número do erro do cadastro de mensagem; O campo mensagem possui a descrição da mensagem do cadastro de mensagem; O campo i-sequen possui o número sequencial do erro;

Execução

Execução:

- O programa **cpp/cpapi020.p** irá executar as seguintes validações:

Número do Erro	Mensagem	Ajuda
16	Parâmetros globais não cadastrados	Parâmetros globais não cadastrados.
200	Parâmetros da Produção não cadastrados	Os Parâmetros da Produção devem ser cadastrados
27372	Função não implantada	A Função de Fator de Concentração deve estar implantada
3941	Versão de integração incorreta	A versão de integração informada através dos parâmetros internos é incompatível com a versão atual da API. Favor contatar suporte técnico.
27817	A ordem &1 possui mais de uma reserva como veículo	Cada ordem de produção pode possuir apenas uma reserva parametrizada como veículo, se este erro ocorrer deve ser revista a engenharia de produto
27474	Não há fator de conversão de &1 para &2	Deve ser cadastrado o fator de conversão entre as unidades de medida

Execução das funções internas

Esta API fornece algumas funções internas que podem ser acessadas se a API for executada persistente e a partir da definição da include `cpp/cpapi020.i1`.

Execução persistente:

```
run cpp/cpapi020.p persistent set h-cpapi020 (input-output table tt-balanceia,
                                     input-output table tt-erro,
                                     input      yes).
```

Procedure `pi-balanceia-veiculo`: Realiza o balanceamento do veículo como se a API fosse executada normalmente.

Exemplo:

```
run pi-balanceia-veiculo in h-cpapi020 (input-output table tt-balanceia,
                                     input-output table tt-erro,
                                     input      yes).
```

Função `f-conv-qtde`: Converte a quantidade de um item de seu fator original para o fator de um lote e vice-versa.

```
function f-conv-qtde returns decimal
```

```
(p-qtde      as dec,
```

p-fator-orig as dec,
p-fator-dest as dec) in h-cpapi020.

Função f-conv-per-ppm: Converte o fator de um item de Concentração para PPM.

p-acao: 1- Concentração p/ PPM
2- PPM / Concentração

function f-conv-per-ppm returns decimal

(p-valor as dec,
p-acao as int) in h-cpapi020.

Função f-retorna-fator: Retorna o fator de conversão entre duas unidades de medida de um item.

function f-retorna-fator returns decimal

(p-it-codigo as char,
p-fm-codigo as char,
p-un-orig as char,
p-un-dest as char) in h-cpapi020.

Função f-conv-un: Retorna a quantidade informada convertida da sua unidade original para a destino.

function f-conv-un returns decimal

(p-qtde as dec,
p-it-codigo as char,
p-fm-codigo as char,
p-un-orig as char,
p-un-dest as char) in h-cpapi020.

Procedure pi-finalizar: Deve ser executada quando não for mais necessário utilizar nenhum método API.

Exemplo:

run pi-finalizar in h-cpapi020.

 Macro desconhecida: 'rate'

[base_de_conhecimento](#) [cpapi020](#) [mcp](#)



[Política de
privacidade](#)

[Termos
de uso](#)