

# Webservice - Consulta de CEP via JEX

Neste tópicos veremos como fazer uma requisição do webservice de CEP via JEX. O Webservice que será usado é o da [ViaCep](#), um webservice gratuito que busca todos os CEP's do Brasil.

## Conceito:

CEP (Código postal) ou (Código de Endereçamento Postal) é um código desenvolvido pelas administrações postais e criado com o intuito de facilitar a organização logística e localização espacial de um endereço postal.

Acessando o site da ViaCep, você encontra toda a documentação necessária para realizar as requisições JSON, XML, entre outras, nos formatos de retorno.

image-1640259727719.png

No print, você pode ver quais os campos criados para puxar as informações conforme o CEP for preenchido. A principal configuração nesse processo, irá partir do primeiro formulário, o campo CEP.

image-1640259764635.png

O CEP possui um evento logo na saída do formulário. Ou seja, uma vez que sair desse campo, clicando fora dele ou pressionando enter/tab, uma série de eventos irá acontecer.

image-1640259794721.png

O código inserido no PEX basicamente executa: Caso o ViaCep retorne as informações do endereço daquele CEP, então, o valor deles no campo deve ser preenchido, fazendo com que as informações sejam gravadas de forma automática. Claro, apenas caso o CEP exista.

Código JEX:

```
async function aoSairCampoFormulario() {  
  
  if (!formulario.CEP.valor) return;  
  const cep = formulario.CEP.valor;  
  const retBuscaCEP = await axios.get('https://viacep.com.br/ws/' + cep + '/json', {}  
);
```

```
formulario.LOGRADOURO.valor = retBuscaCep.data['logradouro']  
formulario.COMPLEMENTO.valor = retBuscaCep.data['complemento']  
formulario.BAIRRO.valor = retBuscaCep.data['bairro']  
formulario.LOCALIDADE.valor = retBuscaCep.data['localidade']  
formulario.UF.valor = retBuscaCep.data['uf']  
formulario.IBGE.valor = retBuscaCep.data['ibge']  
formulario.GIA.valor = retBuscaCep.data['gia']  
formulario.DDD.valor = retBuscaCep.data['ddd']  
formulario.SIAFI.valor = retBuscaCep.data['siafi']
```

```
}
```

---

Revisão #3

Criado 23 December 2021 08:38:19 por Nicolly Andrielly

Atualizado 24 August 2022 16:52:04 por Nicolly Andrielly