

# JEX - Função gerar planilha

**Essa melhoria foi implementada na versão 17.8.3.613 do sistema DOX, importante verificar se a versão está devidamente instalada antes de seguir instruções ou testar o código abaixo.**

**\*\* Além da adição interna da lib SheetJS no serviço JS Sandbox, foi adicionado na aba 'Assistente' de telas de evento em JavaScript, o novo item 'Planilhas (SheetJS)', com as opções:**

- Gerar planilha OpenDocument: Vai inserir código base para geração de uma planilha ODS em um arquivo.
- Gerar planilha Microsoft: Vai inserir código base para geração de uma planilha XLS em um arquivo.

Também foi adicionado em 'Documentações' o link para mais detalhes sobre a lib SheetJS (XLSX), assim como acesso direto no menu de complementação do código (CTRL + Barra de espaço).

Em anexo, está o processo utilizado de teste para validar a funcionalidade. Abaixo, está o código JEX:

```
const clientes      = await conexaoBD.executarConsultaSQL("select idcliforemp, razao from cliforemp where tipocliforemp = 0 and inativo = 'N'");
const fornecedores = await conexaoBD.executarConsultaSQL("select idcliforemp, razao from cliforemp where tipocliforemp = 1 and inativo = 'N'");
const empregados   = await conexaoBD.executarConsultaSQL("select idcliforemp, razao from cliforemp where tipocliforemp = 2 and inativo = 'N'");

const executivesResponse = await axios.get('https://sheetjs.com/data/executive.json');
const executives = executivesResponse.data.map(row => ({
  name: row.name.first + " " + row.name.last,
  birthday: row.bio.birthday
}));

const abaClientes = XLSX.utils.json_to_sheet(clientes, { skipHeader: true });
const abaFornecedores = XLSX.utils.json_to_sheet(fornecedores);
XLSX.utils.sheet_add_aoa(abaFornecedores, [["Código", "Razão"]], { origin: "A1" });
const abaEmpregados = XLSX.utils.json_to_sheet(empregados);
XLSX.utils.sheet_add_aoa(abaEmpregados, [["Código", "Razão"]], { origin: "A1" });
const abaExecutives = XLSX.utils.json_to_sheet(executives);
```

```
const planilha = XLSX.utils.book_new();
XLSX.utils.book_append_sheet(planilha, abaClientes, "Clientes");
XLSX.utils.book_append_sheet(planilha, abaFornecedores, "Fornecedores");
XLSX.utils.book_append_sheet(planilha, abaEmpregados, "Empregados");
XLSX.utils.book_append_sheet(planilha, abaExecutives, "Executives");

const arquivoXLSX = 'C: /Planilha/Pessoas.xlsx';
await fsPromises.rm(arquivoXLSX, { force: true });
XLSX.writeFile(planilha, arquivoXLSX, { compression: true });

const arquivoODS = 'C: /Planilha/Pessoas.ods';
await fsPromises.rm(arquivoODS, { force: true });
XLSX.writeFile(planilha, arquivoODS, { compression: true, bookType: 'ods' });

mensagem.texto = 'Todas as planilhas exportadas com sucesso!';
```

---

Revisão #1

Criado 5 Abril 2023 08:46:46 por Nicolly Andrielly

Atualizado 24 Abril 2023 15:53:22 por Nicolly Andrielly