

## Arquivo: importar\_tr.py

```
from PyQt6.QtWidgets import *
from PyQt6.QtGui import *
from PyQt6.QtCore import *
from modules.utils.add_button import add_button

class TermoReferenciaWidget(QWidget):
    abrirTabelaNova = pyqtSignal()
    carregarTabela = pyqtSignal()
    configurarSqlModelSignal = pyqtSignal()

    def __init__(self, parent, icons):
        super().__init__(parent)
        self.setWindowTitle("Termo de Referência")
        self.resize(800, 600)
        self.parent = parent
        self.icons = icons
        # Configuração do layout principal
        self.layout = QVBoxLayout(self)

        # Configuração do título

        # Adicionar layout de botões ao layout principal
        title_layout = QHBoxLayout()

        title_layout.addStretch()

        title = QLabel("Especificação do Termo de Referência")
        title.setAlignment(Qt.AlignmentFlag.AlignCenter)
        title.setFont(QFont('Arial', 16, QFont.Weight.Bold))
        title_layout.addWidget(title)

        # Usando add_button para criar e adicionar botões
        add_button("Abrir Tabela Nova", "excel_down", self.abrirTabelaNova, title_layout,
self.icons, tooltip="Cria e abre uma nova tabela em Excel", button_size=(200, 30))
        add_button("Carregar Tabela", "excel_up", self.carregarTabela, title_layout, self.icons,
tooltip="Carrega uma tabela existente para o banco de dados", button_size=(200, 30))

        title_layout.addStretch()

        # Adicionar layout de botões ao layout principal
        self.layout.addLayout(title_layout)

        title1 = QLabel("Este passo é necessário para obter as especificações do termo de
referência, que não constam no termo de homologação ou no comprasnet.")
        title1.setAlignment(Qt.AlignmentFlag.AlignLeft)
        title1.setFont(QFont('Arial', 12))
        self.layout.addWidget(title1)

        title2 = QLabel("Importante! O índice da tabela deve ser 'item', 'catalogo', 'descricao' e
'descricao_detalhada'.")
        title2.setAlignment(Qt.AlignmentFlag.AlignLeft)
```

```

title2.setFont(QFont('Arial', 12))
self.layout.addWidget(title2)

# Configurar o QTableView para exibir os dados
self.table_view = QTableView(self)
self.table_view.setSelectionBehavior(QTableView.SelectionBehavior.SelectRows) # Seleciona
a linha inteira
self.table_view.verticalHeader().setVisible(False) # Oculta a coluna de índice
self.layout.addWidget(self.table_view)

# Aplicar estilo CSS para personalização adicional
self.table_view.setStyleSheet("""
    QTableView {
        background-color: #F3F3F3; /* Fundo principal */
        color: #333333; /* Cor do texto */
        gridline-color: #CCCCCC; /* Cor das linhas da grade */
        alternate-background-color: #FFFFFF; /* Fundo alternado */
        selection-background-color: #E0E0E0; /* Fundo ao selecionar */
        selection-color: #000000; /* Cor do texto ao selecionar */
        border: 1px solid #CCCCCC; /* Borda ao redor da tabela */
        font-size: 14px;
    }
    QTableView::item:selected {
        background-color: #E0E0E0; /* Fundo ao selecionar item */
        color: #000000; /* Cor do texto do item selecionado */
    }
    QTableView::item {
        border: 1px solid transparent; /* Borda invisível por padrão */
        padding: 5px;
    }
    QHeaderView::section {
        background-color: #D6D6D6; /* Fundo do cabeçalho */
        color: #333333; /* Cor do texto do cabeçalho */
        font-weight: bold;
        font-size: 14px;
        padding: 4px;
        border: 1px solid #CCCCCC; /* Borda entre as seções */
    }
""")

# Configurar o modelo SQL para visualização
self.configurarSqlModelSignal.emit()

```