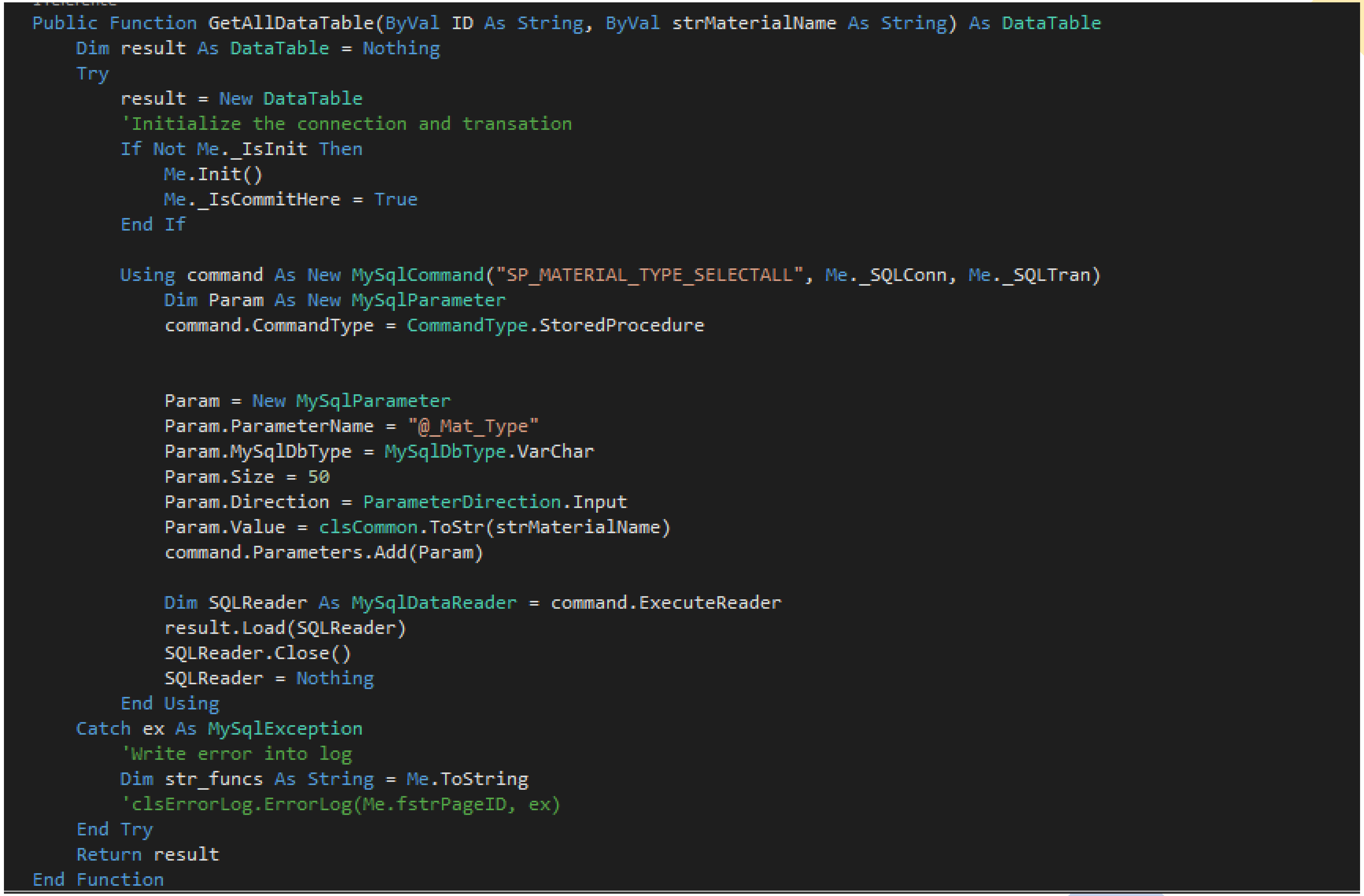
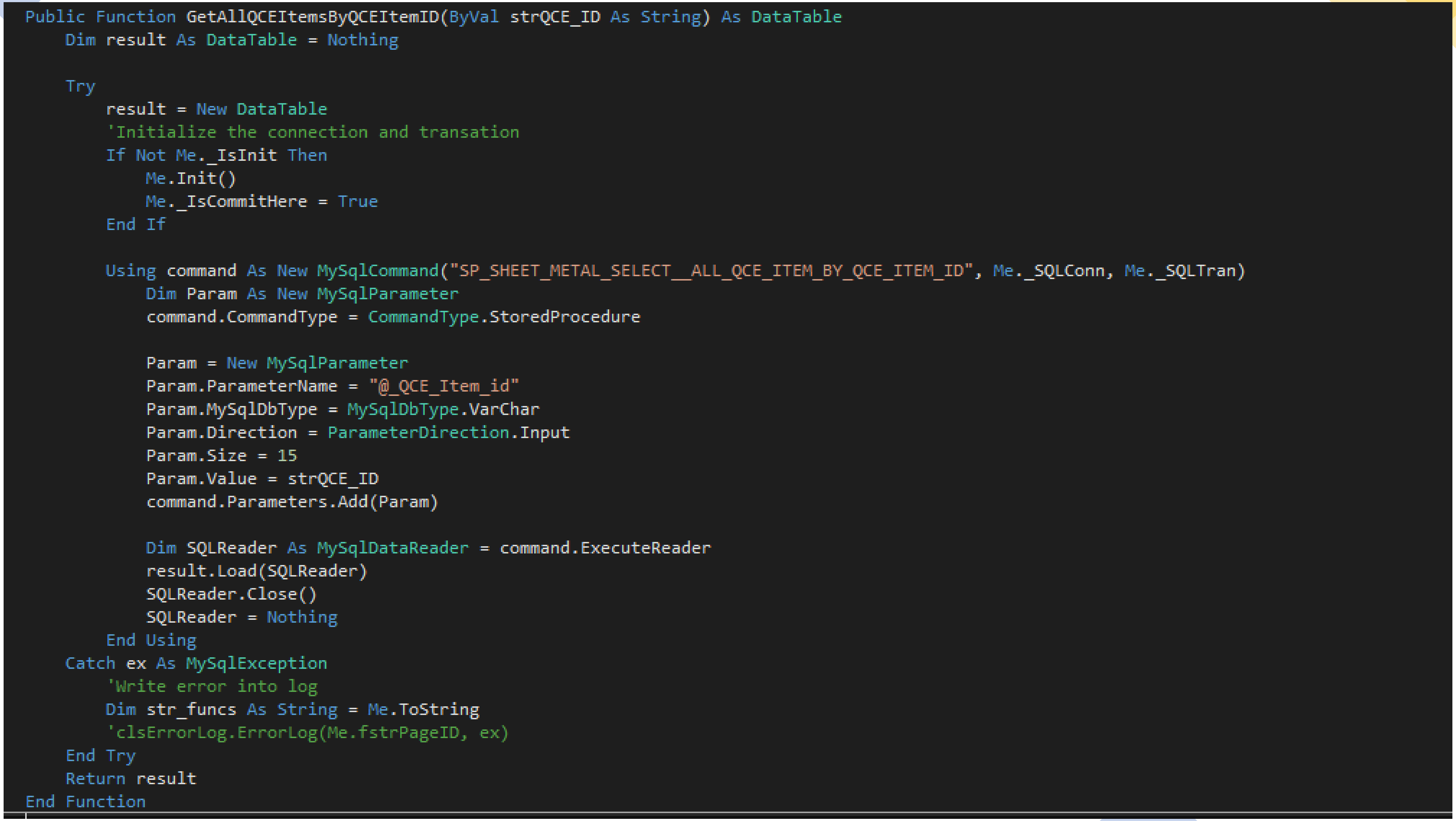
All Material Type class



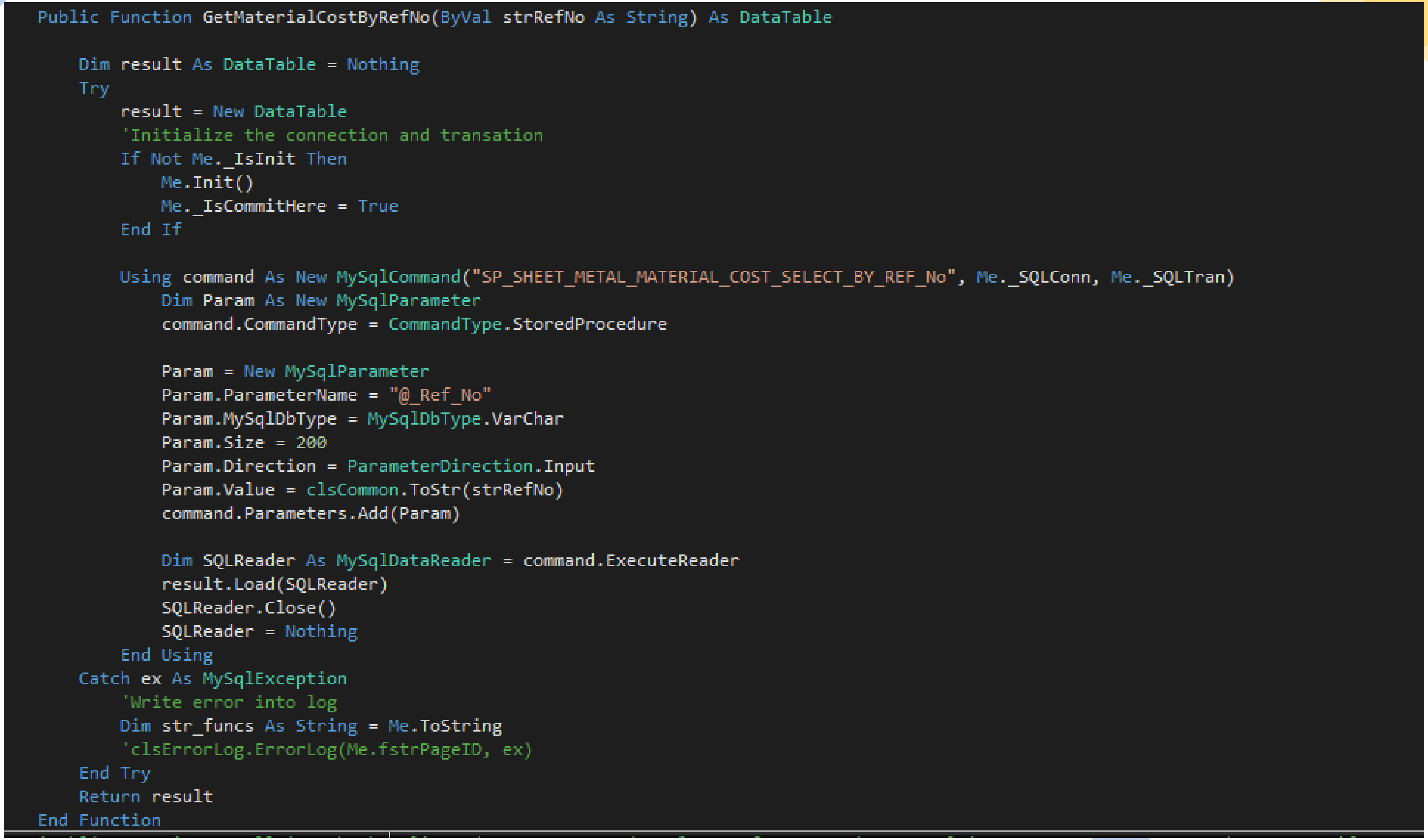
```
Public Function GetAllDataTable(ByVal ID As String, ByVal strMaterialName As String) As DataTable
    Dim result As DataTable = Nothing
    Try
        result = New DataTable
        'Initialize the connection and transation
        If Not Me._IsInit Then
            Me.Init()
            Me._IsCommitHere = True
        End If

        Using command As New MySqlCommand("SP_MATERIAL_TYPE_SELECTALL", Me._SQLConn, Me._SQLTran)
            Dim Param As New MySqlParameter
            command.CommandType = CommandType.StoredProcedure


            Param = New MySqlParameter
            Param.ParameterName = "@_Mat_Type"
            Param.MySqlDbType = MySqlDbType.VarChar
            Param.Size = 50
            Param.Direction = ParameterDirection.Input
            Param.Value = clsCommon.ToStr(strMaterialName)
            command.Parameters.Add(Param)

            Dim SQLReader As MySqlDataReader = command.ExecuteReader
            result.Load(SQLReader)
            SQLReader.Close()
            SQLReader = Nothing
        End Using
    Catch ex As MySqlException
        'Write error into log
        Dim str_funcs As String = Me.ToString
        'clsErrorLog.ErrorLog(Me.fstrPageID, ex)
    End Try
    Return result
End Function
```

Part Number Class



```
Public Function GetAllQCEItemsByQCEItemID(ByVal strQCE_ID As String) As DataTable
    Dim result As DataTable = Nothing

    Try
        result = New DataTable
        'Initialize the connection and transation
        If Not Me._IsInit Then
            Me.Init()
            Me._IsCommitHere = True
        End If

        Using command As New MySqlCommand("SP_SHEET_METAL_SELECT__ALL_QCE_ITEM_BY_QCE_ITEM_ID", Me._SQLConn, Me._SQLTran)
            Dim Param As New MySqlParameter
            command.CommandType = CommandType.StoredProcedure

            Param = New MySqlParameter
            Param.ParameterName = "@_QCE_Item_id"
            Param.MySqlDbType = MySqlDbType.VarChar
            Param.Direction = ParameterDirection.Input
            Param.Size = 15
            Param.Value = strQCE_ID
            command.Parameters.Add(Param)

            Dim SQLReader As MySqlDataReader = command.ExecuteReader
            result.Load(SQLReader)
            SQLReader.Close()
            SQLReader = Nothing
        End Using
    Catch ex As MySqlException
        'Write error into log
        Dim str_funcs As String = Me.ToString
        'clsErrorLog.ErrorLog(Me.fstrPageID, ex)
    End Try
    Return result
End Function
```

Thickness, and other material data Class



```
Public Function GetMaterialCostByRefNo(ByVal strRefNo As String) As DataTable

    Dim result As DataTable = Nothing
    Try
        result = New DataTable
        'Initialize the connection and transation
        If Not Me._IsInit Then
            Me.Init()
            Me._IsCommitHere = True
        End If

        Using command As New MySqlCommand("SP_SHEET_METAL_MATERIAL_COST_SELECT_BY_REF_No", Me._SQLConn, Me._SQLTran)
            Dim Param As New MySqlParameter
            command.CommandType = CommandType.StoredProcedure

            Param = New MySqlParameter
            Param.ParameterName = "@_Ref_No"
            Param.MySqlDbType = MySqlDbType.VarChar
            Param.Size = 200
            Param.Direction = ParameterDirection.Input
            Param.Value = clsCommon.ToStr(strRefNo)
            command.Parameters.Add(Param)

            Dim SQLReader As MySqlDataReader = command.ExecuteReader
            result.Load(SQLReader)
            SQLReader.Close()
            SQLReader = Nothing
        End Using
    Catch ex As MySqlException
        'Write error into log
        Dim str_funcs As String = Me.ToString
        'clsErrorLog.ErrorLog(Me.fstrPageID, ex)
    End Try
    Return result
End Function
```