設定1

FrmMenu.exe.config(app.config)(クライアント)

A

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <system.net>
    <connectionManagement>
      <add address = "*" maxconnection = "900"/>
    </connectionManagement>
  </system.net>
  <system.diagnostics>
    <sources>
      <source name="DefaultSource" switchName="DefaultSwitch">
        <listeners>
          <add name="FileLog" />
        </listeners>
      </source>
    </sources>
    <switches>
      <add name="DefaultSwitch" value="Information" />
    </switches>
    <sharedListeners>
      <add name="FileLog"
        type="Microsoft.VisualBasic.Logging.FileLogTraceListener,
        Microsoft.VisualBasic,
        Version=8.0.0.0,
        Culture=neutral,
        PublicKeyToken=b03f5f7f11d50a3a,
        processorArchitecture=MSIL"
        initializeData="FileLogWriter" />
    </sharedListeners>
  </system.diagnostics>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.8" />
  </startup>
  <system.serviceModel>
    <behaviors>
      <serviceBehaviors>
        <behavior name="ServiceBehavior">
          <serviceThrottling maxConcurrentCalls="600" maxConcurrentSessions="1200"
            maxConcurrentInstances="1000" />
        </behavior>
      </serviceBehaviors>
    </behaviors>
    <bindings>
      <basicHttpBinding>
        <binding
          name="BasicHttpBinding_IService1"
          maxReceivedMessageSize="2147483647"
          maxBufferSize="2147483647"
          maxBufferPoolSize="2147483647"
          sendTimeout="00:10:00"
          receiveTimeout="00:10:00">
          <security mode="Transport">
            <transport clientCredentialType="None" />
          </security>
        </binding>
      </basicHttpBinding>
    </bindings>
    <client>
      <endpoint
      address="https://loadtesting.gohantancs.jp/GTXMcfService/Service.svc"
      binding="basicHttpBinding"
      bindingConfiguration="BasicHttpBinding_IService1"
      contract="ServiceReference1.IService"
      name="BasicHttpBinding_IService1" />
    </client>
  </system.serviceModel>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="System.Runtime.CompilerServices.Unsafe" publicKeyToken="b03f5f7f11d50a3a" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Memory" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-4.0.1.2" newVersion="4.0.1.2" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Buffers" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-4.0.3.0" newVersion="4.0.3.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Threading.Tasks.Extensions" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-4.2.0.1" newVersion="4.2.0.1" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Threading.Channels" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-7.0.0.0" newVersion="7.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="Microsoft.Bcl.AsyncInterfaces" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-7.0.0.0" newVersion="7.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Text.Json" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-7.0.0.0" newVersion="7.0.0.0" />
      </dependentAssembly>
    </assemblyBinding>
  </runtime>
</configuration>
```

web.config(サーバー)

B

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
<!--
  <system.diagnostics>
    <sources>
      <source name="System.ServiceModel.MessageLogging">
        <listeners>
          <add name="messages"
               type="System.Diagnostics.XmlWriterTraceListener"
               initializeData="c:¥Logs¥messages.svclog" />
        </listeners>
      </source>
    </sources>
  </system.diagnostics>
-->
  <system.net>
    <connectionManagement>
      <add address = "*" maxconnection = "384" />
    </connectionManagement>
  </system.net>
  <system.serviceModel>
<!--
    <diagnostics performanceCounters="All" wmiProviderEnabled="True">
      <messageLogging logEntireMessage="True"
                      logMalformedMessages="True"
                      logMessagesAtServiceLevel="True"
                      logMessagesAtTransportLevel="True" />
    </diagnostics>
    <diagnostics performanceCounters="All">
      <messageLogging
        logEntireMessage="true"
        logMalformedMessages="false"
        logMessagesAtServiceLevel="true"
        logMessagesAtTransportLevel="false"
        maxMessagesToLog="3000"
        maxSizeOfMessageToLog="2000" />
    </diagnostics>
-->
    <serviceHostingEnvironment aspNetCompatibilityEnabled="True"
      multipleSiteBindingsEnabled="True" />
    <services>
      <service behaviorConfiguration="ServiceBehavior" name="GTXMcfService.Service">
        <endpoint address="" binding="basicHttpBinding" contract="GTXMcfService.IService">
          <identity>
            <dns value="localhost" />
          </identity>
        </endpoint>
        <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange" />
      </service>
    </services>
    <behaviors>
      <endpointBehaviors>
        <behavior name="NewBehavior0">
<!--
          <dispatcherSynchronization asynchronousSendEnabled="true" maxPendingReceives="1000" />
-->
          <dispatcherSynchronization asynchronousSendEnabled="true" />
        </behavior>
      </endpointBehaviors>
      <serviceBehaviors>
        <behavior name="ServiceBehavior">
          <serviceThrottling maxConcurrentCalls="600" maxConcurrentSessions="1200"
            maxConcurrentInstances="3000" />
          <serviceMetadata httpGetEnabled="true" />
          <serviceDebug includeExceptionDetailInFaults="false" />
        </behavior>
        <behavior name="">
          <serviceMetadata httpGetEnabled="true" httpsGetEnabled="true" />
          <serviceDebug includeExceptionDetailInFaults="false" />
        </behavior>
      </serviceBehaviors>
    </behaviors>
    <bindings>
      <basicHttpBinding>
        <binding maxReceivedMessageSize="2147483647"
                 maxBufferSize="2147483647"
                 maxBufferPoolSize="2147483647"
                 sendTimeout="00:10:00"
                 receiveTimeout="00:10:00">
        </binding>
      </basicHttpBinding>
    </bindings>
  </system.serviceModel>
  <system.web>
    <compilation debug="true" strict="false" explicit="true" targetFramework="4.8" />
    <httpRuntime targetFramework="4.8" maxRequestLength="153600" executionTimeout="600" />
    <customErrors mode="Off" />
  </system.web>
  <system.webServer>
    <security>
      <requestFiltering>
        <requestLimits maxAllowedContentLength="1073741824" />
      </requestFiltering>
    </security>
  </system.webServer>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="System.Runtime.CompilerServices.Unsafe" publicKeyToken="b03f5f7f11d50a3a" culture=
        <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Memory" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-4.0.1.2" newVersion="4.0.1.2" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Buffers" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-4.0.3.0" newVersion="4.0.3.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Threading.Tasks.Extensions" publicKeyToken="cc7b13ffcd2ddd51" culture="neut
        <bindingRedirect oldVersion="0.0.0.0-4.2.0.1" newVersion="4.2.0.1" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Threading.Channels" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-7.0.0.0" newVersion="7.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="Microsoft.Bcl.AsyncInterfaces" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral"
        <bindingRedirect oldVersion="0.0.0.0-7.0.0.0" newVersion="7.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Text.Json" publicKeyToken="cc7b13ffcd2ddd51" culture="neutral" />
        <bindingRedirect oldVersion="0.0.0.0-7.0.0.0" newVersion="7.0.0.0" />
      </dependentAssembly>
    </assemblyBinding>
  </runtime>
</configuration>
```

ClsWrapper.vb(クライアント/サーバー共通処理)

C

```
Option Strict On

Imports ClsDataFormat
Imports CMN = ClsCommon.ClsCom
Imports CST = ClsCommon.ClsCst
Imports ClsDbAcs
Imports System.Collections.Specialized


Public Class ClsWrapper
    Inherits MarshalByRefObject

    '''' ************************************************************************************
    '''' Private メンバ変数
    '''' ************************************************************************************
    Public m_wcf As ServiceReference1.IServiceClient
    Public m_strEndpointAddress As String = ""  'WCFサーバURL
    Public m_recKSys As Rec_KSys 'システム情報
    Private m_recXErr As Rec_XErr 'エラー情報
    Private m_intProgressValue As Integer
    Private m_intProgressValueMax As Integer
    Private m_objOperationContextScope As System.ServiceModel.OperationContextScope

    '''' ************************************************************************************
    Public Sub New(intLogLevel As Integer,
                   blnWcfMode As Boolean,
                   strLogInUserID As String,
                   strLogInJiCd As String,
                   strLogInSchemaName As String,
                   intLogInHeiStDiv As Integer,
                   strLogInComputerName As String,
                   intAuthority As Integer,
                   intCkSkDiv As Integer,
                   intHachuAutoDiv As Integer,
                   intHachuAutoSetDiv As String,
                   intMobileDiv As Integer,
                   intFdGp As Integer)

        MyBase.New()
        m_recKSys = New Rec_KSys
        m_recXErr = New Rec_XErr
        m_recKSys.WcfMode = blnWcfMode
        If blnWcfMode Then
            m_wcf = New ServiceReference1.IServiceClient
            m_strEndpointAddress = m_wcf.Endpoint.Address.Uri.OriginalString
            m_objOperationContextScope = New System.ServiceModel.OperationContextScope(CType(m_wcf.InnerChannel, System.ServiceModel.ICliientChannel))
            With System.ServiceModel.Web.WebOperationContext.Current.OutgoingRequest.Headers
                .Add("LogLevel", CStr(intLogLevel))
                .Add("LogInUserID", strLogInUserID)
                .Add("LogInJiCd", strLogInJiCd)
                .Add("LogInSchemaName", strLogInSchemaName)
                .Add("LogInHeiStDiv", CStr(intLogInHeiStDiv))
                .Add("LogInComputerName", strLogInComputerName)
                .Add("Authority", CStr(intAuthority))
                .Add("CkSkDiv", CStr(intCkSkDiv))
                .Add("HachuAutoDiv", CStr(intHachuAutoDiv))
                .Add("HachuAutoSetDiv", strHachuAutoSetDiv)
                .Add("MobileDiv", CStr(intMobileDiv))
                .Add("FdGp", CStr(intFdGp))
            End With                                                    クライアント側で実行されるコード
        Else
            Dim blnError As Boolean = False
            If System.ServiceModel.Web.WebOperationContext.Current IsNot Nothing Then
                Try
                    With System.ServiceModel.Web.WebOperationContext.Current.IncomingRequest
                        m_recKSys.LogLevel = If(.Headers("LogLevel") = String.Empty, intLogLevel, CInt(.Headers("LogLevel")))
                        m_recKSys.LogInUserID = If(.Headers("LogInUserID") = String.Empty, strLogInUserID, .Headers("LogInUserID"))
                        m_recKSys.LogInJiCd = If(.Headers("LogInJiCd") = String.Empty, strLogInJiCd, .Headers("LogInJiCd"))
                        m_recKSys.LogInSchemaName = If(.Headers("LogInSchemaName") = String.Empty, strLogInSchemaName, .Headers("LogInSchemaName"
                        m_recKSys.LogInHeiStDiv = If(.Headers("LogInHeiStDiv") = String.Empty, intLogInHeiStDiv, CInt(.Headers("LogInHeiStDiv")))
                        m_recKSys.LogInComputerName = If(.Headers("LogInComputerName") = String.Empty, strLogInComputerName, .Headers("LogInCompu
                        m_recKSys.Authority = If(.Headers("Authority") = String.Empty, intAuthority, CInt(.Headers("Authority")))
                        m_recKSys.CkSkDiv = If(.Headers("CkSkDiv") = String.Empty, intCkSkDiv, CInt(.Headers("CkSkDiv")))
                        m_recKSys.HachuAutoDiv = If(.Headers("HachuAutoDiv") = String.Empty, intHachuAutoDiv, CInt(.Headers("HachuAutoDiv")))
                        m_recKSys.HachuAutoSetDiv = If(.Headers("HachuAutoSetDiv") = String.Empty, strHachuAutoSetDiv, .Headers("HachuAutoSetD
                        m_recKSys.MobileDiv = If(.Headers("MobileDiv") = String.Empty, intMobileDiv, CInt(.Headers("MobileDiv")))
                        m_recKSys.FdGp = If(.Headers("FdGp") = String.Empty, intFdGp, CInt(.Headers("FdGp")))
                    End With
                Catch ex As Exception
                    blnError = True
                End Try
            End If

            If blnError Or System.ServiceModel.Web.WebOperationContext.Current Is Nothing Then
                m_recKSys.LogLevel = intLogLevel
                m_recKSys.LogInUserID = strLogInUserID
                m_recKSys.LogInJiCd = strLogInJiCd
                m_recKSys.LogInSchemaName = strLogInSchemaName
                m_recKSys.LogInHeiStDiv = intLogInHeiStDiv
                m_recKSys.LogInComputerName = strLogInComputerName
                m_recKSys.Authority = intAuthority
                m_recKSys.CkSkDiv = intCkSkDiv
                m_recKSys.HachuAutoDiv = intHachuAutoDiv
                m_recKSys.HachuAutoSetDiv = strHachuAutoSetDiv
                m_recKSys.MobileDiv = intMobileDiv
                m_recKSys.FdGp = intFdGp
            End If                                                      サーバー側で実行されるコード
        End If
    End Sub
```

{サンプルコード：GetTKonCkFd}

Service.svc(Service.svc.vb)(サーバー)

D

```
Option Strict On
Imports ClsDataFormat
Imports CMN = ClsCommon.ClsCom
Imports CST = ClsCommon.ClsCst
Imports ClsDbAcs
Imports System.Collections.Specialized
Imports ClsWrapper


Imports System.Threading
Imports System.Threading.Tasks

<System.ServiceModel.ServiceBehavior(ConcurrencyMode:=ConcurrencyMode.Single,
InstanceContextMode:=InstanceContextMode.PerSession,
UseSynchronizationContext:=False)>
Public Class Service
    Implements IService

    Private WRP As ClsWrapper.ClsWrapper
    Public Rec_KSys As ClsDataFormat.Rec_KSys
    Public strDateTime As String = Format(Now(), "yyyy-MM-dd HH:mm:ss.fff")

    Sub New()
        WRP = New ClsWrapper.ClsWrapper(0, False, "", "", "", 0, "", 0, 0, 0, "", 0, 0)

        'm_recKSys.LogLevel
        'm_recKSys.LogInUserID
        'm_recKSys.LogInJiCd
        'm_recKSys.LogInSchemaName
        'm_recKSys.LogInHeiStDiv
        'm_recKSys.LogInComputerName
        'm_recKSys.Authority
        'm_recKSys.CkSkDiv
        'm_recKSys.HachuAutoDiv
        'm_recKSys.HachuAutoSetDive
        'm_recKSys.MobileDiv
        'm_recKSys.FdGp
        Try
            Dim thread As Thread = Thread.CurrentThread
            Dim LogFile As String = "c:¥App¥service¥" & Format(Now(), "yyyyMMddhhmmssfff") & ".txt"
            Using swWriter As New System.IO.StreamWriter(LogFile, True)
                swWriter.WriteLine(Format(Now(), "yyyy/MM/dd HH:mm:ss.fff") &
                                   Control.Chars.Tab & strDateTime &
                                   Control.Chars.Tab & WRP.m_recKSys.LogInUserID &
                                   Control.Chars.Tab & WRP.m_recKSys.LogInComputerName &
                                   Control.Chars.Tab & " GTXMcfServiceを開始します。")

                swWriter.Close()
            End Using
        Catch ex As Exception
        End Try

    End Sub
    Protected Overrides Sub Finalize()
        Try
            Dim thread As Thread = Thread.CurrentThread
            Dim LogFile As String = "c:¥App¥service¥" & Format(Now(), "yyyyMMddhhmmssfff") & ".txt"
            Using swWriter As New System.IO.StreamWriter(LogFile, True)
                swWriter.WriteLine(Format(Now(), "yyyy/MM/dd HH:mm:ss.fff") &
                                   Control.Chars.Tab & strDateTime &
                                   Control.Chars.Tab & WRP.m_recKSys.LogInUserID &
                                   Control.Chars.Tab & WRP.m_recKSys.LogInComputerName &
                                   Control.Chars.Tab & " GTXMcfServiceを終了します。")

                swWriter.Close()
            End Using
        Catch ex As Exception
        End Try

        MyBase.Finalize()
    End Sub
    Public Function LifeCheck() As Boolean Implements IService.LifeCheck
        Return WRP.LifeCheck()
    End Function
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
    Public Function GetTKonCkFd(ByVal strYMD As String _
                              , ByVal intEatTmDiv As Integer _
                              , ByVal strMealCd As String _
                              , ByVal emmSelEtDiv As CST.emmSelEtDiv _
                              , ByVal blnGetTkonMk As Boolean _
                              , Optional ByVal intNutriLevel As Integer = 0 _
                              , Optional ByVal ColMNameNt As Col_MName() = Nothing) As Rec_TKonMk
        Return WRP.GetTKonCkFd(strYMD, intEatTmDiv, strMealCd, emmSelEtDiv, blnGetTkonMk, intNutriLevel, ColMNameNt)
    End Function
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
                                                                        :
End Class
```

E

```
Public Function GetTKonCkFd(ByVal strYMD As String _
                          , ByVal intEatTmDiv As Integer _
                          , ByVal strMealCd As String _
                          , ByVal emmSelEtDiv As CST.emmSelEtDiv _
                          , ByVal blnGetTkonMk As Boolean _
                          , Optional ByVal intNutriLevel As Integer = 0 _
                          , Optional ByVal ColMNameNt As Col_MName() = Nothing) As Rec_TKonMk
    Try
        If m_recKSys.WcfMode Then
            With System.ServiceModel.Web.WebOperationContext.Current.OutgoingRequest.Headers
                .Remove("LogInComputerName")
                .Add("LogInComputerName", System.Reflection.MethodBase.GetCurrentMethod().Name)
            End With
            Return m_wcf.GetTKonCkFd(strYMD, intEatTmDiv, strMealCd, emmSelEtDiv, blnGetTkonMk, intNutriLevel, ColMNameNt)
                                                                    クライアント側で実行されるコード
        Else
            Dim objCTLKon As New CtlKon(m_recKSys, m_recXErr)
            Return objCTLKon.GetTKonCkFd(strYMD, intEatTmDiv, strMealCd, emmSelEtDiv, blnGetTkonMk, intNutriLevel, ColMNameNt)
        End If                                                          サーバー側で実行されるコード
    Catch ex As Exception
        Return Nothing
    End Try
End Function
```

Public Function GetTKonCkFd(サーバー)

```
Public Function GetTKonCkFd(ByVal strYMD As String
                          , ByVal intEatTmDiv As Integer _
                          , ByVal strMealCd As String _
                          , ByVal emmSelEtDiv As CST.emmSelEtDiv _
                          , ByVal blnGetTkonMk As Boolean _
                          , Optional ByVal intNutriLevel As Integer = 0 _
                          , Optional ByVal ColMNameNt As Col_MName() = Nothing) As Rec_TKonMk

    Me.LogOut(CST.emmLogType.MethodST, System.Reflection.MethodBase.GetCurrentMethod().Name, True)
    Dim objDB As New DBTKon(Me.SysInf, Me.ErrInf)
    Dim objDBGrp As New DBTKonGrpInf(Me.SysInf, Me.ErrInf)

    '<--- LoadTesting 2024/01/11 --->
    Dim p As System.Diagnostics.Process = System.Diagnostics.Process.GetCurrentProcess()
    Dim strHTst_GetHostName As String = System.Net.Dns.GetHostName() & "(" & CStr(p.Id) & ")"
    Me.LogOut(Cls1Common.ClsCst.emmLogType.Information, Me.SysInf.LoginUserID & ":" & strHTst_GetHostName & ":Start[loadtesting]")
    '<--- LoadTesting 2024/01/11 --->

    Try
        objDB.DBConnect()
        objDBGrp.DBConnect(objDB)

        Dim objRecTkonMk As New Rec_TKonMk

        If blnGetTkonMk = True Then
            objDB.SelectMode = DBTKon.emmSelectMode.Status
            objDB.KonDiv = 0
            objDB.CtlYMDSt = strYMD
            objDB.CtlYMDEd = strYMD
            objDB.EatTmDiv = intEatTmDiv
            objDB.colGetKeyInf = New Col_XItmLst
            objDB.colGetKeyInf.Add(strMealCd, CType(emmSelEtDiv, Integer), Nothing)

            objDB.Select(objDB.GetSelectString())
            Dim objColTKonMonth As Col_TKonMonth = CType(objDB.GetData(), Col_TKonMonth)
                                                                        :
                                                                        :
        Return objRecTkonMk

    Catch ex As Exception
        Me.LogOut(ex)
        Return Nothing
    Finally
        If Not objDBGrp Is Nothing Then
            objDBGrp.DBClose()
            objDBGrp.DBDisconnect()
            objDBGrp = Nothing
        End If
        If Not objDB Is Nothing Then
            objDB.DBClose()
            objDB.DBDisconnect()
            objDB = Nothing
        End If
        '<--- LoadTesting 2024/01/11 --->
        If Win(Me.SysInf.LoginUserID, 1, 4) = "load" Then
            Me.LogOut(Cls1Common.ClsCst.emmLogType.Information, Me.SysInf.LoginUserID & ":" & strHTst_GetHostName & ":End[loadtesting]")
        End If
        '<--- LoadTesting 2024/01/11 --->
        Me.LogOut(CST.emmLogType.MethodED, System.Reflection.MethodBase.GetCurrentMethod().Name, True)
    End Try
End Function
```