接口：

/// <summary>

/// 获取未回合同扣款主表信息

/// </summary>

/// <returns></returns>

[OperationContract]

DataTable GetDeductNoReturn(DeductNoReturnSelectModel model);

/// <summary>

/// 获取未回合同扣款计算期间

/// </summary>

/// <returns></returns>

[OperationContract]

DataTable GetCalDate();

/// <summary>

/// 未回合同扣款添加期间

/// </summary>

/// <param name="CalDate">期间</param>

/// <param name="AIDate">AI截至日期</param>

/// <param name="PbDate">发布截至日期</param>

/// <param name="UserId"></param>

/// <returns></returns>

[OperationContract]

bool AddDeductNoReturnCalDate(DateTime CalDate, DateTime AIDate, DateTime PbDate, string UserId);

/// <summary>

/// 删除计算期间

/// </summary>

/// <param name="PId">期间ID</param>

/// <returns></returns>

[OperationContract]

string DeleteDeductNoReturnCalDate(int PId, string UserId);

/// <summary>

/// 计算扣款基数

/// </summary>

/// <param name="PId"></param>

/// <param name="UserId"></param>

[OperationContract]

string CalcluDeductNoReturn(int PId, string UserId);

/// <summary>

/// 当前合同号扣款基数从新计算

/// </summary>

/// <param name="PId"></param>

/// <param name="UserId"></param>

[OperationContract]

string CalcluDeductNoReturnR(int PId, string UserId,string ContractNos);

/// <summary>

/// 获取未回合同扣款详情

/// </summary>

/// <param name="DNRId"></param>

/// <returns></returns>

[OperationContract]

DataTable GetDeductNoReturnDetail(string DNRId);

/// <summary>

/// 添加或修改未回合同扣款详情信息进行金额调整

/// </summary>

/// <param name="model"></param>

/// <param name="IDo">0 添加 1 修改</param>

/// <returns></returns>

[OperationContract]

DataTable AddorEditDeductNoReturn(DeductNoReturnDetailsModel model,int IDo);

/// <summary>

/// 获取计算期间COMBOX数据源头

/// </summary>

/// <returns></returns>

[OperationContract]

IList<QueryDictionaryModel> GetCalDateList();

/// <summary>

/// 删除未回合同详情

/// </summary>

/// <param name="DeleteId"></param>

[OperationContract]

void DeletdNoReturnDetails(int DeleteId);

/// <summary>

/// 修改合同号扣款规则

/// </summary>

/// <param name="ContractNo"></param>

/// <param name="DeductType"></param>

/// <param name="dateBegin"></param>

/// <param name="dateEnd"></param>

[OperationContract]

string addDeductRule(string ContractNo, int DeductType, DateTime dateBegin, DateTime dateEnd,string UserId, int FileId, string FileName,string SalesVPId,string Remark);

/// <summary>

/// 获取规则修改历史

/// </summary>

/// <param name="ContractNo"></param>

/// <returns></returns>

[OperationContract]

DataTable GetRuleHostory(string ContractNo);

/// <summary>

/// 获取设置规则列表

/// </summary>

/// <param name="model"></param>

/// <returns></returns>

[OperationContract]

DataTable GetDeductRuleList(DeductNoReturnSelectModel model);

/// <summary>

/// 删除已设置的扣款规则

/// </summary>

/// <param name="MContractNo"></param>

[OperationContract]

void DeleteDeductNoReturnRule(string CId);

/// <summary>

/// 未回合同扣款审核

/// </summary>

/// <param name="PId">期间ID</param>

/// <param name="userId">使用人</param>

/// <param name="iDo">0 审核 1 复审 9 反审</param>

[OperationContract]

string DecudtNoReturnAudit(int PId,string userId,int iDo);

/// <summary>

/// 获取主合同回款信息详情

/// </summary>

/// <param name="Mcontractno"></param>

/// <returns></returns>

[OperationContract]

DataTable GetRefundDetails(DeductNoReturnSelectModel model);

/// <summary>

/// 获取退款历史

/// </summary>

/// <param name="Mcontractno"></param>

/// <returns></returns>

[OperationContract]

DataTable GetRefundHistoryDetails(DeductNoReturnSelectModel model);

/// <summary>

/// 获取子合同回款详情

/// </summary>

/// <param name="McontractNo"></param>

/// <returns></returns>

[OperationContract]

DataTable GetRefundDetail(string McontractNo, string SalesId);

/// <summary>

/// 添加未回合同扣款退款信息

/// </summary>

/// <param name="McontractNo"></param>

/// <returns></returns>

[OperationContract]

void addRefundBackDetails(DeductReturnBack model);

/// <summary>

/// 获取该合同暂扣的总额

/// </summary>

/// <param name="ContractNo"></param>

/// <returns></returns>

[OperationContract]

double GetTotalDeductAmount(string ContractNo);

/// <summary>

/// 审核未回合同扣款退款

/// </summary>

/// <param name="State"></param>

[OperationContract]

string AuditReturnRefund(int State,string RIds,string UserId);

/// <summary>

/// 未回合同扣款退款根据销售员汇总

/// </summary>

/// <param name="date">查询月份</param>

[OperationContract]

DataTable GetRefundHistoryDetailsTotal(DateTime date);

/// <summary>

/// 根据销售员汇总当月的扣款金额

/// </summary>

/// <param name="PId"></param>

/// <param name="DataCenter">数据中心</param>

/// <returns></returns>

[OperationContract]

DataTable GetDeductNoReturnTotal(int PId,string DataCenter);

/// <summary>

/// 删除扣款信息

/// </summary>

/// <param name="RId"></param>

[OperationContract]

string DeleteRefund(string RId);

/// <summary>

/// 不考虑计算期间获取未回合同扣款的主表信息

/// </summary>

/// <param name="model"></param>

/// <returns></returns>

[OperationContract]

DataTable GetDeductNoReturnAll(DeductNoReturnSelectModel model);

/// <summary>

/// 获取未回合同扣款排除部门

/// </summary>

/// <returns></returns>

[OperationContract]

DataTable GetDedcutNotReturnExclude(string DeptId);

/// <summary>

/// 根据ID获取未回合同扣款排除部门

/// </summary>

/// <returns></returns>

[OperationContract]

DataTable GetDedcutNotReturnExcludeById(string EId);

/// <summary>

/// 删除排除部门

/// </summary>

/// <returns></returns>

[OperationContract]

string DelDedcutNotReturnExclude(int RId);

/// <summary>

/// 添加和编辑排除部门

/// </summary>

/// <param name="RId">排除部门表ID</param>

/// <param name="iDo">0 添加 1 修改</param>

/// <param name="DeptId">部门id</param>

/// <param name="Datebegin">开始时间</param>

/// <param name="DateEnd">结束时间</param>

/// <param name="UserId">操作人</param>

/// <returns></returns>

[OperationContract]

string EditDedcutNotReturnExclude(int RId,int iDo,string DeptId,DateTime Datebegin,DateTime DateEnd,string UserId);

/// <summary>

/// 读取扣款规则文件列表

/// </summary>

/// <param name="CId"></param>

/// <returns></returns>

[OperationContract]

DataTable GetDedcutNotReturnConfigFile(string CId);

/// <summary>

/// 删除扣款规则审核文件

/// </summary>

/// <param name="FId"></param>

/// <returns></returns>

[OperationContract]

DataTable DelDedcutNotReturnConfigFile(string FId);

/// <summary>

/// 上传扣款规则审核文件

/// </summary>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <returns></returns>

[OperationContract]

string AddDedcutNotReturnConfigFile(string CId,int FileId,string FileName,string UserId);

/// <summary>

/// 根据主合同号批量上传扣款规则审核文件

/// </summary>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <returns></returns>

[OperationContract]

string BatchAddDedcutNotReturnConfigFile(string McontractNos, int FileId, string FileName, string UserId);

/// <summary>

/// 获取发布转移合同

/// </summary>

/// <param name=""></param>

/// <returns></returns>

[OperationContract]

DataTable GetDeductNotReturnPubRela(string MContractNo,string OContractNo,string PId);

/// <summary>

/// 发布转移合同调整

/// </summary>

/// <param name="McontractNo"></param>

/// <param name="datas"></param>

/// <returns></returns>

[OperationContract]

string EditDeductNotReturnPubRela(string McontractNo,string PId, List<EditDeductNotReturnPubRelaModel> Modellist,string UserId);

/// <summary>

/// 修改VO经理

/// </summary>

/// <param name="CId"></param>

/// <param name="SalesVPId"></param>

/// <returns></returns>

[OperationContract]

string EditSalesVPId(string CId, string SalesVPId);

/// <summary>

///获取发布转移列表

/// </summary>

/// <param name="ContractNos"></param>

/// <returns></returns>

[OperationContract]

DataTable GetPubTran(string ContractNos);

/// <summary>

///未回合同扣款根据主合同号获取销售员

/// </summary>

/// <param name="ContractNos"></param>

/// <returns></returns>

[OperationContract]

string GetDeductNotReturnSalesId(string ContractNo);

2 服务：

/// <summary>

/// 获取未回合同扣款主表信息

/// </summary>

/// <returns></returns>

public DataTable GetDeductNoReturn(DeductNoReturnSelectModel model)

{

string sqlwhere = string.Empty;

if (model.SalesIsnull)

sqlwhere = "AND isnull(SALESID,'')=''";

string sqlwhere1 = string.Empty;

if (model.IsDeDuctZero)

sqlwhere1 = "having sum(DeductAmount) <>0";

DataTable dt = new DataTable();

string sql = string.Format(@"select A.DNRId,A.PId,A.MContractNo,A.BaseAmount,A.AState,A.CreateOn,OP2.OperatorName CreateBy,

A.FAuditOn,OP3.OperatorName FAuditBy,A.SAuditOn,OP4.OperatorName SAuditBy,

CASE WHEN ISNULL(SC.VindicatorId,'')='' THEN CASE WHEN OP.IsStop=1 THEN '' ELSE SC.SignerId END ELSE SC.VindicatorId END SalesId,CASE WHEN ISNULL(OPS.OperatorName,'')='' THEN CASE WHEN OP.IsStop=1 THEN '' ELSE OP.OperatorName END ELSE OPS.OperatorName END OperatorName,

CASE WHEN ISNULL(dps.DeptName,'') ='' THEN CASE WHEN OP.IsStop=1 THEN '' ELSE dp.DeptName END ELSE dps.DeptName END DeptName,

B.DeductAmount,SC.CSignDate,SC.SignDeptId,SA.SignAName,BR.Brand,CT.CustomerName,

CASE WHEN SC.ContLevel IN(1,3,8) THEN '全国单'ELSE '二线单'END ContLevelS,

case when A.AState=1 then '待审核' when A.AState=2 then '审核' when A.AState=3 then '审核确认' end AStateName,

DataCenterCity = Case When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C001.A01.D009' Then '上海数据中心'

When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C002.A01.D009' Then '北京数据中心'

Else '' End

into #temp\_noreturn

from FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.MContractNo=SC.ContractNo

LEFT JOIN PR\_tb\_e\_Operator OP ON SC.SignerId =OP.OperatorId

left join PR\_tb\_b\_Department DP on op.DeptId=dp.DeptId

LEFT JOIN PR\_tb\_e\_Operator OPS ON SC.VindicatorId =OPS.OperatorId

left join PR\_tb\_b\_Department DPs on ops.DeptId=dps.DeptId

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

LEFT JOIN PR\_tb\_e\_Operator OP2 ON A.CreateBy=OP2.OperatorId

LEFT JOIN PR\_tb\_e\_Operator OP3 ON A.FAuditBy=OP3.OperatorId

LEFT JOIN PR\_tb\_e\_Operator OP4 ON A.SAuditBy=OP4.OperatorId

LEFT JOIN FSO\_tb\_b\_SignAName SA ON SC.SignAName=SA.Code

LEFT JOIN FSO\_tb\_e\_Customer CT ON SC.SignCustId=CT.CustomerId

LEFT JOIN FSO\_tb\_b\_Brand BR ON ISNULL(SC.Brand,0)=BR.Code

left join (select DB.City,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(City,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by City,OP.DeptId)AC ON SC.SignCity=AC.City

LEFT JOIN(select DB.DeptId Dept,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(DB.DeptId,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by DB.DeptId,OP.DeptId)AD on CHARINDEX(AD.Dept,SC.SignDeptId) = 1

where (MContractNo like '%{0}%' or CHARINDEX(MContractNo,'{0}') > 0)

and isnull(CASE WHEN ISNULL(SC.VindicatorId,'')=''THEN CASE WHEN OP.IsStop=1 THEN '' ELSE SC.SignerId END ELSE SC.VindicatorId END,'') like '%{1}%'

and ISNULL(SA.SignAName,'') like '%{5}%'

and ISNULL(BR.Brand,'') like '%{6}%'

and ISNULL(CT.CustomerName,'') like '%{7}%'

and A.PId>='{2}' and A.PId<='{4}'

and a.MContractNo not in (select MContractNo from FSO\_tb\_p\_DedcutNotReturnConfig where RuleType=4)

--找出已转移的合同的前合同号

select \*

INTO #TEMP\_ROCONTRACTNO

from (select CASE WHEN ISNULL(SC1.MainContractNo,'')='' THEN SC1.ContractNo ELSE SC1.MainContractNo END OContractNo,

CASE WHEN ISNULL(SC2.MainContractNo,'')='' THEN SC2.ContractNo ELSE SC2.MainContractNo END RContractNo

from FSO\_tb\_e\_SalesPublishRelation A

LEFT JOIN FSO\_tb\_e\_SalesContract SC1 ON A.OContractNo=SC1.ContractNo

LEFT JOIN FSO\_tb\_e\_SalesContract SC2 ON A.RContractNo=SC2.ContractNo

where OFromType=1 and RFromType=1

union

Select CASE WHEN ISNULL(SC1.MainContractNo,'')='' THEN SC1.ContractNo ELSE SC1.MainContractNo END OContractNo,

CASE WHEN ISNULL(SC2.MainContractNo,'')='' THEN SC2.ContractNo ELSE SC2.MainContractNo END RContractNo

From Desc\_tb\_PublishRelation pr

inner join FSO\_tb\_e\_SalesPublish sp1 on sp1.DescPublishId = pr.OPublishId

inner join FSO\_tb\_e\_SalesPublish sp2 on sp2.DescPublishId = pr.RPublishId

inner join FSO\_tb\_e\_SalesContract sc1 on sp1.contractId = sc1.ContractId

inner join FSO\_tb\_e\_SalesContract sc2 on sp2.contractId = sc2.ContractId

where pr.RType ='PbSame') A

group by OContractNo,RContractNo

--最终查询

SELECT A.DNRId,A.PId,A.MContractNo,A.BaseAmount,A.AState,A.CreateOn,A.CreateBy,

A.FAuditOn,FAuditBy,A.SAuditOn,SAuditBy,SignAName,Brand,CustomerName,

SalesId,OperatorName,DeptName,sum(DeductAmount) DeductAmount,CSignDate,SignDeptId,

ContLevelS,AStateName,DataCenterCity ,B.isfiled,C.DeductAmountAll,CASE WHEN C.DeductAmountAll=0 THEN 0 ELSE 1 END ISSPAMOUNT,CONVERT(VARCHAR(10),YEAR(fdate)) +'-'+CONVERT(VARCHAR(10),MONTH(fdate)) fdate,T.OContractNo

from #temp\_noreturn A

LEFT JOIN (select A.MContractNo,case when MAX(isnull(C.FileId,0))>0 then 1 else 0 end isfiled from #temp\_noreturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnConfig B ON A.MContractNo=B.MContractNo

left join FSO\_tb\_p\_DedcutNotReturnConfigFile C ON B.CId=C.CId

GROUP BY A.MContractNo)B

ON A.MContractNo=B.MContractNo

LEFT JOIN (select A.MContractNo,abs(SUM(ISNULL(B.DeductAmount,0))) DeductAmountAll from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B on A.DNRId=B.DNRId AND B.DeductType=2

WHERE A.PId<='{4}'

group by A.MContractNo)C ON A.MContractNo=C.MContractNo

left join FSO\_tb\_p\_DedcutNotReturnPeriod DRP ON A.PID=DRP.PID

left join (SELECT A.RContractNo,

STUFF((SELECT ','+ISNULL(B.OContractNo,'') FROM #TEMP\_ROCONTRACTNO B WHERE A.RContractNo = B.RContractNo FOR XML PATH('')),1,1,'') AS OContractNo

FROM #TEMP\_ROCONTRACTNO A

GROUP BY A.RContractNo

)T on a.MContractNo=t.RContractNo

where DataCenterCity like '%{3}%' {8}

GROUP BY A.DNRId,A.PId,A.MContractNo,A.BaseAmount,A.AState,A.CreateOn,A.CreateBy,

A.FAuditOn,FAuditBy,A.SAuditOn,SAuditBy,

SalesId,OperatorName,DeptName,CSignDate,SignDeptId,

ContLevelS,AStateName,DataCenterCity,B.isfiled,C.DeductAmountAll,FDate,SignAName,Brand,CustomerName,T.OContractNo

{9}

order by a.pid

drop table #temp\_noreturn

drop table #TEMP\_ROCONTRACTNO

", model.MContractNo,model.SalesId,model.PId,model.DataCenter,model.EndPId,model.SignAName,model.Brand,model.CustomerName, sqlwhere,sqlwhere1);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DedcutNotReturn";

return dt;

}

/// <summary>

/// 获取未回合同扣款计算期间

/// </summary>

/// <returns></returns>

public DataTable GetCalDate()

{

DataTable dt = new DataTable();

string sql = string.Format(@"select PId,CONVERT(VARCHAR(10),YEAR(FDATE)) +'-'+CONVERT(VARCHAR(10),MONTH(FDATE)) FDATE, CONVERT(VARCHAR(10),YEAR(PBTODATE))+'-'+CONVERT(VARCHAR(10),MONTH(PBTODATE)) PBTODATE,CONVERT(VARCHAR(10),YEAR(AITODATE))+'-'+CONVERT(VARCHAR(10),MONTH(AITODATE)) AITODATE,OP.OperatorName CREATEBYNAME,A.CREATEON,A.CREATEBY

from FSO\_tb\_p\_DedcutNotReturnPeriod A

LEFT JOIN PR\_tb\_e\_Operator OP ON A.CREATEBY=OP.OperatorId

order by PId desc");

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "CalDate";

return dt;

}

/// <summary>

/// 未回合同扣款添加期间

/// </summary>

/// <param name="CalDate">期间</param>

/// <param name="AIDate">AI截至日期</param>

/// <param name="PbDate">发布截至日期</param>

/// <param name="UserId"></param>

/// <returns></returns>

public bool AddDeductNoReturnCalDate(DateTime CalDate, DateTime AIDate, DateTime PbDate, string UserId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"insert into FSO\_tb\_p\_DedcutNotReturnPeriod(FDate,PbToDate,AIToDate,CreateOn,CreateBy)

values('{0}','{1}','{2}',GETDATE(),'{3}')", CalDate, PbDate, AIDate, UserId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "AddCalDate";

return true;

}

/// <summary>

/// 删除计算期间

/// </summary>

/// <param name="PId">期间ID</param>

/// <returns></returns>

public string DeleteDeductNoReturnCalDate(int PId, string UserId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

if (select count(\*) from FSO\_TB\_P\_DEDCUTNOTRETURN where PId='{0}')>0

begin

if (select top 1 Astate from FSO\_TB\_P\_DEDCUTNOTRETURN where PId='{0}')='1'

begin

delete from FSO\_tb\_p\_DedcutNotReturnDetail where DNRId IN (SELECT DNRId FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}')

delete from FSO\_TB\_P\_DEDCUTNOTRETURN where PId='{0}'

delete from FSO\_tb\_p\_DedcutNotReturnPeriod where PId='{0}'

delete from FSO\_tb\_p\_DedcutNotReturnContract where PId='{0}'

select '删除成功'

end

else

select '该期间已经审核,请先反审才可删除期间'

end

else

begin

delete from FSO\_tb\_p\_DedcutNotReturnPeriod where PId='{0}'

select '删除成功'

end

", PId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "AddCalDate";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 计算扣款基数

/// </summary>

/// <param name="PId"></param>

/// <param name="UserId"></param>

public string CalcluDeductNoReturn(int PId, string UserId)

{

string sql = string.Format(@"

select ROW\_NUMBER() over(order by pid desc) rowsn,PId,FDate,DATEADD(MM,DATEDIFF(MM,0,fdate),0) begindate,DATEADD(MM,DATEDIFF(MM,0,fdate)+1,0) enddate

INTO #TEMP\_PID

from FSO\_tb\_p\_DedcutNotReturnPeriod where PId<='{0}' order by PId desc

DECLARE @PBYEAR int

DECLARE @PBMONTH int

DECLARE @AIYEAR int

DECLARE @AIMONTH int

DECLARE @FDATE DATETIME

select @FDATE=fdate,@PBYEAR=YEAR(fdate),@PBMONTH=MONTH(fdate)-1,@AIYEAR=YEAR(fdate),@AIMONTH=MONTH(fdate) from FSO\_tb\_p\_DedcutNotReturnPeriod where PId='{0}'

if @PBMONTH=0

SELECT @PBMONTH=12,@PBYEAR=@PBYEAR-1

--获取该AI期间下的所有AI数据

SELECT ContractId,SUM(Amount) Amount

INTO #TEMP\_VoucherAI

FROM FSO\_tb\_e\_VoucherAI

WHERE ((FYear=@AIYEAR AND FPeriod<=@AIMONTH) or (FYear<@AIYEAR)) and ShareFlag = 0 and LapFlag = 0

GROUP BY ContractId

--获取该PB期间下的所有发布额数据

SELECT ContractId,SUM(Amount) Amount

INTO #TEMP\_VoucherPB

FROM FSO\_tb\_e\_VoucherAI\_A

WHERE ((FYear=@PBYEAR AND FPeriod<=@PBMONTH) or (FYear<@PBYEAR)) and ShareFlag = 0 and LapFlag = 0

GROUP BY ContractId

--回合同计算期限临时表

select CONTRACTNO

into #temp\_contreturn

from (select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,MIN(SP.BeginDate) PbBeginDate

from FSO\_tb\_e\_SalesContract SC

left join FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

where SC.TCompanyNo IN (100,101,103) AND SignCustId NOT IN (SELECT CustomerId FROM FSO\_tb\_b\_internal) and (ContReturnId not in (1,3) or (ContReturnId in (1,3) and ContReturnDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0)))

GROUP BY CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END )BB

GROUP BY CONTRACTNO

HAVING MIN(PbBeginDate)<DATEADD(MM,DATEDIFF(MM,0,@FDATE),0)

--回确认计算期间临时表

select CONTRACTNO

into #temp\_confirm

from (select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,SP.BeginDate

from FSO\_tb\_e\_SalesContract SC

LEFT JOIN FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

WHERE SP.ConfirmState=0 and SC.TCompanyNo IN (100,101,103) AND SignCustId NOT IN (SELECT CustomerId FROM FSO\_tb\_b\_internal)

)CC

GROUP BY CONTRACTNO

HAVING MIN(BeginDate)<DATEADD(MM,DATEDIFF(MM,0,@FDATE),0)

--将子合同扣款基数表插入临时表

select A.\* INTO #TEMP\_BASEAMOUNT

from

(SELECT A.ContractId,A.Amount-ISNULL(B.Amount,0) AMOUNT,CASE WHEN ISNULL(SC.VindicatorId,'')='' THEN CASE WHEN OP.IsStop=1 THEN '' ELSE SC.SignerId END ELSE SC.VindicatorId END SignerId,SC.CSignDate,SC.SignDeptId,CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO

FROM #TEMP\_VoucherPB A

LEFT JOIN #TEMP\_VoucherAI B ON A.ContractId=B.ContractId

INNER JOIN FSO\_tb\_e\_SalesContract SC ON A.ContractId=SC.ContractId AND SC.TCompanyNo IN (100,101,103) AND SignCustId NOT IN (SELECT CustomerId FROM FSO\_tb\_b\_internal)

LEFT JOIN PR\_tb\_e\_Operator op on SC.SignerId=OP.OperatorId

)A

INNER join #temp\_contreturn B

ON A.CONTRACTNO=B.CONTRACTNO

where a.ContractId in (select ContractId from FSO\_tb\_e\_SalesContract SC where SC.ContReturnId not in (1,3) or (SC.ContReturnId in (1,3) and sc.ContReturnDate>=(select enddate from #TEMP\_PID where PId='{0}')))

and (a.ContractId in (select SP.ContractId from FSO\_tb\_e\_SalesContract SC

inner JOIN FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

WHERE (ConfirmState =0 AND BeginDate<DATEADD(MM,DATEDIFF(MM,0,@FDATE),0)) OR (ConfirmState !=0 AND ConfirmDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0))

GROUP BY SP.ContractId) OR

A.CONTRACTNO IN (select CONTRACTNO from #temp\_confirm))

drop table #temp\_contreturn

drop table #temp\_confirm

--删除排除部门

DELETE FROM #TEMP\_BASEAMOUNT

WHERE ContractId IN

(

SELECT ContractId FROM #TEMP\_BASEAMOUNT A

INNER join FSO\_tb\_p\_DedcutNotReturnExclude DRE ON DRE.DeptId=A.SignDeptId AND A.CSignDate>=DRE.BeginDate AND A.CSignDate<=DRE.EndDate

)

--删除西安部门非A9,A10主体的

DELETE FROM #TEMP\_BASEAMOUNT

WHERE ContractId IN

(

SELECT A.ContractId FROM #TEMP\_BASEAMOUNT A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.ContractId=SC.ContractId

LEFT JOIN FSO\_tb\_b\_SignAName SA ON SC.SignAName=SA.Code

WHERE SA.SignAName NOT IN ('A9','A10') AND SC.SignDeptId IN (select DeptId from PR\_tb\_b\_Department where DeptName like '%西安%' and NotSalesDept=0)

)

--删除4A客户

delete from #TEMP\_BASEAMOUNT where ContractId IN

(select ContractId

from FSO\_tb\_e\_SalesContract sc

inner join FSO\_tb\_b\_Agent A ON SC.SignCustId=A.CustomerId and sc.SignDate between a.BeginDate and a.EndDate)

--将子合同扣款基数及信息插入子合同详情表

delete from FSO\_tb\_p\_DedcutNotReturnContract where pid='{0}'

insert into FSO\_tb\_p\_DedcutNotReturnContract(PId,mcontractno,contractid,baseamount,updateon)

select '{0}',CONTRACTNO,ContractId,AMOUNT,getdate() from #TEMP\_BASEAMOUNT

IF NOT EXISTS(SELECT PID FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PId='{0}' AND AState>1)

BEGIN

delete from FSO\_tb\_p\_DedcutNotReturnDetail where DNRId IN (SELECT DNRId FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}')

delete from FSO\_TB\_P\_DEDCUTNOTRETURN where PId='{0}'

INSERT INTO FSO\_TB\_P\_DEDCUTNOTRETURN(PId,MContractNo,BaseAmount,AState,CreateOn,CreateBy)

SELECT '{0}',CONTRACTNO,SUM(AMOUNT) AMOUNTS,'1',GETDATE(),'{1}'

FROM #TEMP\_BASEAMOUNT

GROUP BY CONTRACTNO

HAVING SUM(AMOUNT)>100

INSERT INTO FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

SELECT DNRId,1,C.SignerId,

CASE WHEN SC.ContLevel IN(1,3,8) THEN

CASE WHEN D.ProAmount>1000000 AND (SELECT COUNT(\*) FROM FSO\_TB\_P\_DEDCUTNOTRETURN AC WHERE AC.MContractNo=A.MContractNo AND A.PId IN (SELECT TOP 4 PID FROM #TEMP\_PID))<4 THEN 0

ELSE 1 END

ELSE CASE WHEN (SELECT COUNT(\*) FROM FSO\_TB\_P\_DEDCUTNOTRETURN AC WHERE AC.MContractNo=A.MContractNo AND A.PId IN (SELECT TOP 4 PID FROM #TEMP\_PID))=4 THEN 1

ELSE 0 END END DeductType,

CASE WHEN A.BaseAmount<500000 THEN 500 WHEN 500000<=A.BaseAmount AND A.BaseAmount<1000000 THEN 1000 WHEN 1000000<=A.BaseAmount AND A.BaseAmount<5000000 THEN 2000 ELSE 5000 END DeductAmount ,GETDATE(),'{1}'

FROM FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN (SELECT CONTRACTNO,SignerId FROM #TEMP\_BASEAMOUNT GROUP BY CONTRACTNO,SignerId) C ON A.MContractNo=C.CONTRACTNO

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON SC.ContractNo=A.MContractNo

left join (select sum(CASE WHEN ISNULL(C.TotalAmount,0)>0 THEN TotalAmount ELSE ProAmount END) ProAmount,CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO

from FSO\_tb\_e\_SalesContract SC

LEFT JOIN FSO\_tb\_e\_AgreeRelateContract B ON SC.ContractId=B.ContractId

LEFT JOIN FSO\_tb\_e\_SalesAgreeContract C ON B.SalesAgreeId=C.Id

GROUP BY CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END) D ON A.MContractNo=D.ContractNo

WHERE A.PId='{0}'

IF EXISTS(select CONTRACTNO from

(select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,ContReturnId,ContReturnDate

from FSO\_tb\_e\_SalesContract SC)A

where ContReturnId in (1,3) and ContReturnDate>=(select begindate from #TEMP\_PID where rowsn=4) and ContReturnDate<(select enddate from #TEMP\_PID where rowsn=1) AND CONTRACTNO IN (SELECT MContractNo FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}'))

BEGIN

--将计算期间中还未回款合同状态更新为暂扣

SELECT DNRID

into #temp\_DeContractNo

FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}' AND MContractNo IN

(select CONTRACTNO from

(select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,ContReturnId,ContReturnDate

from FSO\_tb\_e\_SalesContract SC)A

where ContReturnId in (1,3) and ContReturnDate>=(select begindate from #TEMP\_PID where rowsn=4) and ContReturnDate<(select enddate from #TEMP\_PID where rowsn=1) AND CONTRACTNO IN (SELECT MContractNo FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}')

)

update FSO\_tb\_p\_DedcutNotReturnDetail set DeductType=0 where DeductType=1 and DNRID IN (select DNRId from #temp\_DeContractNo)

drop table #temp\_DeContractNo

END

INSERT INTO FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

SELECT DNRID,ISINPUT,SALESID,MAX(RULETYPE),MIN(DEDUCTAMOUNT),GETDATE() CREATEON,CREATEBY FROM

(SELECT b.DNRId,1 ISINPUT,c.SalesId,a.RuleType,

case when (a.RuleType=2 or a.RuleType=3) then

case when d.FDate >= DATEADD(MM,DATEDIFF(MM,0,a.BeginDate),0) and d.FDate < DATEADD(MM,DATEDIFF(MM,0,a.EndDate)+1,0) then (0-MAX(c.DeductAmount))

else 0 end

else (0-MAX(c.DeductAmount)) end DEDUCTAMOUNT

,'{1}' CREATEBY

FROM FSO\_tb\_p\_DedcutNotReturn B

inner join FSO\_tb\_p\_DedcutNotReturnConfig A on a.MContractNo=b.MContractNo

left join FSO\_tb\_p\_DedcutNotReturnDetail C on b.DNRId=c.DNRId

left join FSO\_tb\_p\_DedcutNotReturnPeriod D ON B.PId=D.PId

WHERE B.PId='{0}' and c.IsInput=1

group by b.DNRId,c.SalesId,a.BeginDate,a.EndDate,d.FDate,a.RuleType)A

group by DNRId,ISINPUT,SALESID,CREATEBY

declare @PIdB int

select @PIdB=PID FROM #TEMP\_PID WHERE rowsn=4

--将特批不扣过期部分的应扣款插入扣款详情表

insert into FSO\_tb\_p\_DedcutNotReturnDetail (DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

select A.DNRId,0,Info.SalesId,1,abs(SUM(INFO.DeductAmount)) DeductAmount,GETDATE(),'{1}'

from (SELECT A.SalesId,A.DeductAmount DeductAmount,MContractNo

FROM(

select A.DNRId,A.PId,A.MContractNo,CASE WHEN ISNULL(DRC.BeginDate,'')='' THEN B.DeductAmount ELSE B.DeductAmount\*0.7 END DeductAmount,DRC.BeginDate,DRC.EndDate,B.SalesId

from FSO\_tb\_p\_DedcutNotReturn A

left join #TEMP\_PID P ON A.PId=P.PId

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join FSO\_tb\_p\_DedcutNotReturnConfig DRC ON A.MContractNo=DRC.MContractNo and RuleType=2 AND DATEADD(MM,DATEDIFF(MM,0,DRC.BeginDate),0)>=P.begindate AND DATEADD(MM,DATEDIFF(MM,0,DRC.EndDate)+1,0)<=P.enddate

WHERE A.PId<'{0}' AND B.DeductType=2 AND A.PID>=@PIdB

GROUP BY A.DNRId,A.PId,A.MContractNo,DRC.BeginDate,DRC.EndDate,B.SalesId,B.DeductAmount

UNION

select A.DNRId,A.PId,A.MContractNo,CASE WHEN ISNULL(DRC.BeginDate,'')='' THEN 0 ELSE B.DeductAmount\*0.3 END DeductAmount,DRC.BeginDate,DRC.EndDate,DRC.SalesVPId SalesId

from FSO\_tb\_p\_DedcutNotReturn A

left join #TEMP\_PID P ON A.PId=P.PId

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join FSO\_tb\_p\_DedcutNotReturnConfig DRC ON A.MContractNo=DRC.MContractNo and RuleType=2 AND DATEADD(MM,DATEDIFF(MM,0,DRC.BeginDate),0)>=P.begindate AND DATEADD(MM,DATEDIFF(MM,0,DRC.EndDate)+1,0)<=P.enddate

WHERE A.PId<'{0}' AND B.DeductType=2 AND A.PID>=@PIdB

GROUP BY A.DNRId,A.PId,A.MContractNo,DRC.BeginDate,DRC.EndDate,DRC.SalesVPId,B.DeductAmount

)A

GROUP BY PId,MContractNo,DeductAmount,SalesId,MContractNo

)Info

left join FSO\_tb\_p\_DedcutNotReturn A ON Info.MContractNo=A.MContractNo

WHERE A.PId='{0}' AND A.DNRID IN (select MIN(A.DNRId) DNRId from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

where B.DeductType=1

GROUP BY MContractNo)

GROUP BY DNRId,SalesId

ORDER BY DNRId

SELECT '计算成功'

DROP TABLE #TEMP\_BASEAMOUNT

END

ELSE

begin

SELECT '该期间已经进行过计算'

DROP TABLE #TEMP\_BASEAMOUNT

end

drop table #TEMP\_VoucherAI

drop table #TEMP\_VoucherPB

DROP TABLE #TEMP\_PID

", PId,UserId);

DataSet ds = new DataSet();

IDbConnection db = GetDbConnection();

SqlConnection conn = new SqlConnection(db.ConnectionString);

if (conn != null)

{

conn.Open();

SqlCommand cmd = new SqlCommand();

cmd.CommandTimeout = 60 \* 10;

cmd.Connection = conn;

cmd.CommandText = sql;

SqlDataAdapter da = new SqlDataAdapter(cmd);

da.Fill(ds);

conn.Close();

}

DataTable dt = ds.Tables[0];

dt.TableName = "CalcluDeductNoReturn";

string Result = dt.Rows[0][0].ToString();

return Result;

}

/// <summary>

/// 计算当前合同号扣款基数

/// </summary>

/// <param name="PId"></param>

/// <param name="UserId"></param>

public string CalcluDeductNoReturnR(int PId, string UserId,string ContractNos)

{

string sql = string.Format(@"

select ROW\_NUMBER() over(order by pid desc) rowsn,PId,FDate,DATEADD(MM,DATEDIFF(MM,0,fdate),0) begindate,DATEADD(MM,DATEDIFF(MM,0,fdate)+1,0) enddate

INTO #TEMP\_PID

from FSO\_tb\_p\_DedcutNotReturnPeriod where PId<='{0}' order by PId desc

DECLARE @PBYEAR int

DECLARE @PBMONTH int

DECLARE @AIYEAR int

DECLARE @AIMONTH int

DECLARE @FDATE DATETIME

select @FDATE=fdate,@PBYEAR=YEAR(fdate),@PBMONTH=MONTH(fdate)-1,@AIYEAR=YEAR(fdate),@AIMONTH=MONTH(fdate) from FSO\_tb\_p\_DedcutNotReturnPeriod where PId='{0}'

if @PBMONTH=0

SELECT @PBMONTH=12,@PBYEAR=@PBYEAR-1

--获取该AI期间下的所有AI数据

SELECT A.ContractId,SUM(A.Amount) Amount

INTO #TEMP\_VoucherAI

FROM FSO\_tb\_e\_VoucherAI A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.ContractId=SC.ContractId

WHERE ((FYear=@AIYEAR AND FPeriod<=@AIMONTH) or (FYear<@AIYEAR)) and ShareFlag = 0 and LapFlag = 0 AND ((SC.CONTRACTNO like '%{2}%' or CHARINDEX(SC.CONTRACTNO,'{2}')>0) OR ( ISNULL(SC.MainContractNo,'') like '%{2}%' or CHARINDEX( ISNULL(SC.MainContractNo,''),'{2}')>0))

GROUP BY A.ContractId

--获取该PB期间下的所有发布额数据

SELECT A.ContractId,SUM(A.Amount) Amount

INTO #TEMP\_VoucherPB

FROM FSO\_tb\_e\_VoucherAI\_A A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.ContractId=SC.ContractId

WHERE ((FYear=@PBYEAR AND FPeriod<=@PBMONTH) or (FYear<@PBYEAR)) and ShareFlag = 0 and LapFlag = 0 AND ((SC.CONTRACTNO like '%{2}%' or CHARINDEX(SC.CONTRACTNO,'{2}')>0) OR ( ISNULL(SC.MainContractNo,'') like '%{2}%' or CHARINDEX( ISNULL(SC.MainContractNo,''),'{2}')>0))

GROUP BY A.ContractId

--回合同计算期限临时表

select CONTRACTNO

into #temp\_contreturn

from (select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,MIN(SP.BeginDate) PbBeginDate

from FSO\_tb\_e\_SalesContract SC

left join FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

where SC.TCompanyNo IN (100,101,103) AND SignCustId NOT IN (SELECT CustomerId FROM FSO\_tb\_b\_internal) and (ContReturnId not in (1,3) or (ContReturnId in (1,3) and ContReturnDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0)))

GROUP BY CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END )BB

GROUP BY CONTRACTNO

HAVING MIN(PbBeginDate)<DATEADD(MM,DATEDIFF(MM,0,@FDATE),0)

--回确认计算期间临时表

select CONTRACTNO

into #temp\_confirm

from (select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,SP.BeginDate

from FSO\_tb\_e\_SalesContract SC

LEFT JOIN FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

WHERE SP.ConfirmState=0 AND SC.TCompanyNo IN (100,101,103) AND SignCustId NOT IN (SELECT CustomerId FROM FSO\_tb\_b\_internal)

)CC

GROUP BY CONTRACTNO

HAVING MIN(BeginDate)<DATEADD(MM,DATEDIFF(MM,0,@FDATE),0)

--将子合同扣款基数表插入临时表

select A.\* INTO #TEMP\_BASEAMOUNT

from

(SELECT A.ContractId,A.Amount-ISNULL(B.Amount,0) AMOUNT,CASE WHEN ISNULL(SC.VindicatorId,'')='' THEN CASE WHEN OP.IsStop=1 THEN '' ELSE SC.SignerId END ELSE SC.VindicatorId END SignerId,SC.CSignDate,SC.SignDeptId,CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO

FROM #TEMP\_VoucherPB A

LEFT JOIN #TEMP\_VoucherAI B ON A.ContractId=B.ContractId

INNER JOIN FSO\_tb\_e\_SalesContract SC ON A.ContractId=SC.ContractId AND SC.TCompanyNo IN (100,101,103) AND SignCustId NOT IN (SELECT CustomerId FROM FSO\_tb\_b\_internal)

LEFT JOIN PR\_tb\_e\_Operator op on SC.SignerId=OP.OperatorId

)A

INNER join #temp\_contreturn B

ON A.CONTRACTNO=B.CONTRACTNO

where a.ContractId in (select ContractId from FSO\_tb\_e\_SalesContract SC where SC.ContReturnId not in (1,3) or (SC.ContReturnId in (1,3) and sc.ContReturnDate>=(select enddate from #TEMP\_PID where PId='{0}')))

and (a.ContractId in (select SP.ContractId from FSO\_tb\_e\_SalesContract SC

inner JOIN FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

WHERE (ConfirmState =0 AND BeginDate<DATEADD(MM,DATEDIFF(MM,0,@FDATE),0)) OR (ConfirmState !=0 AND ConfirmDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0))

GROUP BY SP.ContractId) OR

A.CONTRACTNO IN (select CONTRACTNO from #temp\_confirm))

AND (A.CONTRACTNO like '%{2}%' or CHARINDEX(A.CONTRACTNO,'{2}') > 0)

drop table #temp\_contreturn

drop table #temp\_confirm

--删除排除部门

DELETE FROM #TEMP\_BASEAMOUNT

WHERE ContractId IN

(

SELECT ContractId FROM #TEMP\_BASEAMOUNT A

INNER join FSO\_tb\_p\_DedcutNotReturnExclude DRE ON DRE.DeptId=A.SignDeptId AND A.CSignDate>=DRE.BeginDate AND A.CSignDate<=DRE.EndDate

)

--删除西安部门非A9,A10主体的

DELETE FROM #TEMP\_BASEAMOUNT

WHERE ContractId IN

(

SELECT A.ContractId FROM #TEMP\_BASEAMOUNT A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.ContractId=SC.ContractId

LEFT JOIN FSO\_tb\_b\_SignAName SA ON SC.SignAName=SA.Code

WHERE SA.SignAName NOT IN ('A9','A10') AND SC.SignDeptId IN (select DeptId from PR\_tb\_b\_Department where DeptName like '%西安%' and NotSalesDept=0)

)

--删除4A客户

delete from #TEMP\_BASEAMOUNT where ContractId IN

(select ContractId

from FSO\_tb\_e\_SalesContract sc

inner join FSO\_tb\_b\_Agent A ON SC.SignCustId=A.CustomerId and sc.SignDate between a.BeginDate and a.EndDate)

--将子合同扣款基数及信息插入子合同详情表

delete from FSO\_tb\_p\_DedcutNotReturnContract where pid='{0}' AND (MContractNo like '%{2}%' or CHARINDEX(MContractNo,'{2}') > 0)

insert into FSO\_tb\_p\_DedcutNotReturnContract(PId,mcontractno,contractid,baseamount,updateon)

select '{0}',CONTRACTNO,ContractId,AMOUNT,getdate() from #TEMP\_BASEAMOUNT

IF NOT EXISTS(SELECT PID FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PId='{0}' AND AState>1)

BEGIN

delete from FSO\_tb\_p\_DedcutNotReturnDetail where DNRId IN (SELECT DNRId FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}' AND (MContractNo like '%{2}%' or CHARINDEX(MContractNo,'{2}') > 0))

delete from FSO\_TB\_P\_DEDCUTNOTRETURN where PId='{0}' AND (MContractNo like '%{2}%' or CHARINDEX(MContractNo,'{2}') > 0)

INSERT INTO FSO\_TB\_P\_DEDCUTNOTRETURN(PId,MContractNo,BaseAmount,AState,CreateOn,CreateBy)

SELECT '{0}',CONTRACTNO,SUM(AMOUNT) AMOUNTS,'1',GETDATE(),'{1}'

FROM #TEMP\_BASEAMOUNT

GROUP BY CONTRACTNO

HAVING SUM(AMOUNT)>100

INSERT INTO FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

SELECT DNRId,1,C.SignerId,

CASE WHEN SC.ContLevel IN(1,3,8) THEN

CASE WHEN D.ProAmount>1000000 AND (SELECT COUNT(\*) FROM FSO\_TB\_P\_DEDCUTNOTRETURN AC WHERE AC.MContractNo=A.MContractNo AND A.PId IN (SELECT TOP 4 PID FROM #TEMP\_PID))<4 THEN 0

ELSE 1 END

ELSE CASE WHEN (SELECT COUNT(\*) FROM FSO\_TB\_P\_DEDCUTNOTRETURN AC WHERE AC.MContractNo=A.MContractNo AND A.PId IN (SELECT TOP 4 PID FROM #TEMP\_PID))=4 THEN 1

ELSE 0 END END DeductType,

CASE WHEN A.BaseAmount<500000 THEN 500 WHEN 500000<=A.BaseAmount AND A.BaseAmount<1000000 THEN 1000 WHEN 1000000<=A.BaseAmount AND A.BaseAmount<5000000 THEN 2000 ELSE 5000 END DeductAmount ,GETDATE(),'{1}'

FROM FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN (SELECT CONTRACTNO,SignerId FROM #TEMP\_BASEAMOUNT GROUP BY CONTRACTNO,SignerId) C ON A.MContractNo=C.CONTRACTNO

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON SC.ContractNo=A.MContractNo

left join (select sum(CASE WHEN ISNULL(C.TotalAmount,0)>0 THEN TotalAmount ELSE ProAmount END) ProAmount,CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO

from FSO\_tb\_e\_SalesContract SC

LEFT JOIN FSO\_tb\_e\_AgreeRelateContract B ON SC.ContractId=B.ContractId

LEFT JOIN FSO\_tb\_e\_SalesAgreeContract C ON B.SalesAgreeId=C.Id

GROUP BY CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END) D ON A.MContractNo=D.ContractNo

WHERE A.PId='{0}' AND (A.MContractNo like '%{2}%' or CHARINDEX(A.MContractNo,'{2}') > 0)

IF EXISTS(select CONTRACTNO from

(select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,ContReturnId,ContReturnDate

from FSO\_tb\_e\_SalesContract SC)A

where ContReturnId in (1,3) and ContReturnDate>=(select begindate from #TEMP\_PID where rowsn=4) and ContReturnDate<(select enddate from #TEMP\_PID where rowsn=1) AND CONTRACTNO IN (SELECT MContractNo FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}' AND (MContractNo like '%{2}%' or CHARINDEX(MContractNo,'{2}') > 0)))

BEGIN

--将计算期间中还未回款合同状态更新为暂扣

SELECT DNRID

into #temp\_DeContractNo

FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}' AND MContractNo IN

(select CONTRACTNO from

(select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO,ContReturnId,ContReturnDate

from FSO\_tb\_e\_SalesContract SC)A

where ContReturnId in (1,3) and ContReturnDate>=(select begindate from #TEMP\_PID where rowsn=4) and ContReturnDate<(select enddate from #TEMP\_PID where rowsn=1) AND CONTRACTNO IN (SELECT MContractNo FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PID='{0}' AND (MContractNo like '%{2}%' or CHARINDEX(MContractNo,'{2}') > 0))

)

update FSO\_tb\_p\_DedcutNotReturnDetail set DeductType=0 where DeductType=1 and DNRID IN (select DNRId from #temp\_DeContractNo)

drop table #temp\_DeContractNo

END

INSERT INTO FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

SELECT DNRID,ISINPUT,SALESID,MAX(RULETYPE),MIN(DEDUCTAMOUNT),GETDATE() CREATEON,CREATEBY FROM

(SELECT b.DNRId,1 ISINPUT,c.SalesId,a.RuleType,

case when (a.RuleType=2 or a.RuleType=3) then

case when d.FDate >= DATEADD(MM,DATEDIFF(MM,0,a.BeginDate),0) and d.FDate < DATEADD(MM,DATEDIFF(MM,0,a.EndDate)+1,0) then (0-MAX(c.DeductAmount))

else 0 end

else (0-MAX(c.DeductAmount)) end DEDUCTAMOUNT

,'{1}' CREATEBY

FROM FSO\_tb\_p\_DedcutNotReturn B

inner join FSO\_tb\_p\_DedcutNotReturnConfig A on a.MContractNo=b.MContractNo

left join FSO\_tb\_p\_DedcutNotReturnDetail C on b.DNRId=c.DNRId

left join FSO\_tb\_p\_DedcutNotReturnPeriod D ON B.PId=D.PId

WHERE B.PId='{0}' and c.IsInput=1 AND (B.MContractNo like '%{2}%' or CHARINDEX(B.MContractNo,'{2}') > 0)

group by b.DNRId,c.SalesId,a.BeginDate,a.EndDate,d.FDate,a.RuleType)A

group by DNRId,ISINPUT,SALESID,CREATEBY

declare @PIdB int

select @PIdB=PID FROM #TEMP\_PID WHERE rowsn=4

--将特批不扣过期部分的应扣款插入扣款详情表

insert into FSO\_tb\_p\_DedcutNotReturnDetail (DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

select A.DNRId,0,Info.SalesId,1,abs(SUM(INFO.DeductAmount)) DeductAmount,GETDATE(),'{1}'

from (SELECT A.SalesId,A.DeductAmount DeductAmount,MContractNo

FROM(

select A.DNRId,A.PId,A.MContractNo,CASE WHEN ISNULL(DRC.BeginDate,'')='' THEN B.DeductAmount ELSE B.DeductAmount\*0.7 END DeductAmount,DRC.BeginDate,DRC.EndDate,B.SalesId

from FSO\_tb\_p\_DedcutNotReturn A

left join #TEMP\_PID P ON A.PId=P.PId

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join FSO\_tb\_p\_DedcutNotReturnConfig DRC ON A.MContractNo=DRC.MContractNo and RuleType=2 AND DATEADD(MM,DATEDIFF(MM,0,DRC.BeginDate),0)>=P.begindate AND DATEADD(MM,DATEDIFF(MM,0,DRC.EndDate)+1,0)<=P.enddate

WHERE A.PId<'{0}' AND B.DeductType=2 AND (A.MContractNo like '%{2}%' or CHARINDEX(A.MContractNo,'{2}') > 0) AND A.PID>=@PIdB

GROUP BY A.DNRId,A.PId,A.MContractNo,DRC.BeginDate,DRC.EndDate,B.SalesId,B.DeductAmount

UNION

select A.DNRId,A.PId,A.MContractNo,CASE WHEN ISNULL(DRC.BeginDate,'')='' THEN 0 ELSE B.DeductAmount\*0.3 END DeductAmount,DRC.BeginDate,DRC.EndDate,DRC.SalesVPId SalesId

from FSO\_tb\_p\_DedcutNotReturn A

left join #TEMP\_PID P ON A.PId=P.PId

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join FSO\_tb\_p\_DedcutNotReturnConfig DRC ON A.MContractNo=DRC.MContractNo and RuleType=2 AND DATEADD(MM,DATEDIFF(MM,0,DRC.BeginDate),0)>=P.begindate AND DATEADD(MM,DATEDIFF(MM,0,DRC.EndDate)+1,0)<=P.enddate

WHERE A.PId<'{0}' AND B.DeductType=2 AND (A.MContractNo like '%{2}%' or CHARINDEX(A.MContractNo,'{2}') > 0) AND A.PID>=@PIdB

GROUP BY A.DNRId,A.PId,A.MContractNo,DRC.BeginDate,DRC.EndDate,DRC.SalesVPId,B.DeductAmount

)A

GROUP BY PId,MContractNo,DeductAmount,SalesId,MContractNo

)Info

left join FSO\_tb\_p\_DedcutNotReturn A ON Info.MContractNo=A.MContractNo

WHERE A.PId='{0}' AND A.DNRID IN (select MIN(A.DNRId) DNRId from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

where B.DeductType=1

GROUP BY MContractNo )

GROUP BY DNRId,SalesId

ORDER BY DNRId

SELECT '计算成功'

DROP TABLE #TEMP\_BASEAMOUNT

END

ELSE

begin

SELECT '该期间已经进行过计算'

DROP TABLE #TEMP\_BASEAMOUNT

end

drop table #TEMP\_VoucherAI

drop table #TEMP\_VoucherPB

DROP TABLE #TEMP\_PID

", PId, UserId,ContractNos);

DataSet ds = new DataSet();

IDbConnection db = GetDbConnection();

SqlConnection conn = new SqlConnection(db.ConnectionString);

if (conn != null)

{

conn.Open();

SqlCommand cmd = new SqlCommand();

cmd.CommandTimeout = 60 \* 10;

cmd.Connection = conn;

cmd.CommandText = sql;

SqlDataAdapter da = new SqlDataAdapter(cmd);

da.Fill(ds);

conn.Close();

}

DataTable dt = ds.Tables[0];

dt.TableName = "CalcluDeductNoReturn";

string Result = dt.Rows[0][0].ToString();

return Result;

}

/// <summary>

/// 获取未回合同扣款详情

/// </summary>

/// <param name="DNRId">主表ID</param>

/// <returns></returns>

public DataTable GetDeductNoReturnDetail(string DNRId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select DRD.ID,DRD.DNRId,DRD.IsInput,DRD.SalesId,DRD.DeductType,DRD.DeductAmount,DRD.CreateOn,DRD.CreateBy,OP2.OperatorName CreateByName,drd.UpdateOn,DRD.UpdateBy,OP3.OperatorName UpdateByName,

CASE WHEN DRD.DeductType=0 THEN '暂扣'WHEN DRD.DeductType=1 THEN '必扣不退'WHEN DRD.DeductType=2 THEN '特批不扣' WHEN DRD.DeductType=3 THEN '其他免扣'WHEN DRD.DeductType=4 THEN '永久赦免' END DeductName,

DR.MContractNo,OP.OperatorName

from FSO\_tb\_p\_DedcutNotReturnDetail DRD

left join FSO\_tb\_p\_DedcutNotReturn DR ON DRD.DNRId=DR.DNRId

left join PR\_tb\_e\_Operator OP ON DRD.SalesId=OP.OperatorId

left join PR\_tb\_e\_Operator OP2 ON DRD.CreateBy=OP2.OperatorId

left join PR\_tb\_e\_Operator OP3 ON DRD.UpdateBy=OP3.OperatorId

where DRD.dnrid='{0}' and ((DRD.deducttype=2 and deductamount !=0) OR DRD.deducttype!=2) and DRD.DeductAmount <>0

", DNRId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "AddCalDate";

return dt;

}

/// <summary>

/// 添加或修改未回合同扣款详情信息进行金额调整

/// </summary>

/// <param name="model"></param>

/// <param name="IDo">0 添加 1 修改</param>

/// <returns></returns>

public DataTable AddorEditDeductNoReturn(DeductNoReturnDetailsModel model, int IDo)

{

DataTable dt = new DataTable();

string sql = string.Empty;

if (IDo == 0)

{

sql = string.Format(@"insert into FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,deducttype,deductAmount,createon,createby)

values('{0}','{1}','{2}','{3}','{4}',getdate(),'{5}')

", model.DNRId, model.IsInput, model.SalesId, model.DeductType, model.DeductAmount, model.UserId);

}

else if (IDo == 1)

{

sql = string.Format(@"update FSO\_tb\_p\_DedcutNotReturnDetail set

SalesId='{0}',DeductType='{1}',DeductAmount='{2}',UpdateOn=GETDATE(),UpdateBy='{3}'

where Id='{4}'

", model.SalesId, model.DeductType, model.DeductAmount, model.UserId,model.Id);

}

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "AddOrEditDeductNoReturn";

return dt;

}

/// <summary>

/// 删除未回合同详情

/// </summary>

/// <param name="DeleteId"></param>

public void DeletdNoReturnDetails(int DeleteId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

delete from FSO\_tb\_p\_DedcutNotReturnDetail where id='{0}'

", DeleteId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DeleteDetails";

}

/// <summary>

/// 修改扣款规则

/// </summary>

/// <param name="ContractNo"></param>

/// <param name="DeductType"></param>

/// <param name="dateBegin"></param>

/// <param name="dateEnd"></param>

public string addDeductRule(string ContractNo, int DeductType, DateTime dateBegin, DateTime dateEnd, string UserId, int FileId, string FileName,string SalesVPId,string Remark)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

if ('{1}'=2 or '{1}'=3)

begin

select CId,MContractNo,BeginDate,EndDate

into #temp\_Config

from FSO\_tb\_p\_DedcutNotReturnConfig where MContractNo='{0}' and (RuleType=2 or RuleType=3)

declare @CId int

declare @BeginDate datetime

declare @EndDate datetime

declare @result bit

select @result=1

while exists(select cid from #temp\_Config)

begin

select top 1 @CId=CId,@BeginDate=BeginDate,@EndDate=EndDate from #temp\_Config

if((DATEADD(MM,DATEDIFF(MM,0,@BeginDate),0)<='{2}' and DATEADD(MM,DATEDIFF(MM,0,@EndDate)+1,0)>'{2}') or ( DATEADD(M,0,DATEADD(MM,DATEDIFF(MM,0,@EndDate)+1,0)) >'{3}' and DATEADD(M,0,DATEADD(MM,DATEDIFF(MM,0,@BeginDate),0))<='{3}'))

select @result=0

delete from #temp\_Config where CId=@CId

end

if(@result=1)

begin

insert into FSO\_tb\_p\_DedcutNotReturnConfig(MContractNo,RuleType,BeginDate,EndDate,CreateOn,CreateBy,FileId,FileName,SalesVPId,[description])

values('{0}','{1}','{2}','{3}',GETDATE(),'{4}','{5}','{6}','{7}','{8}')

insert into FSO\_tb\_p\_DedcutNotReturnConfigFile(CId,FileId,FileName,isValid,CreateOn,CreateBy)

select CId,'{5}','{6}',1,GETDATE(),'{4}' from FSO\_tb\_p\_DedcutNotReturnConfig where MContractNo='{0}' and RuleType='{1}' and begindate='{2}' and enddate='{3}'

select '规则修改成功,请重新计算!'

end

else

select '规则生效时间冲突'

end

else

begin

insert into FSO\_tb\_p\_DedcutNotReturnConfig(MContractNo,RuleType,CreateOn,CreateBy,FileId,FileName,SalesVPId,[description])

values('{0}','{1}',GETDATE(),'{4}','{5}','{6}','{7}','{8}')

insert into FSO\_tb\_p\_DedcutNotReturnConfigFile(CId,FileId,FileName,isValid,CreateOn,CreateBy)

select CId,'{5}','{6}',1,GETDATE(),'{4}' from FSO\_tb\_p\_DedcutNotReturnConfig where MContractNo='{0}' and RuleType='{1}'

select '规则修改成功,请重新计算!'

end

", ContractNo, DeductType, dateBegin, dateEnd,UserId,FileId,FileName, SalesVPId, Remark);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "editDeductRule";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 获取规则修改历史

/// </summary>

/// <param name="ContractNo"></param>

/// <returns></returns>

public DataTable GetRuleHostory(string ContractNo)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select CId,ChangeNo,MContractNo,RuleType,BeginDate,EndDate,CreateOn,OP1.OperatorName CreateBy,UpdateOn,OP2.OperatorName UpdateBy,

CASE WHEN RuleType=0 THEN '暂扣'WHEN RuleType=1 THEN '必扣不退'WHEN RuleType=2 THEN '特批不扣' WHEN RuleType=3 THEN '其他免扣'WHEN RuleType=4 THEN '永久赦免' END RuleTypeName ,A.FileId,A.FileName

from FSO\_tb\_p\_DedcutNotReturnConfigL A

left join PR\_tb\_e\_Operator OP1 on a.CreateBy=OP1.OperatorId

left join PR\_tb\_e\_Operator OP2 on a.UpdateBy=OP2.OperatorId

where MContractNo='{0}'

order by changeno desc

", ContractNo);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetRuleHostory";

return dt;

}

/// <summary>

/// 获取设置规则列表

/// </summary>

/// <param name="model"></param>

/// <returns></returns>

public DataTable GetDeductRuleList(DeductNoReturnSelectModel model)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select distinct CId,A.MContractNo,RuleType,CONVERT(VARCHAR(10),YEAR(BeginDate)) +'-'+CONVERT(VARCHAR(10),MONTH(BeginDate)) BeginDate ,

CONVERT(VARCHAR(10),YEAR(EndDate)) +'-'+CONVERT(VARCHAR(10),MONTH(EndDate)) EndDate ,A.CreateOn,OP1.OperatorName CreateBy,A.UpdateOn,OP2.OperatorName UpdateBy,

CASE WHEN RuleType=0 THEN '暂扣'WHEN RuleType=1 THEN '必扣不退'WHEN RuleType=2 THEN '特批不扣' WHEN RuleType=3 THEN '其他免扣'WHEN RuleType=4 THEN '永久赦免' END RuleTypeName ,

A.FileId,A.FileName,'读取文件' ReadFile,OP3.OperatorName SalesVPId,A.[DESCRIPTION]

from FSO\_tb\_p\_DedcutNotReturnConfig A

left join PR\_tb\_e\_Operator OP1 on a.CreateBy=OP1.OperatorId

left join PR\_tb\_e\_Operator OP2 on a.UpdateBy=OP2.OperatorId

left join PR\_tb\_e\_Operator OP3 on a.SalesVPId=OP3.OperatorId

left join FSO\_TB\_P\_DEDCUTNOTRETURN B ON A.MContractNo=B.MContractNo

where 1=1 and (A.MContractNo like '%{0}%' or CHARINDEX(A.MContractNo,'{0}') > 0) and (RuleType like '%{2}%' or '{2}'=0)

order by A.MContractNo

--and B.PId='{1}'

", model.MContractNo,model.PId,model.DeductRuleType);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDeductRuleList";

return dt;

}

/// <summary>

/// 删除已设置的扣款规则

/// </summary>

/// <param name="MContractNo"></param>

public void DeleteDeductNoReturnRule(string CId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

insert into FSO\_tb\_p\_DedcutNotReturnConfigL(CId,ChangeNo,MContractNo,RuleType,BeginDate,EndDate,CreateOn,CreateBy,UpdateOn,UpdateBy)

select CId,(select count(\*) from FSO\_tb\_p\_DedcutNotReturnConfigL where MContractNo='{0}')+1,MContractNo,RuleType,BeginDate,EndDate,CreateOn,CreateBy,UpdateOn,UpdateBy from FSO\_tb\_p\_DedcutNotReturnConfig

where CId='{0}'

delete from FSO\_tb\_p\_DedcutNotReturnConfig where CId='{0}'

", CId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DeleteDeductNoReturnRule";

}

/// <summary>

/// 未回合同扣款审核

/// </summary>

/// <param name="PId">期间ID</param>

/// <param name="userId">使用人</param>

/// <param name="iDo">0 审核 1 审核确认 9 反审</param>

public string DecudtNoReturnAudit(int PId, string userId, int iDo)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

if '{2}'=0

begin

IF(SELECT TOP 1 AState FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PId='{0}')=1

begin

update FSO\_TB\_P\_DEDCUTNOTRETURN set FAuditBy='{1}' ,FAuditOn=GETDATE(),AState=3

where PId='{0}'

select '初审成功'

end

else

select '只有待审核状态的合同才可进行初审'

end

else if '{2}'=1

begin

IF(SELECT TOP 1 AState FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PId='{0}')=2

begin

update FSO\_TB\_P\_DEDCUTNOTRETURN set SAuditBy='{1}' ,SAuditOn=GETDATE(),AState=3

where PId='{0}'

select '审核确认成功'

end

else

select '只有审核状态的合同才可进行审核确认'

end

else if '{2}'=9

begin

IF(SELECT TOP 1 AState FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PId='{0}')=2

begin

update FSO\_TB\_P\_DEDCUTNOTRETURN set FAuditBy='' ,FAuditOn=null,AState=1

where PId='{0}'

select '反审成功'

end

ELSE IF(SELECT TOP 1 AState FROM FSO\_TB\_P\_DEDCUTNOTRETURN WHERE PId='{0}')=3

begin

update FSO\_TB\_P\_DEDCUTNOTRETURN set SAuditBy='' ,SAuditOn=null,AState=1

where PId='{0}'

select '反审成功'

end

ELSE

SELECT '待审核状态不可进行反审'

end

", PId, userId, iDo);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DecudtNoReturnAudit";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 获取回款信息详情

/// </summary>

/// <param name="Mcontractno"></param>

/// <returns></returns>

public DataTable GetRefundDetails(DeductNoReturnSelectModel model)

{

DataTable dt = new DataTable();

string sql = string.Empty;

string sqlEnd = string.Empty;

string sqlwhere = string.Empty;

if (!model.IncludeHistory)

{

sqlwhere = string.Format(@"AND MContractNo in (SELECT A.MContractNo FROM FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId =B.DNRId

WHERE A.PId>7

GROUP BY A.PId,A.MContractNo

HAVING SUM(B.DeductAmount)>0 ) ");

}

//正常查询

if (!model.MustDeduct)

{

sql = string.Format(@"

DECLARE @FDATE DATETIME

SELECT TOP 1 @FDATE=FDATE FROM FSO\_tb\_p\_DedcutNotReturnPeriod ORDER BY PId DESC

--先找到所有未回确认的子合同 再在里面排除已经回合同的

select SC.ContractId,SC.ContractNo,SC.MainContractNo,SC.ContReturnId,SC.ContReturnDate,SP.ConfirmState,SP.ConfirmDate

into #temp\_Confirm

FROM FSO\_tb\_e\_SalesContract SC

LEFT JOIN FSO\_tb\_e\_SalesPublish SP ON SC.ContractId=SP.ContractId

WHERE ConfirmState=0 OR (ConfirmState !=0 AND ConfirmDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0))

--在从里面排除回合同的子合同

select CASE WHEN ISNULL(A.MainContractNo,'')='' THEN A.ContractNo ELSE A.MainContractNo END CONTRACTNO

into #temp\_ContReturn

from #temp\_Confirm A

WHERE A.ContReturnId NOT IN (1,3) OR (A.ContReturnId in (1,3) AND A.ContReturnDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0))

group by CASE WHEN ISNULL(A.MainContractNo,'')='' THEN A.ContractNo ELSE A.MainContractNo END

--从合同销售表中排除以上的合同就是完全回合同和回确认的合同

select CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END CONTRACTNO

into #temp\_contractno

from FSO\_tb\_e\_SalesContract SC

where not exists(select CONTRACTNO from #temp\_ContReturn C where C.CONTRACTNO=(CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END))

GROUP BY CASE WHEN ISNULL(SC.MainContractNo,'')='' THEN SC.ContractNo ELSE SC.MainContractNo END

select \* from (SELECT A.\*,CONVERT(VARCHAR(10),YEAR(FDATE)) +'-'+CONVERT(VARCHAR(10),MONTH(FDATE)) FDATE FROM (select MAX(A.PId) PId,A.MContractNo,B.SalesId,SUM(B.DeductAmount) DeductAmount,op.OperatorName salesname,C.RefundAmount,SUM(B.DeductAmount)-isnull(C.RefundAmount,0) difAmount

,SA.SignAName,BR.Brand,CT.CustomerName,

DataCenterCity = Case When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C001.A01.D009' Then '上海数据中心'

When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C002.A01.D009' Then '北京数据中心'

Else '' End

from FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join PR\_tb\_e\_Operator OP on b.SalesId=op.OperatorId

left join (select MContractNo,sum(RefundAmount) RefundAmount from FSO\_tb\_p\_DedcutNotReturnRefund

group by MContractNo)C ON A.MContractNo=C.MContractNo

inner join #temp\_contractno D ON A.MContractNo=D.CONTRACTNO

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.MContractNo=SC.ContractNo

LEFT JOIN FSO\_tb\_b\_SignAName SA ON SC.SignAName=SA.Code

LEFT JOIN FSO\_tb\_e\_Customer CT ON SC.SignCustId=CT.CustomerId

LEFT JOIN FSO\_tb\_b\_Brand BR ON ISNULL(SC.Brand,0)=BR.Code

left join (select DB.City,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(City,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by City,OP.DeptId)AC ON SC.SignCity=AC.City

LEFT JOIN(select DB.DeptId Dept,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(DB.DeptId,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by DB.DeptId,OP.DeptId)AD on CHARINDEX(AD.Dept,SC.SignDeptId) > 0 and CHARINDEX(AD.Dept,SC.SignDeptId) <2

WHERE (A.MContractNo like '%{0}%' or CHARINDEX(A.MContractNo,'{0}') > 0)

AND isnull(B.SALESID,'') LIKE '%{1}%' and a.AState=3

and ISNULL(SA.SignAName,'') like '%{2}%'

and ISNULL(BR.Brand,'') like '%{3}%'

and ISNULL(CT.CustomerName,'') like '%{4}%'

AND A.MContractNo not in( select MContractNo from FSO\_tb\_p\_DedcutNotReturn where DNRId in (SELECT DNRId FROM FSO\_tb\_p\_DedcutNotReturnDetail WHERE DeductType=1))

group by A.MContractNo,B.SalesId,op.OperatorName,C.RefundAmount,SA.SignAName,BR.Brand,CT.CustomerName,AC.DeptId,AD.DeptId)A

left join FSO\_tb\_p\_DedcutNotReturnPeriod DN ON A.PId=DN.PId)A

where 1=1 AND DataCenterCity LIKE '%{5}%' {6}

", model.MContractNo, model.SalesId, model.SignAName, model.Brand, model.CustomerName, model.DataCenter, sqlwhere);

sqlEnd = string.Format(@"

DROP TABLE #temp\_ContReturn

DROP TABLE #temp\_Confirm

drop table #temp\_contractno");

}

//包含所有

else

{

sql = string.Format(@"

select \* from (SELECT A.\*,CONVERT(VARCHAR(10),YEAR(FDATE)) +'-'+CONVERT(VARCHAR(10),MONTH(FDATE)) FDATE FROM (select MAX(A.PId) PId,A.MContractNo,B.SalesId,SUM(B.DeductAmount) DeductAmount,op.OperatorName salesname,C.RefundAmount,SUM(B.DeductAmount)-isnull(C.RefundAmount,0) difAmount

,SA.SignAName,BR.Brand,CT.CustomerName,

DataCenterCity = Case When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C001.A01.D009' Then '上海数据中心'

When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C002.A01.D009' Then '北京数据中心'

Else '' End

from FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join PR\_tb\_e\_Operator OP on b.SalesId=op.OperatorId

left join (select MContractNo,sum(RefundAmount) RefundAmount from FSO\_tb\_p\_DedcutNotReturnRefund

group by MContractNo)C ON A.MContractNo=C.MContractNo

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.MContractNo=SC.ContractNo

LEFT JOIN FSO\_tb\_b\_SignAName SA ON SC.SignAName=SA.Code

LEFT JOIN FSO\_tb\_e\_Customer CT ON SC.SignCustId=CT.CustomerId

LEFT JOIN FSO\_tb\_b\_Brand BR ON ISNULL(SC.Brand,0)=BR.Code

left join (select DB.City,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(City,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by City,OP.DeptId)AC ON SC.SignCity=AC.City

LEFT JOIN(select DB.DeptId Dept,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(DB.DeptId,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by DB.DeptId,OP.DeptId)AD on CHARINDEX(AD.Dept,SC.SignDeptId) > 0 and CHARINDEX(AD.Dept,SC.SignDeptId) <2

WHERE (A.MContractNo like '%{0}%' or CHARINDEX(A.MContractNo,'{0}') > 0)

AND isnull(B.SALESID,'') LIKE '%{1}%' and a.AState=3

and ISNULL(SA.SignAName,'') like '%{2}%'

and ISNULL(BR.Brand,'') like '%{3}%'

and ISNULL(CT.CustomerName,'') like '%{4}%'

group by A.MContractNo,B.SalesId,op.OperatorName,C.RefundAmount,SA.SignAName,BR.Brand,CT.CustomerName,AC.DeptId,AD.DeptId)A

left join FSO\_tb\_p\_DedcutNotReturnPeriod DN ON A.PId=DN.PId)A

where 1=1 AND DataCenterCity LIKE '%{5}%' {6}

", model.MContractNo, model.SalesId, model.SignAName, model.Brand, model.CustomerName, model.DataCenter, sqlwhere);

sqlEnd = string.Empty;

}

if (!model.IsZero)

sql += "AND difAmount!=0";

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql+ sqlEnd);

dt.TableName = "GetRefundDetails";

return dt;

}

/// <summary>

/// 获取退款历史

/// </summary>

/// <param name="Mcontractno"></param>

/// <returns></returns>

public DataTable GetRefundHistoryDetails(DeductNoReturnSelectModel model)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select A.\*,OP.OperatorName,'读取文件' ReadFile,case when AState=0 then '待审核' when AState=1 then '审核完成' else '审核退回' end AStateName

,SA.SignAName,BR.Brand,CT.CustomerName

from FSO\_tb\_p\_DedcutNotReturnRefund A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.MContractNo=SC.ContractNo

LEFT JOIN FSO\_tb\_b\_SignAName SA ON SC.SignAName=SA.Code

LEFT JOIN FSO\_tb\_e\_Customer CT ON SC.SignCustId=CT.CustomerId

LEFT JOIN FSO\_tb\_b\_Brand BR ON ISNULL(SC.Brand,0)=BR.Code

left join PR\_tb\_e\_Operator OP on a.SalesId=op.OperatorId

WHERE (A.MContractNo like '%{0}%' or CHARINDEX(A.MContractNo,'{0}') > 0)

and ISNULL(A.SalesId,'') like '%{1}%'

and A.FDate>=DATEADD(MM,DATEDIFF(MM,0,'{2}'),0) AND A.FDate < DATEADD(MM,DATEDIFF(MM,0,'{3}')+1,0)

and ISNULL(SA.SignAName,'') like '%{4}%'

and ISNULL(BR.Brand,'') like '%{5}%'

and ISNULL(CT.CustomerName,'') like '%{6}%'

", model.MContractNo, model.SalesId, model.BeginDate,model.EndDate,model.SignAName,model.Brand,model.CustomerName);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetReturnHistoryDetails";

return dt;

}

/// <summary>

/// 获取子合同回款详情

/// </summary>

/// <param name="McontractNo"></param>

/// <returns></returns>

public DataTable GetRefundDetail(string McontractNo, string SalesId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

DECLARE @FDATE DATETIME

SELECT TOP 1 @FDATE=FDATE FROM FSO\_tb\_p\_DedcutNotReturnPeriod ORDER BY PId DESC

SELECT A.\*,B.SalesId,B.DeductAmount,B.DeductType,C.CONTRACTID,C.BASEAMOUNT CBASEAMOUNT,OP.OperatorName SignerName,CONVERT(VARCHAR(10),YEAR(FDATE)) +'-'+CONVERT(VARCHAR(10),MONTH(FDATE)) FDATE,case when C.BASEAMOUNT<0 then 0 else C.BASEAMOUNT/A.BaseAmount end SCALE

INTO #TEMP\_RETURN

from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnContract C ON A.PId=C.PID AND A.MContractNo=C.MCONTRACTNO

LEFT JOIN PR\_tb\_e\_Operator op on B.SalesId=OP.OperatorId

left join FSO\_tb\_p\_DedcutNotReturnPeriod DRP ON A.PID=DRP.PId

DELETE FROM #TEMP\_RETURN WHERE DNRID IN (SELECT DNRId FROM #TEMP\_RETURN GROUP BY DNRId HAVING SUM(DeductAmount)=0)

SELECT SC.ContractNo,DeductAmount,DeductAmount\*A.SCALE SDeductAmount,A.\* ,case when sc.ContReturnId in (1,3) and ContReturnDate<DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0) then 1 else 0 end ContReturn,CASE WHEN C.confirm='no' then 0 else 1 end confirm

FROM #TEMP\_RETURN A

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.CONTRACTID=SC.ContractId

left join (select distinct contractid,'no' confirm from FSO\_tb\_e\_SalesPublish WHERE ConfirmState=0 or (ConfirmState!=0 and ConfirmDate>=DATEADD(MM,DATEDIFF(MM,0,@FDATE)+1,0) ))C ON A.ContractId=C.ContractId

WHERE A.MContractNo ='{0}'

DROP TABLE #TEMP\_RETURN

", McontractNo);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetReturnHistoryDetails";

return dt;

}

/// <summary>

/// 添加未回合同扣款退款信息

/// </summary>

/// <param name="McontractNo"></param>

/// <returns></returns>

public void addRefundBackDetails(DeductReturnBack model)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

insert into FSO\_tb\_p\_DedcutNotReturnRefund (FDate,MContractNo,SalesId,RefundAmount,FileId,FileName,Astate,CreateOn,CreateBy)

values('{0}','{1}','{2}','{3}','{4}','{5}',0,GetDate(),'{6}')

", model.DateRe,model.MContractNo,model.SignerId,model.ReturnAmount,model.FileId,model.FileName,model.UserId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "addRefundBackDetails";

}

/// <summary>

/// 获取该合同暂扣的总额

/// </summary>

/// <param name="ContractNo"></param>

/// <returns></returns>

public double GetTotalDeductAmount(string ContractNo)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select SUM(B.DeductAmount) DeductAmount from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

WHERE A.MContractNo='{0}'

", ContractNo);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetTotalDeductAmount";

return Convert.ToDouble(dt.Rows[0][0]);

}

/// <summary>

/// 审核未回合同扣款退款

/// </summary>

/// <param name="State">1 通过 2 退回</param>

public string AuditReturnRefund(int State, string RIds, string UserId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

if('{0}'=1)

begin

if exists(select RId from FSO\_tb\_p\_DedcutNotReturnRefund where RId in ({1}) and AState=1)

begin

select '审核失败,所选合同里有已审核状态的'

end

else

begin

update FSO\_tb\_p\_DedcutNotReturnRefund set AState='{0}' where RId in ({1})

select '审核完成'

end

end

else if('{0}'=2)

begin

if exists(select RId from FSO\_tb\_p\_DedcutNotReturnRefund where RId in ({1}) and AState!=1)

begin

select '反审失败,所选合同里有未审核或已退回状态的'

end

else

begin

update FSO\_tb\_p\_DedcutNotReturnRefund set AState='{0}' where RId in ({1})

select '反审完成'

end

end

", State,RIds,UserId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetTotalDeductAmount";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 未回合同扣款退款根据销售员汇总

/// </summary>

/// <param name="date">查询月份</param>

public DataTable GetRefundHistoryDetailsTotal(DateTime date)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select A.SalesId,OP.OperatorName SalesName,SUM(RefundAmount) TotalAmount from FSO\_tb\_p\_DedcutNotReturnRefund A

LEFT JOIN PR\_tb\_e\_Operator OP ON A.SalesId=OP.OperatorId

where A.FDate>=DATEADD(MM,DATEDIFF(MM,0,'{0}'),0) AND A.FDate<=DATEADD(MM,DATEDIFF(MM,0,'{1}'),0)

GROUP BY A.SalesId,OP.OperatorName

", date,date.AddMonths(1));

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetRefundHistoryDetailsTotal";

return dt;

}

/// <summary>

/// 根据销售员汇总当月的扣款金额

/// </summary>

/// <param name="PId"></param>

/// <param name="DataCenter"></param>

/// <returns></returns>

public DataTable GetDeductNoReturnTotal(int PId, string DataCenter)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select SalesId,DeductAmountTotal,SalesName,DataCenterCity

from(select b.SalesId,SUM(b.DeductAmount) DeductAmountTotal,op.OperatorName SalesName

,DataCenterCity = Case When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C001.A01.D009' Then '上海数据中心'

When (CASE WHEN ISNULL(AC.DeptId,'')='' THEN AD.DeptId ELSE AC.DeptId END) = 'YJ.C002.A01.D009' Then '北京数据中心'

Else '' End

from FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join PR\_tb\_e\_Operator OP on b.SalesId=op.OperatorId

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.MContractNo=SC.ContractNo

left join (select DB.City,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(City,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by City,OP.DeptId)AC ON SC.SignCity=AC.City

LEFT JOIN(select DB.DeptId Dept,OP.DeptId from Desc\_tb\_BrandAuditUser DB

LEFT JOIN PR\_tb\_e\_Operator op on DB.UserId=op.OperatorId and op.IsStop=0

where isnull(DB.DeptId,'')!='' and op.DeptId in ('YJ.C001.A01.D009','YJ.C002.A01.D009') and GETDATE() between BeginDate and EndDate

group by DB.DeptId,OP.DeptId)AD on CHARINDEX(AD.Dept,SC.SignDeptId) = 1

where A.PId='{0}' AND A.DNRID NOT IN (SELECT A.DNRId from FSO\_TB\_P\_DEDCUTNOTRETURN A

inner JOIN FSO\_tb\_p\_DedcutNotReturnConfig B ON A.MContractNo=B.MContractNo AND B.RuleType=4

where a.PId='{0}')

GROUP BY b.SalesId,op.OperatorName,AC.DeptId,AD.DeptId)A

where DataCenterCity like '%{1}%' and DeductAmountTotal!=0

", PId,DataCenter);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDeductNoReturnTotal";

return dt;

}

/// <summary>

/// 删除扣款信息

/// </summary>

/// <param name="RId"></param>

public string DeleteRefund(string RId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

if (select AState from FSO\_tb\_p\_DedcutNotReturnRefund where RId='{0}') in(0,2)

begin

delete from FSO\_tb\_p\_DedcutNotReturnRefund where RId='{0}'

select '删除成功'

end

else

select '该退款记录已审核，请先反审再进行删除'

", RId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DeleteRefund";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 不考虑计算期间获取未回合同扣款的主表信息

/// </summary>

/// <param name="model"></param>

/// <returns></returns>

public DataTable GetDeductNoReturnAll(DeductNoReturnSelectModel model)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select A.DNRId,A.PId,A.MContractNo,A.BaseAmount,A.AState,

B.SalesId,OP.OperatorName,SUM(B.DeductAmount) DeductAmount,SC.CSignDate,SC.SignDeptId,

CASE WHEN SC.ContLevel IN(1,3,8) THEN '全国单'ELSE '二线单'END ContLevelS,

case when A.AState=1 then '待审核' when A.AState=2 then '审核' when A.AState=3 then '审核确认' end AStateName,

DataCenterCity = Case When SubmitO.DeptId = 'YJ.C001.A01.D009' Then '上海数据中心'

When SubmitO.DeptId = 'YJ.C002.A01.D009' Then '北京数据中心'

Else '' End

into #temp\_noreturn

from FSO\_TB\_P\_DEDCUTNOTRETURN A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

LEFT JOIN PR\_tb\_e\_Operator OP ON B.SalesId=OP.OperatorId

LEFT JOIN FSO\_tb\_e\_SalesContract SC ON A.MContractNo=SC.ContractNo

inner join PR\_tb\_e\_Operator SubmitO on SubmitO.OperatorId = SC.SubmitOperator

where MContractNo like '%{0}%'

and SalesId like '%{1}%'

and a.MContractNo not in (select ContractNo from FSO\_tb\_p\_DedcutNotReturnConfig where DeductType=4)

GROUP BY A.DNRId,A.PId,A.MContractNo,A.BaseAmount,A.AState,A.CreateOn,A.CreateBy,A.FAuditOn,A.FAuditBy,A.SAuditOn,

A.SAuditBy,B.SalesId,OP.OperatorName,SC.ContLevel

,OP.OperatorName,SubmitO.DeptId,SC.CSignDate,SC.SignDeptId

SELECT A.\*,CONVERT(VARCHAR(10),YEAR(fdate)) +'-'+CONVERT(VARCHAR(10),MONTH(fdate)) fdate

from #temp\_noreturn A

left join FSO\_tb\_p\_DedcutNotReturnPeriod B ON A.PID=B.PID

where DataCenterCity like '%{2}%'

drop table #temp\_noreturn

", model.MContractNo,model.SalesId,model.DataCenter);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDeductNoReturnAll";

return dt;

}

/// <summary>

/// 获取未回合同扣款排除部门

/// </summary>

/// <returns></returns>

public DataTable GetDedcutNotReturnExclude(string DeptId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select A.EId,a.DeptId,a.BeginDate,a.EndDate,a.CreateOn,a.UpdateOn,B.DeptName,op.OperatorName CreateBy,OP2.OperatorName UpdateBy from FSO\_tb\_p\_DedcutNotReturnExclude A

left join PR\_tb\_b\_Department B ON A.DEPTID=B.DeptId

left join PR\_tb\_e\_Operator OP on a.CreateBy=op.OperatorId

left join PR\_tb\_e\_Operator OP2 on a.UpdateBy=OP2.OperatorId

where a.DeptId like '%{0}%' or DeptName like '%{0}%'

", DeptId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDedcutNotReturnExclude";

return dt;

}

/// <summary>

/// 获取未回合同扣款排除部门

/// </summary>

/// <returns></returns>

public DataTable GetDedcutNotReturnExcludeById(string EId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select A.EId,a.DeptId,a.BeginDate,a.EndDate,a.CreateOn,a.UpdateOn,B.DeptName,op.OperatorName CreateBy,OP2.OperatorName UpdateBy from FSO\_tb\_p\_DedcutNotReturnExclude A

left join PR\_tb\_b\_Department B ON A.DEPTID=B.DeptId

left join PR\_tb\_e\_Operator OP on a.CreateBy=op.OperatorId

left join PR\_tb\_e\_Operator OP2 on a.UpdateBy=OP2.OperatorId

where A.EId=‘{0}’'

", EId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDedcutNotReturnExclude";

return dt;

}

/// <summary>

/// 删除排除部门

/// </summary>

/// <returns></returns>

public string DelDedcutNotReturnExclude(int RId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

delete from FSO\_tb\_p\_DedcutNotReturnExclude where EId='{0}'

select '删除成功'

",RId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DelDedcutNotReturnExclude";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 添加和编辑排除部门

/// </summary>

/// <param name="RId">排除部门表ID</param>

/// <param name="iDo">0 添加 1 修改</param>

/// <param name="DeptId">部门id</param>

/// <param name="Datebegin">开始时间</param>

/// <param name="DateEnd">结束时间</param>

/// <param name="UserId">操作人</param>

/// <returns></returns>

public string EditDedcutNotReturnExclude(int RId, int iDo, string DeptId, DateTime Datebegin, DateTime DateEnd, string UserId)

{

DataTable dt = new DataTable();

string sql = string.Empty;

if (iDo == 0)

{

sql = string.Format(@"

insert into FSO\_tb\_p\_DedcutNotReturnExclude (DeptId,BeginDate,EndDate,CreateBy,CreateOn)

values('{0}','{1}','{2}','{3}',GETDATE())

select '添加成功'

", DeptId,Datebegin,DateEnd,UserId);

}

else if (iDo == 1)

{

sql = string.Format(@"

update FSO\_tb\_p\_DedcutNotReturnExclude

set DeptId='{0}',BeginDate='{1}',EndDate='{2}',UpdateBy='{3}',UpdateOn=GETDATE()

where EId='{4}'

select '修改成功'

", DeptId, Datebegin, DateEnd, UserId,RId);

}

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "EditDedcutNotReturnExclude";

return dt.Rows[0][0].ToString();

}

/// <summary>

/// 读取扣款规则文件列表

/// </summary>

/// <param name="CId"></param>

/// <returns></returns>

public DataTable GetDedcutNotReturnConfigFile(string CId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"select A.\*,op.OperatorName from FSO\_tb\_p\_DedcutNotReturnConfigFile A

LEFT JOIN PR\_tb\_e\_Operator OP ON A.CreateBy=op.OperatorId

where CId='{0}' AND ISVALID=1", CId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDedcutNotReturnConfigFile";

return dt;

}

/// <summary>

/// 删除扣款规则审核文件

/// </summary>

/// <param name="FId"></param>

/// <returns></returns>

public DataTable DelDedcutNotReturnConfigFile(string FId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"update FSO\_tb\_p\_DedcutNotReturnConfigFile set ISVALID=0 where FId='{0}'", FId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "DelDedcutNotReturnConfigFile";

return dt;

}

/// <summary>

/// 上传扣款规则审核文件

/// </summary>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <returns></returns>

public string AddDedcutNotReturnConfigFile(string CId, int FileId, string FileName, string UserId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"insert into FSO\_tb\_p\_DedcutNotReturnConfigFile (CId,FileId,FileName,IsValid,CreateOn,CreateBy)

values('{0}','{1}','{2}',1,GETDATE(),'{3}')", CId, FileId, FileName, UserId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "AddDedcutNotReturnConfigFile";

return "上传成功";

}

/// <summary>

/// 根据主合同号批量上传扣款规则审核文件

/// </summary>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <param name=""></param>

/// <returns></returns>

public string BatchAddDedcutNotReturnConfigFile(string McontractNos, int FileId, string FileName, string UserId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

insert into FSO\_tb\_p\_DedcutNotReturnConfigFile (CId,FileId,FileName,IsValid,CreateOn,CreateBy)

select CId,'{1}','{2}',1,GETDATE(),'{3}' from FSO\_tb\_p\_DedcutNotReturnConfig where CHARINDEX(MContractNo,'{0}')>0 ", McontractNos, FileId, FileName, UserId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "BatchAddDedcutNotReturnConfigFile";

return "上传成功";

}

/// <summary>

/// 获取发布转移合同

/// </summary>

/// <param name=""></param>

/// <returns></returns>

public DataTable GetDeductNotReturnPubRela(string MContactNo,string OContractNo, string PId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

--找出所有含有必扣不退合同的合同号

select MContractNo

into #temp\_McontractNo

from FSO\_tb\_p\_DedcutNotReturn A

where EXISTS(SELECT DNRId FROM FSO\_tb\_p\_DedcutNotReturnDetail WHERE DeductType=1 AND DNRId=A.DNRId)

GROUP BY MContractNo

select A.DNRId,A.PId,CONVERT(VARCHAR(10),YEAR(FDATE)) +'-'+CONVERT(VARCHAR(10),MONTH(FDATE)) FDATE,A.MContractNo,SUM(B.DeductAmount) DeductAmount ,B.SalesId ,OP.OperatorName

from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

left join FSO\_tb\_p\_DedcutNotReturnPeriod C ON A.PId=C.PId

left join PR\_tb\_e\_Operator OP ON B.SalesId=OP.OperatorId

where CHARINDEX(A.MContractNo,'{0}')>0 AND A.AState=3 AND A.MContractNo <>'{2}'

AND NOT EXISTS (SELECT MContractNo FROM #temp\_McontractNo WHERE MContractNo=A.MContractNo)

GROUP BY A.DNRId,A.PId,A.MContractNo,FDATE,B.SalesId ,OP.OperatorName

DROP TABLE #temp\_McontractNo

", OContractNo,PId, MContactNo);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDeductNotReturnPubRela";

return dt;

}

/// <summary>

/// 发布转移合同调整

/// </summary>

/// <param name="McontractNo"></param>

/// <param name="datas"></param>

/// <returns></returns>

public string EditDeductNotReturnPubRela(string McontractNo,string PId, List<EditDeductNotReturnPubRelaModel> Modellist,string UserId)

{

foreach(var i in Modellist)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

if not exists(select DNRId from FSO\_tb\_p\_DedcutNotReturn where PId='{1}' AND MContractNo='{0}')

begin

--将新合同主表信息插入

insert into FSO\_tb\_p\_DedcutNotReturn(PId,MContractNo,BaseAmount,AState,CreateOn,CreateBy)

select '{1}','{0}',BaseAmount,AState,CreateOn,CreateBy from FSO\_tb\_p\_DedcutNotReturn

where PId='{1}' AND MContractNo='{4}'

end

--将新合同子表信息插入

insert into FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

select A.DNRId,0,'{3}',3,'{2}',GETDATE(),'{5}' from FSO\_tb\_p\_DedcutNotReturn A

WHERE A.PId='{1}' AND A.MContractNo='{0}'

--对原合同插入一个抵消金额

insert into FSO\_tb\_p\_DedcutNotReturnDetail(DNRId,IsInput,SalesId,DeductType,DeductAmount,CreateOn,CreateBy)

select A.DNRId,0,'{3}',3,0-({2}),GETDATE(),'{5}' from FSO\_tb\_p\_DedcutNotReturn A

WHERE A.PId='{1}' AND A.MContractNo='{4}'

", McontractNo, i.Pid,i.adjustAmount,i.SalesId,i.MContractNo,UserId,PId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "EditDeductNotReturnPubRela";

}

return "修改成功";

}

/// <summary>

/// 修改VO经理

/// </summary>

/// <param name="CId"></param>

/// <param name="SalesVPId"></param>

/// <returns></returns>

public string EditSalesVPId(string CId, string SalesVPId)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

update FSO\_tb\_p\_DedcutNotReturnConfig set SalesVPId='{1}' WHERE CId='{0}'

", CId, SalesVPId);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "EditSalesVPId";

return "修改成功";

}

/// <summary>

///获取发布转移列表

/// </summary>

/// <param name="ContractNos"></param>

/// <returns></returns>

public DataTable GetPubTran(string ContractNos)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

--找出已转移的合同的前合同号

select \*

INTO #TEMP\_ROCONTRACTNO

from (select CASE WHEN ISNULL(SC1.MainContractNo,'')='' THEN SC1.ContractNo ELSE SC1.MainContractNo END OContractNo,

CASE WHEN ISNULL(SC2.MainContractNo,'')='' THEN SC2.ContractNo ELSE SC2.MainContractNo END RContractNo

from FSO\_tb\_e\_SalesPublishRelation A

LEFT JOIN FSO\_tb\_e\_SalesContract SC1 ON A.OContractNo=SC1.ContractNo

LEFT JOIN FSO\_tb\_e\_SalesContract SC2 ON A.RContractNo=SC2.ContractNo

where OFromType=1 and RFromType=1

union

Select CASE WHEN ISNULL(SC1.MainContractNo,'')='' THEN SC1.ContractNo ELSE SC1.MainContractNo END OContractNo,

CASE WHEN ISNULL(SC2.MainContractNo,'')='' THEN SC2.ContractNo ELSE SC2.MainContractNo END RContractNo

From Desc\_tb\_PublishRelation pr

inner join FSO\_tb\_e\_SalesPublish sp1 on sp1.DescPublishId = pr.OPublishId

inner join FSO\_tb\_e\_SalesPublish sp2 on sp2.DescPublishId = pr.RPublishId

inner join FSO\_tb\_e\_SalesContract sc1 on sp1.contractId = sc1.ContractId

inner join FSO\_tb\_e\_SalesContract sc2 on sp2.contractId = sc2.ContractId

where pr.RType ='PbSame') A

group by OContractNo,RContractNo

--找出所有含有必扣不退合同的合同号

select MContractNo

into #temp\_McontractNo

from FSO\_tb\_p\_DedcutNotReturn A

where EXISTS(SELECT DNRId FROM FSO\_tb\_p\_DedcutNotReturnDetail WHERE DeductType=1 AND DNRId=A.DNRId)

GROUP BY MContractNo

select A.RContractNo MCONTRACTNO,

STUFF((SELECT ','+ISNULL(B.OContractNo,'') FROM #TEMP\_ROCONTRACTNO B WHERE A.RContractNo = B.RContractNo AND A.RContractNo<>B.OContractNo FOR XML PATH('')),1,1,'') AS OContractNo,SUM(D.DeductAmount) DeductAmount

FROM #TEMP\_ROCONTRACTNO A

left join FSO\_tb\_p\_DedcutNotReturn C ON A.OContractNo=C.MContractNo

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail D ON C.DNRId=D.DNRId

WHERE (A.RContractNo like '%{0}%' or CHARINDEX(A.RContractNo,'{0}') > 0) AND RContractNo<>OContractNo

AND NOT EXISTS (SELECT MContractNo FROM #temp\_McontractNo WHERE MContractNo=C.MContractNo)

AND C.AState=3

GROUP BY A.RContractNo

HAVING SUM(D.DeductAmount)>0

DROP TABLE #TEMP\_ROCONTRACTNO

DROP TABLE #temp\_McontractNo

", ContractNos);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetPubTran";

return dt;

}

/// <summary>

///未回合同扣款根据主合同号获取销售员

/// </summary>

/// <param name="ContractNos"></param>

/// <returns></returns>

public string GetDeductNotReturnSalesId(string ContractNo)

{

DataTable dt = new DataTable();

string sql = string.Format(@"

select TOP 1 B.SalesId from FSO\_tb\_p\_DedcutNotReturn A

LEFT JOIN FSO\_tb\_p\_DedcutNotReturnDetail B ON A.DNRId=B.DNRId

WHERE A.MContractNo='{0}'

", ContractNo);

dt = AdoTemplate.DataTableCreate(CommandType.Text, sql);

dt.TableName = "GetDeductNotReturnSalesId";

return dt.Rows[0][0].ToString();

}

3界面:

public partial class AddCalDateForm : Form

{

#region 私有成员

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

#region 公共成员

public DateTime NewDate;

#endregion

public AddCalDateForm()

{

InitializeComponent();

}

#region 私有事件

/// <summary>

/// 加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void AddCalDateForm\_Load(object sender, EventArgs e)

{

DateCalDate.Value = NewDate;

}

/// <summary>

/// 添加期间确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void bntSubmit\_Click(object sender, EventArgs e)

{

DateTime CalDate = DateCalDate.Value;

DateTime AIDate = DateAIDate.Value;

DateTime PbDate = DatePbDate.Value;

if (fSOServiceClient.AddDeductNoReturnCalDate(CalDate, AIDate, PbDate, UserId))

{

MessageBox.Show("期间添加成功", "提示");

Close();

}

}

/// <summary>

/// 添加期间取消

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

#endregion

}

public partial class DedcutNotReturnExcludeForm : Form

{

#region 私有成员

private static DedcutNotReturnExcludeForm instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

public DedcutNotReturnExcludeForm()

{

InitializeComponent();

dgvExclude.AutoGenerateColumns = false;

}

#region 私有事件

/// <summary>

/// 添加部门

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 添加部门ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DepartmentForm form = new DepartmentForm();

form.ShowDialog();

}

/// <summary>

/// 修改部门

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 修改部门ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DataTable dt = dgvExclude.DataSource as DataTable;

if (dt != null)

{

var Form = new DepartmentForm();

Form.RId = Convert.ToInt32(dt.Rows[dgvExclude.CurrentRow.Index]["EId"]);

Form.iDo = 1;

Form.ShowDialog();

GetExclude();

}

}

/// <summary>

/// 删除部门

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 删除部门ToolStripMenuItem\_Click(object sender, EventArgs e)

{

MessageBoxButtons mess = MessageBoxButtons.OKCancel;

DialogResult dr = MessageBox.Show("确定要删除该部门吗，删除之后，该规则将在下次未回合同扣款计算时生效？", "提示", mess);

if (dr == DialogResult.OK)

{

DataTable dt = dgvExclude.DataSource as DataTable;

if (dt != null)

{

int RId = Convert.ToInt32(dt.Rows[dgvExclude.CurrentRow.Index]["EId"]);

MessageBox.Show(fSOServiceClient.DelDedcutNotReturnExclude(RId), "提示", MessageBoxButtons.OK);

GetExclude();

}

}

}

//加载

private void DedcutNotReturnExcludeForm\_Load(object sender, EventArgs e)

{

GetExclude();

}

//关闭窗体

private void DedcutNotReturnExcludeForm\_FormClosed(object sender, FormClosedEventArgs e)

{

instance = null;

Dispose();

}

/// <summary>

/// 查询按钮

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSearch\_Click(object sender, EventArgs e)

{

GetExclude();

}

/// <summary>

/// 清空按钮

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnClear\_Click(object sender, EventArgs e)

{

txtDept.Text = string.Empty;

}

#endregion

#region 公共方法

/// <summary>

/// 创建窗体

/// </summary>

/// <returns></returns>

public static DedcutNotReturnExcludeForm CreateForm()

{

if (instance == null)

{

instance = new DedcutNotReturnExcludeForm();

}

return instance;

}

#endregion

#region 私有方法

/// <summary>

/// 获取排除部门信息

/// </summary>

private void GetExclude()

{

string DeptId = txtDept.Text;

dgvExclude.DataSource = fSOServiceClient.GetDedcutNotReturnExclude(DeptId);

}

#endregion

}

public partial class DeductNoReturnAddRuleForm : Form

{

#region 公共成员

public DateTime DefaultDate;

#endregion

#region 私有成员

private string UserId = Global.UserId;

private IFSOService fSOServiceClient = Global.FSOService;

private string FileName = string.Empty;

private int FileId = 0;

#endregion

public DeductNoReturnAddRuleForm()

{

InitializeComponent();

}

#region 私有事件

/// <summary>

/// 加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnRuleForm\_Load(object sender, EventArgs e)

{

txtSalesId.HostAddress = Global.LinkService.HostAddress;

//txtContractNo.Text = ContractNo;

var listItem = new List<KeyValuePair<int, string>>();

listItem.Add(new KeyValuePair<int, string>(2, "特批不扣"));

listItem.Add(new KeyValuePair<int, string>(3, "其他免扣"));

//listItem.Add(new KeyValuePair<int, string>(3, "特殊确认"));

listItem.Add(new KeyValuePair<int, string>(4, "永久赦免"));

txtRule.DataSource = listItem;

txtRule.DisplayMember = "Value";

txtRule.ValueMember = "Key";

txtRule.SelectedIndex = -1;

txtRule.SelectedIndexChanged += TxtRule\_SelectedIndexChanged;

DateBegin.Value = DateEnd.Value = DefaultDate;

}

private void TxtRule\_SelectedIndexChanged(object sender, EventArgs e)

{

if (txtRule.SelectedIndex == 0|| txtRule.SelectedIndex == 1)

{

txtSalesId.Enabled = true;

DateBegin.Enabled = true;

DateEnd.Enabled = true;

}

else

{

txtSalesId.Enabled = false;

DateBegin.Enabled = false;

DateEnd.Enabled = false;

}

}

/// <summary>

/// 确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSubmit\_Click(object sender, EventArgs e)

{

if (CheckData())

{

DateTime dateBegin = DateBegin.Value;

DateTime dateEnd = DateEnd.Value;

int DeductType = Convert.ToInt32(txtRule.SelectedValue);

string ContractNo = txtContractNo.Text;

string SalesVPId = txtSalesId.OperatorId;

string Remark = txtRemark.Text;

string Result=fSOServiceClient.addDeductRule(ContractNo, DeductType, dateBegin, dateEnd, UserId, FileId, FileName, SalesVPId, Remark);

MessageBox.Show(Result, "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

DialogResult = DialogResult.OK;

Close();

}

else

{

MessageBox.Show("数据格式有问题，请检查合同号,VP经理,扣款规则是否选填", "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

/// <summary>

/// 取消

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

/// <summary>

/// 选择合同号

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void button1\_Click(object sender, EventArgs e)

{

string contractId = string.Empty;

string contractNo = string.Empty;

SelectContractForm.Select(out contractId, out contractNo);

txtContractNo.Text = contractNo;

}

/// <summary>

/// 上传附件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSupport\_Click(object sender, EventArgs e)

{

OpenFileDialog fileDialog = new OpenFileDialog();

fileDialog.Title = "请选择文件";

if (fileDialog.ShowDialog() == DialogResult.OK)

{

FileName = fileDialog.FileName;//返回文件的完整路径

FileId = Global.BaseOperate.FSOFileUpload("DedcutNotReturn", FileName, "规则设置");

txtFileName.Text = FileName;

}

}

#endregion

#region 私有方法

/// <summary>

/// 检查数据

/// </summary>

/// <returns></returns>

private bool CheckData()

{

bool result = true;

if (string.IsNullOrEmpty(txtContractNo.Text))

result = false;

if (txtRule.SelectedIndex == -1)

result = false;

if (txtRule.SelectedIndex == 0 && string.IsNullOrEmpty(txtSalesId.OperatorId))

result = false;

return result;

}

#endregion

}

public partial class DeductNoReturnDetailsAddForm : Form

{

public DeductNoReturnDetailsAddForm()

{

InitializeComponent();

}

#region 公共成员

public int DetailsId = 0;

public int DNRId = 0;

public string SalesId = string.Empty;

public int DeductType = 1;

public double DeductAmount = 0;

#endregion

#region 私有成员

private string UserId = Global.UserId;

private IFSOService fSOServiceClient = Global.FSOService;

#endregion

#region 私有事件

/// <summary>

/// 画面加载事件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnDetailsAddForm\_Load(object sender, EventArgs e)

{

txtSalesId.HostAddress = Global.LinkService.HostAddress;

var listItem = new List<KeyValuePair<int, string>>();

listItem.Add(new KeyValuePair<int, string>(0, "暂扣"));

listItem.Add(new KeyValuePair<int, string>(1, "必扣不退"));

listItem.Add(new KeyValuePair<int, string>(2, "特批不扣"));

//listItem.Add(new KeyValuePair<int, string>(3, "特殊确认"));

listItem.Add(new KeyValuePair<int, string>(4, "永久赦免"));

txtDeductType.DataSource = listItem;

txtDeductType.DisplayMember = "Value";

txtDeductType.ValueMember = "Key";

txtDeductType.SelectedIndex = 1;

txtDeductType.SelectedIndex = DeductType;

if (!string.IsNullOrEmpty(SalesId))

{

txtSalesId.OperatorId = SalesId;

txtDeductAmount.Text = DeductAmount.ToString("0.00");

}

}

/// <summary>

/// 确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSubmit\_Click(object sender, EventArgs e)

{

if (CheckData())

{

DeductNoReturnDetailsModel model = new DeductNoReturnDetailsModel();

model.UserId = UserId;

model.DNRId = DNRId;

model.Id = DetailsId;

model.IsInput = false;

model.SalesId = txtSalesId.OperatorId;

model.DeductAmount = Convert.ToDouble(txtDeductAmount.Text);

model.DeductType = Convert.ToInt32(txtDeductType.SelectedValue);

if (DetailsId > 0)

fSOServiceClient.AddorEditDeductNoReturn(model, 1);

else

fSOServiceClient.AddorEditDeductNoReturn(model, 0);

MessageBox.Show("成功！", "提示");

DialogResult = DialogResult.OK;

Close();

}

else

{

MessageBox.Show("数据格式有问题，请检查", "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

/// <summary>

/// 取消

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

#endregion

#region 私有方法

/// <summary>

/// 检查数据

/// </summary>

/// <returns></returns>

private bool CheckData()

{

bool result = true;

if (string.IsNullOrEmpty(txtSalesId.OperatorId))

result = false;

if (txtDeductType.SelectedIndex == -1)

result = false;

try

{

int var1 = Convert.ToInt32(txtDeductAmount.Text);

}

catch

{

result = false;

}

return result;

}

#endregion

}

public partial class DeductNoReturnDetailsForm : Form

{

#region 公共成员

public int DNRId = 0;

#endregion

#region 私有成员

//private static DeductNoReturnForm instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

private int DeductType = -1;

#endregion

public DeductNoReturnDetailsForm()

{

InitializeComponent();

DgvDeductDetails.AutoGenerateColumns = false;

}

#region 私有事件

/// <summary>

/// 页面加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnDetailsForm\_Load(object sender, EventArgs e)

{

GetDetails();

}

/// <summary>

/// 添加调整金额

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 添加ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DeductNoReturnDetailsAddForm form = new DeductNoReturnDetailsAddForm();

form.DNRId = DNRId;

form.DeductType = DeductType;//默认扣款规则

form.ShowDialog();

if(form.DialogResult==DialogResult.OK)

GetDetails();

}

/// <summary>

/// 编辑扣款详情

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 修改ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DataTable dt = DgvDeductDetails.DataSource as DataTable;

if (!(bool)dt.Rows[DgvDeductDetails.CurrentRow.Index]["IsInput"])

{

DeductNoReturnDetailsAddForm form = new DeductNoReturnDetailsAddForm();

form.DNRId = DNRId;

form.DetailsId = Convert.ToInt32(dt.Rows[DgvDeductDetails.CurrentRow.Index]["Id"]);

form.SalesId = dt.Rows[DgvDeductDetails.CurrentRow.Index]["SalesId"].ToString();

form.DeductType= Convert.ToInt32(dt.Rows[DgvDeductDetails.CurrentRow.Index]["DeductType"]);

form.DeductAmount = Convert.ToDouble(dt.Rows[DgvDeductDetails.CurrentRow.Index]["DeductAmount"]);

form.Text = "编辑调整金额";

form.ShowDialog();

GetDetails();

}

else

{

MessageBox.Show("系统自动录入的数据不可修改!", "提示");

}

}

/// <summary>

/// 删除详情

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 删除ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DataTable dt = DgvDeductDetails.DataSource as DataTable;

if (!(bool)dt.Rows[DgvDeductDetails.CurrentRow.Index]["IsInput"])

{

int DeleteId = Convert.ToInt32(dt.Rows[DgvDeductDetails.CurrentRow.Index]["Id"]);

fSOServiceClient.DeletdNoReturnDetails(DeleteId);

GetDetails();

}

else

{

MessageBox.Show("系统自动录入的数据不可修改!", "提示");

}

}

//关闭窗体

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

#endregion

#region 私有方法

/// <summary>

/// 获取详情信息

/// </summary>

private void GetDetails()

{

DgvDeductDetails.DataSource = fSOServiceClient.GetDeductNoReturnDetail(DNRId.ToString());

if (((DataTable)DgvDeductDetails.DataSource).Rows.Count > 0)

{

DeductType = (Int32)DgvDeductDetails.Rows[0].Cells["Column11"].Value;

}

}

#endregion

}

public partial class DeductNoReturnForm : Form

{

#region 私有成员

private static DeductNoReturnForm instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

const string AStateName = "待审核"; //审核判断

#endregion

public DeductNoReturnForm()

{

InitializeComponent();

SetRight();

dgvDeduct.CellContentDoubleClick += DgvDeduct\_CellContentDoubleClick;

dgvDeduct.CellContentClick += DgvDeduct\_CellContentClick;

DgvCalDate.AutoGenerateColumns = false;

dgvDeduct.AutoGenerateColumns = false;

dgvDeductRule.AutoGenerateColumns = false;

DgvDeductDetails.AutoGenerateColumns = false;

dgvDeductTotal.AutoGenerateColumns = false;

dgvPubTran.AutoGenerateColumns = false;

//dgvDeductNoReturnAll.AutoGenerateColumns = false;

}

#region 公共方法

/// <summary>

/// 创建窗体

/// </summary>

/// <returns></returns>

public static DeductNoReturnForm CreateForm()

{

if (instance == null)

{

instance = new DeductNoReturnForm();

}

return instance;

}

#endregion

#region 私有事件

/// <summary>

/// 画面加载发生事件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnForm\_Load(object sender, EventArgs e)

{

txtSalesId.HostAddress = Global.LinkService.HostAddress;

GetCalDateList();

GetCalDate();

GetDeductRuleList();

}

/// <summary>

/// 查询数据

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void BtnSearch\_Click(object sender, EventArgs e)

{

if (TabControl1.SelectedIndex == 0)

GetDeductNoReturn();

else if (TabControl1.SelectedIndex == 1)

GetDeductRule();

else if (TabControl1.SelectedIndex == 2)

GetDeductNoReturnTotal();

else if (TabControl1.SelectedIndex == 3)

GetPubTran();

}

/// <summary>

/// 清空按钮

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void BtnClear\_Click(object sender, EventArgs e)

{

txtContractNo.Text = string.Empty;

txtSalesId.OperatorId = string.Empty;

txtCalDate.SelectedIndex = 0;

txtCalEndDate.SelectedIndex = 0;

}

/// <summary>

/// 添加期间

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void ToolStripMenuItem1\_Click(object sender, EventArgs e)

{

DateTime NewDate = Convert.ToDateTime(fSOServiceClient.GetCalDate().Rows[0]["FDATE"]).AddMonths(1);

AddCalDateForm form = new AddCalDateForm();

form.NewDate = NewDate;

form.ShowDialog();

GetCalDate();

GetCalDateList();

}

/// <summary>

/// 删除期间

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void ToolStripMenuItem2\_Click(object sender, EventArgs e)

{

MessageBoxButtons mess = MessageBoxButtons.OKCancel;

DialogResult dr = MessageBox.Show("确定要删除该期间及相关数据吗？", "提示", mess);

if (dr == DialogResult.OK)

{

DgvCalDate.EndEdit();

DataTable dt = DgvCalDate.DataSource as DataTable;

var PId = Convert.ToInt32(dt.Rows[DgvCalDate.CurrentRow.Index]["PId"]);

MessageBox.Show(fSOServiceClient.DeleteDeductNoReturnCalDate(PId, UserId), "提示", MessageBoxButtons.OK);

GetCalDate();

GetCalDateList();

}

}

/// <summary>

/// 计算获取扣款基数

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCalAmount\_Click(object sender, EventArgs e)

{

DataTable dt = DgvCalDate.DataSource as DataTable;

if (dt.Rows.Count > 0)

{

WaitFormService.CreateWaitForm();

int PId = Convert.ToInt32(dt.Rows[0]["PId"]);

string result = fSOServiceClient.CalcluDeductNoReturn(PId, UserId);

WaitFormService.CloseWaitForm();

MessageBox.Show(result, "提示", MessageBoxButtons.OK);

GetDeductNoReturn();

}

}

/// <summary>

/// 双击获取未回合同扣款详情页

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DgvDeduct\_CellContentDoubleClick(object sender, DataGridViewCellEventArgs e)

{

EditDeductNoReturn();

}

/// <summary>

/// 右键获取未回合同扣款详情页

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 调整扣款金额ToolStripMenuItem\_Click(object sender, EventArgs e)

{

EditDeductNoReturn();

}

/// <summary>

/// 右键设置该合同扣款规则

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 修改扣款规则ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DataTable dt = dgvDeduct.DataSource as DataTable;

var Form = new DeductNoReturnRuleForm();

Form.DefaultDate = Convert.ToDateTime(fSOServiceClient.GetCalDate().Rows[0]["FDATE"]);

Form.ContractNo = dgvDeduct.CurrentRow.Cells["Column14"].Value.ToString();

Form.ShowDialog();

}

/// <summary>

/// 扣款规则修改历史获取

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 扣款规则修改历史ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DataTable dt = dgvDeduct.DataSource as DataTable;

var Form = new DeductNoReturnRuleHistoryForm();

Form.ContractNo = dt.Rows[dgvDeduct.CurrentRow.Index]["MContractNo"].ToString();

Form.ShowDialog();

}

/// <summary>

/// 删除设置的规则

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 删除ToolStripMenuItem\_Click(object sender, EventArgs e)

{

MessageBoxButtons mess = MessageBoxButtons.OKCancel;

DialogResult dr = MessageBox.Show("确定要删除该规则设置吗，删除之后，该规则将在下次未回合同扣款计算时生效？", "提示", mess);

if (dr == DialogResult.OK)

{

DgvCalDate.EndEdit();

var CId = dgvDeductRule.CurrentRow.Cells["Column46"].Value.ToString();

fSOServiceClient.DeleteDeductNoReturnRule(CId);

GetDeductRule();

}

}

/// <summary>

/// 审核

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnAudit\_Click(object sender, EventArgs e)

{

DeductNoReturnAudit(0);

}

/// <summary>

/// 审核确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSAudit\_Click(object sender, EventArgs e)

{

DeductNoReturnAudit(1);

}

/// <summary>

/// 反审

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnAntiAudit\_Click(object sender, EventArgs e)

{

DeductNoReturnAudit(9);

}

/// <summary>

/// 导出

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnExport\_Click(object sender, EventArgs e)

{

SaveFileDialog sfd = new SaveFileDialog();

sfd.Filter = "\*.xls|\*.xls";

if (sfd.ShowDialog() == DialogResult.OK)

{

bool Result = false;

if (TabControl1.SelectedIndex == 0)

Result = DataGridViewExportExcelClass.SaveXls(sfd.FileName, dgvDeduct);

else if (TabControl1.SelectedIndex == 1)

Result = DataGridViewExportExcelClass.SaveXls(sfd.FileName, dgvDeductRule);

else if (TabControl1.SelectedIndex == 2)

Result = DataGridViewExportExcelClass.SaveXls(sfd.FileName, dgvDeductTotal);

if (Result)

System.Diagnostics.Process.Start(sfd.FileName);

}

}

/// <summary>

/// 添加扣款规则

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 添加ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DeductNoReturnAddRuleForm form = new DeductNoReturnAddRuleForm();

form.DefaultDate = Convert.ToDateTime(fSOServiceClient.GetCalDate().Rows[0]["FDATE"]);

form.ShowDialog();

if (form.DialogResult == DialogResult.OK)

GetDeductRule();

}

/// <summary>

/// 单击某行获得详情信息

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DgvDeduct\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

if (dgvDeduct.CurrentRow != null)

{

DataTable dt = dgvDeduct.DataSource as DataTable;

if (dt != null)

{

string DNRId = dgvDeduct.CurrentRow.Cells["Column12"].Value.ToString();

DgvDeductDetails.DataSource = fSOServiceClient.GetDeductNoReturnDetail(DNRId);

}

}

}

/// <summary>

/// 关闭窗口

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnForm\_FormClosed(object sender, FormClosedEventArgs e)

{

instance = null;

Dispose();

}

/// <summary>

/// 读取上传文件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 读取上传文件ToolStripMenuItem\_Click(object sender, EventArgs e)

{

//ReadFiles();

DeductNoReturnRuleFile Form = new DeductNoReturnRuleFile();

DataTable dt = dgvDeductRule.DataSource as DataTable;

if (dt != null)

{

var CId = dgvDeductRule.CurrentRow.Cells["Column46"].Value.ToString();

Form.CId = CId;

}

Form.ShowDialog();

}

/// <summary>

/// 读取文件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void dgvDeductRule\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewColumn column = dgvDeductRule.Columns[e.ColumnIndex];

if (column is DataGridViewButtonColumn)

{

ReadFiles();

}

}

}

/// <summary>

/// 获取规则详情

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnGetRules\_Click(object sender, EventArgs e)

{

DataTable dt = dgvDeduct.DataSource as DataTable;

if (dt != null)

{

DeductNoReturnSelectModel model = new DeductNoReturnSelectModel();

model.MyId = UserId;

model.MContractNo = dgvDeduct.CurrentRow.Cells["Column14"].Value.ToString();

model.PId = Convert.ToInt32(txtCalDate.SelectedValue);

TabControl1.SelectTab(1);

WaitFormService.CreateWaitForm();

dgvDeductRule.DataSource = fSOServiceClient.GetDeductRuleList(model);

WaitFormService.CloseWaitForm();

}

}

/// <summary>

/// 选项卡切换

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void tabControl1\_SelectedIndexChanged(object sender, EventArgs e)

{

if (TabControl1.SelectedIndex == 0)

{

txtBrand.Enabled = txtCustomerName.Enabled = txtSignAName.Enabled = ck1.Enabled = btnCalAmount.Enabled = btnCalAmountR.Enabled = ckIsZero.Enabled = true;

txtCalEndDate.Enabled = true;

btnGetRules.Enabled = true;

txtRule.Enabled = false;

txtContractNo.Enabled = true;

txtSalesId.Enabled = true;

ckbj.Enabled = cksh.Enabled = ckall.Enabled = true;

btnFAudit.Enabled = btnAntiAudit.Enabled = btnAddFile.Enabled = true;

}

else if (TabControl1.SelectedIndex == 1)

{

txtBrand.Enabled = txtCustomerName.Enabled = txtSignAName.Enabled = ck1.Enabled = btnCalAmount.Enabled = btnCalAmountR.Enabled = ckIsZero.Enabled = false;

txtCalEndDate.Enabled = false;

btnGetRules.Enabled = false;

txtRule.Enabled = true;

txtContractNo.Enabled = true;

txtSalesId.Enabled = true;

ckbj.Enabled = cksh.Enabled = ckall.Enabled = false;

btnFAudit.Enabled = btnAntiAudit.Enabled = btnAddFile.Enabled = false;

}

else if (TabControl1.SelectedIndex == 2)

{

txtBrand.Enabled = txtCustomerName.Enabled = txtSignAName.Enabled = ck1.Enabled = btnCalAmount.Enabled = btnCalAmountR.Enabled = ckIsZero.Enabled = false;

txtCalEndDate.Enabled = false;

btnGetRules.Enabled = false;

txtRule.Enabled = false;

txtContractNo.Enabled = false;

txtSalesId.Enabled = false;

ckbj.Enabled = cksh.Enabled = ckall.Enabled = true;

btnFAudit.Enabled = btnAntiAudit.Enabled = btnAddFile.Enabled = false;

}

else if (TabControl1.SelectedIndex == 3)

{

txtBrand.Enabled = txtCustomerName.Enabled = txtSignAName.Enabled = ck1.Enabled = btnCalAmount.Enabled = btnCalAmountR.Enabled = ckIsZero.Enabled = false;

txtCalEndDate.Enabled = false;

btnGetRules.Enabled = false;

txtRule.Enabled = false;

txtContractNo.Enabled = true;

txtSalesId.Enabled = false;

ckbj.Enabled = cksh.Enabled = ckall.Enabled = false;

btnFAudit.Enabled = btnAntiAudit.Enabled = btnAddFile.Enabled = false;

}

}

/// <summary>

/// 未回合同退款

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 设置回款金额ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DeductReturnAddForm form = new DeductReturnAddForm();

form.ShowDialog();

}

/// <summary>

/// 对当前结果进行计算

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void txtCalAmountR\_Click(object sender, EventArgs e)

{

DataTable dt = DgvCalDate.DataSource as DataTable;

if (dt.Rows.Count > 0)

{

int PId = Convert.ToInt32(dt.Rows[0]["PId"]);

if (PId != Convert.ToInt32(txtCalDate.SelectedValue) || PId != Convert.ToInt32(txtCalEndDate.SelectedValue))

{

MessageBox.Show("计算期间选择错误！", "提示", MessageBoxButtons.OK);

}

else

{

DataGridView dgv = dgvDeduct;

if (dgv.Rows.Count > 0)

{

string ContractNos = string.Empty;

foreach (DataGridViewRow i in dgv.Rows)

{

ContractNos += i.Cells["Column14"].Value.ToString() + ",";

}

string result = fSOServiceClient.CalcluDeductNoReturnR(PId, UserId, ContractNos.Substring(0, ContractNos.Length - 1));

WaitFormService.CloseWaitForm();

MessageBox.Show(result, "提示", MessageBoxButtons.OK);

GetDeductNoReturn();

}

}

}

}

/// <summary>

/// 批量上传审核文件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnAddFile\_Click(object sender, EventArgs e)

{

string McontractNos = string.Empty;

DataTable dt = dgvDeduct.DataSource as DataTable;

if (dt != null)

{

//获取所有被选中的合同号

int count = Convert.ToInt16(this.dgvDeduct.Rows.Count.ToString());

for (int i = 0; i < count; i++)

{

DataGridViewCheckBoxCell checkCell = (DataGridViewCheckBoxCell)dgvDeduct.Rows[i].Cells["choose"];

Boolean flag = Convert.ToBoolean(checkCell.Value);

if (flag == true) //查找被选择的数据行

{

checkCell.Value = false;

McontractNos += dgvDeduct.Rows[i].Cells["Column14"].Value + ",";

}

}

McontractNos = McontractNos.Substring(0, McontractNos.Length - 1);

//上传文件

OpenFileDialog fileDialog = new OpenFileDialog();

fileDialog.Title = "请选择文件";

if (fileDialog.ShowDialog() == DialogResult.OK)

{

string FileName = fileDialog.FileName;//返回文件的完整路径

int FileId = Global.BaseOperate.FSOFileUpload("DedcutNotReturn", FileName, "规则设置");

fSOServiceClient.BatchAddDedcutNotReturnConfigFile(McontractNos, FileId, FileName, UserId);

}

}

}

//发布转移合同调整

private void 发布转移合同调整ToolStripMenuItem\_Click(object sender, EventArgs e)

{

DataTable dt = dgvPubTran.DataSource as DataTable;

if (dt != null)

{

string OMcontractNo = dgvPubTran.CurrentRow.Cells["Column11"].Value.ToString();

if (!string.IsNullOrEmpty(OMcontractNo))

{

DeductNotReturnPubRelaForm form = new DeductNotReturnPubRelaForm();

form.OContractNo = OMcontractNo;

form.MContractNo = dgvPubTran.CurrentRow.Cells["Column10"].Value.ToString();

form.PId = "-1";

form.ShowDialog();

dgvPubTran.DataSource = fSOServiceClient.GetPubTran(txtContractNo.Text);

}

else

{

MessageBox.Show("该合同不可调整", "提示", MessageBoxButtons.OK);

}

}

}

/// <summary>

/// 修改VP经理

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 修改VP经理ToolStripMenuItem\_Click(object sender, EventArgs e)

{

MessageBoxButtons mess = MessageBoxButtons.OKCancel;

DialogResult dr = MessageBox.Show("是否确定要修改VP经理！", "提示", mess);

if (dr == DialogResult.OK)

{

DgvCalDate.EndEdit();

var CId = dgvDeductRule.CurrentRow.Cells["Column46"].Value.ToString();

DeductNoReturnSalesVPEdit form = new DeductNoReturnSalesVPEdit();

form.CId = CId;

form.ShowDialog();

GetDeductRule();

}

}

#endregion

#region 私有方法

/// <summary>

/// 获取发布转移列表

/// </summary>

private void GetPubTran()

{

dgvPubTran.DataSource= fSOServiceClient.GetPubTran(txtContractNo.Text);

}

/// <summary>

/// 根据销售员获取当月的扣款金额总合

/// </summary>

private void GetDeductNoReturnTotal()

{

int PId = Convert.ToInt32(txtCalDate.SelectedValue);

string DataCenter = GetDataCenter();

DataTable dt = fSOServiceClient.GetDeductNoReturnTotal(PId,DataCenter);

dgvDeductTotal.DataSource = dt;

}

/// <summary>

/// 获取数据中心条件

/// </summary>

/// <returns></returns>

private string GetDataCenter()

{

string DataCenter = string.Empty;

if(ckall.Checked)

DataCenter = "";

else if(ckbj.Checked)

DataCenter = "北京数据中心";

else if (cksh.Checked)

DataCenter = "上海数据中心";

return DataCenter;

}

/// <summary>

/// 获取未回合同扣款主表信息

/// </summary>

private void GetDeductNoReturn()

{

string DataCenter = GetDataCenter();

DeductNoReturnSelectModel model = new DeductNoReturnSelectModel();

model.MyId = UserId;

model.SalesId = txtSalesId.OperatorId;

model.MContractNo = txtContractNo.Text;

model.PId = Convert.ToInt32(txtCalDate.SelectedValue);

model.EndPId = Convert.ToInt32(txtCalEndDate.SelectedValue);

model.DataCenter = DataCenter;

model.SignAName = txtSignAName.Text;

model.Brand = txtBrand.Text;

model.CustomerName = txtCustomerName.Text;

model.SalesIsnull = ck1.Checked;

model.IsDeDuctZero = ckIsZero.Checked;

WaitFormService.CreateWaitForm();

DataTable dt = fSOServiceClient.GetDeductNoReturn(model);

dgvDeduct.DataSource = dt;

DgvDeductDetails.DataSource = new DataTable();

WaitFormService.CloseWaitForm();

}

/// <summary>

/// 获取所有计算期间未回合同扣款主表信息

/// </summary>

private void GetDeductNoReturnAll()

{

string DataCenter = GetDataCenter();

DeductNoReturnSelectModel model = new DeductNoReturnSelectModel();

model.MyId = UserId;

model.SalesId = txtSalesId.OperatorId;

model.MContractNo = txtContractNo.Text;

model.PId = Convert.ToInt32(txtCalDate.SelectedValue);

model.DataCenter = DataCenter;

WaitFormService.CreateWaitForm();

DataTable dt = fSOServiceClient.GetDeductNoReturnAll(model);

//dgvDeductNoReturnAll.DataSource = dt;

WaitFormService.CloseWaitForm();

}

/// <summary>

/// 获取未回合同扣款期间

/// </summary>

private void GetCalDate()

{

DataTable dt = fSOServiceClient.GetCalDate();

DgvCalDate.DataSource = dt;

}

/// <summary>

/// 获取未回合同计算期间列表COMBOX数据绑定

/// </summary>

private void GetCalDateList()

{

IList<QueryDictionaryModel> CalDateList = fSOServiceClient.GetCalDateList();

txtCalDate.DataSource = CalDateList;

txtCalDate.ValueMember = "Key";

txtCalDate.DisplayMember = "Value";

txtCalDate.SelectedIndex = 0;

IList<QueryDictionaryModel> CalDateList2 = fSOServiceClient.GetCalDateList();

txtCalEndDate.DataSource = CalDateList2;

txtCalEndDate.ValueMember = "Key";

txtCalEndDate.DisplayMember = "Value";

txtCalEndDate.SelectedIndex = 0;

}

/// <summary>

/// 未回合同详情

/// </summary>

private void EditDeductNoReturn()

{

DataTable dt = dgvDeduct.DataSource as DataTable;

if (dt != null)

{

if (dgvDeduct.CurrentRow.Cells["Column2"].Value.ToString() == AStateName)

{

var Form = new DeductNoReturnDetailsForm();

Form.DNRId = Convert.ToInt32(dgvDeduct.CurrentRow.Cells["Column12"].Value);

Form.ShowDialog();

}

else

{

MessageBox.Show("审核状态的数据无法编辑!", "提示", MessageBoxButtons.OK);

}

}

}

/// <summary>

/// 获取未回合同规则表

/// </summary>

private void GetDeductRule()

{

DeductNoReturnSelectModel model = new DeductNoReturnSelectModel();

model.MyId = UserId;

//model.SalesId = txtSalesId.OperatorId;

model.MContractNo = txtContractNo.Text;

model.PId = Convert.ToInt32(txtCalDate.SelectedValue);

model.DeductRuleType= Convert.ToInt32(txtRule.SelectedValue);

WaitFormService.CreateWaitForm();

dgvDeductRule.DataSource = fSOServiceClient.GetDeductRuleList(model);

WaitFormService.CloseWaitForm();

}

/// <summary>

/// 未回合同扣款审核

/// </summary>

/// <param name="iDo"></param>

/// <returns></returns>

private void DeductNoReturnAudit(int iDo)

{

DataTable dt = DgvCalDate.DataSource as DataTable;

if (dt != null)

{

var PId = Convert.ToInt32(dt.Rows[DgvCalDate.CurrentRow.Index]["PId"]);

var userId = UserId;

MessageBox.Show(fSOServiceClient.DecudtNoReturnAudit(PId, userId, iDo), "提示", MessageBoxButtons.OK);

GetDeductNoReturn();

}

}

/// <summary>

/// 读取文件

/// </summary>

private void ReadFiles()

{

DataTable dt = dgvDeductRule.DataSource as DataTable;

if (dt != null)

{

var FileId = dt.Rows[dgvDeductRule.CurrentRow.Index]["FileId"].ToString();

if (!string.IsNullOrEmpty(FileId))

{

if (Convert.ToInt32(FileId) > 0)

{

var FilePath = Global.BaseOperate.FSOFileRead(Convert.ToInt32(FileId));

System.Diagnostics.Process.Start(FilePath);

}

else

{

MessageBox.Show("没有上传文件", "提示", MessageBoxButtons.OK);

}

}

else

{

MessageBox.Show("没有上传文件", "提示", MessageBoxButtons.OK);

}

}

}

/// <summary>

/// 获取规则查询列表及数据中心列表

/// </summary>

private void GetDeductRuleList()

{

var listItem = new List<KeyValuePair<int, string>>();

listItem.Add(new KeyValuePair<int, string>(0, ""));

listItem.Add(new KeyValuePair<int, string>(2, "特批不扣"));

listItem.Add(new KeyValuePair<int, string>(3, "其他免扣"));

listItem.Add(new KeyValuePair<int, string>(4, "永久赦免"));

txtRule.DataSource = listItem;

txtRule.DisplayMember = "Value";

txtRule.ValueMember = "Key";

txtRule.SelectedIndex = -1;

//var listItem2 = new List<KeyValuePair<int, string>>();

//listItem2.Add(new KeyValuePair<int, string>(0, ""));

//listItem2.Add(new KeyValuePair<int, string>(1, "北京数据中心"));

//listItem2.Add(new KeyValuePair<int, string>(2, "上海数据中心"));

//txtDataCenter.DataSource = listItem2;

//txtDataCenter.DisplayMember = "Value";

//txtDataCenter.ValueMember = "Key";

//txtDataCenter.SelectedIndex = -1;

}

//查询权限设置

private void SetRight()

{

btnCalAmount.Enabled= 调整扣款金额ToolStripMenuItem.Enabled = Global.BaseOperate.CheckFuncRights(24230104);

BtnSearch.Enabled = Global.BaseOperate.CheckFuncRights(24230101);

btnFAudit.Enabled = Global.BaseOperate.CheckFuncRights(24230102);

btnSAudit.Enabled = Global.BaseOperate.CheckFuncRights(24230103);

修改扣款规则ToolStripMenuItem.Enabled= Global.BaseOperate.CheckFuncRights(24230111);

添加ToolStripMenuItem.Enabled= Global.BaseOperate.CheckFuncRights(24230111);

}

#endregion

}

public partial class DeductNoReturnRefundForm : Form

{

#region 私有成员

private static DeductNoReturnRefundForm instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

#region 公共方法

/// <summary>

/// 创建窗体

/// </summary>

/// <returns></returns>

public static DeductNoReturnRefundForm CreateForm()

{

if (instance == null)

{

instance = new DeductNoReturnRefundForm();

}

return instance;

}

public DeductNoReturnRefundForm()

{

InitializeComponent();

SetRight();

dgvReturn.AutoGenerateColumns = false;

dgvReturnHistory.AutoGenerateColumns = false;

dgvReturnDetails.AutoGenerateColumns = false;

}

#endregion

#region 私有事件

/// <summary>

/// 页面加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnRefundForm\_Load(object sender, EventArgs e)

{

txtSalesId.HostAddress = Global.LinkService.HostAddress;

DateCalDate.Value=DateBegin.Value=DateEnd.Value= Convert.ToDateTime(fSOServiceClient.GetCalDate().Rows[0]["FDATE"]);

}

/// <summary>

/// 查询

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btSearch\_Click(object sender, EventArgs e)

{

if (tabControl1.SelectedIndex == 0)

{

GetReturn();

}

else if (tabControl1.SelectedIndex == 1)

{

GetReturnHistory();

}

else if (tabControl1.SelectedIndex == 2)

{

GetReturnTotalHistory();

}

}

/// <summary>

/// 获取退款详情

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void dgvReturn\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

if (dgvReturn.CurrentRow != null)

{

DataTable dt = dgvReturn.DataSource as DataTable;

if (dt != null)

{

string McontractNo = dgvReturn.CurrentRow.Cells["Column37"].Value.ToString();

string SalesId = dgvReturn.CurrentRow.Cells["Column38"].Value.ToString();

dgvReturnDetails.DataSource = fSOServiceClient.GetRefundDetail(McontractNo, SalesId);

}

}

}

/// <summary>

/// 未回合同退款点击

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnReturn\_Click(object sender, EventArgs e)

{

DataTable dt = dgvReturn.DataSource as DataTable;

if (dt != null)

{

string McontractNo = dgvReturn.CurrentRow.Cells["Column37"].Value.ToString();

DeductReturnAddForm form = new DeductReturnAddForm();

form.ContractNo = McontractNo;

form.ShowDialog();

}

}

/// <summary>

/// 读取文件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void dgvReturnHistory\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewColumn column = dgvReturnHistory.Columns[e.ColumnIndex];

if (column is DataGridViewButtonColumn)

{

ReadFiles();

}

}

}

/// <summary>

/// 清空

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnClear\_Click(object sender, EventArgs e)

{

txtContractNo.Text = string.Empty;

txtSalesId.OperatorId = string.Empty;

}

/// <summary>

/// 选项卡切换

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void tabControl1\_SelectedIndexChanged(object sender, EventArgs e)

{

if (tabControl1.SelectedIndex == 0)

{

txtSignAName.Enabled = txtCustomerName.Enabled = txtBrand.Enabled = true;

btnReturn.Enabled = true;

btnExport.Enabled = true;

btnAudit.Enabled = false;

btnAntiAudit.Enabled = false;

btnCAll.Enabled = btnCNone.Enabled = false;

DateCalDate.Enabled = false;

txtContractNo.Enabled = true;

txtSalesId.Enabled = true;

DateBegin.Enabled = DateEnd.Enabled = false;

ck1.Enabled =ckHistory.Enabled =ckMust.Enabled = true;

}

else if (tabControl1.SelectedIndex == 1)

{

txtSignAName.Enabled = txtCustomerName.Enabled = txtBrand.Enabled = true;

btnReturn.Enabled = false;

btnExport.Enabled = true;

btnAudit.Enabled = true;

btnAntiAudit.Enabled = true;

btnCAll.Enabled = btnCNone.Enabled = true;

DateCalDate.Enabled = false;

txtContractNo.Enabled = true;

txtSalesId.Enabled = true;

DateBegin.Enabled = DateEnd.Enabled = true;

ck1.Enabled = ckHistory.Enabled = ckMust.Enabled = false;

}

else if (tabControl1.SelectedIndex == 2)

{

txtSignAName.Enabled = txtCustomerName.Enabled = txtBrand.Enabled = false;

btnReturn.Enabled = false;

btnExport.Enabled = true;

btnAudit.Enabled = false;

btnAntiAudit.Enabled = false;

btnCAll.Enabled = btnCNone.Enabled = false;

DateCalDate.Enabled = true;

txtContractNo.Enabled = false;

txtSalesId.Enabled = false;

DateBegin.Enabled = DateEnd.Enabled = ckMust.Enabled = false;

ck1.Enabled = ckHistory.Enabled = false;

}

}

/// <summary>

/// 导出报表

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnExport\_Click(object sender, EventArgs e)

{

SaveFileDialog sfd = new SaveFileDialog();

sfd.Filter = "\*.xls|\*.xls";

if (sfd.ShowDialog() == DialogResult.OK)

{

bool Result = false;

if(tabControl1.SelectedIndex==1)

Result = DataGridViewExportExcelClass.SaveXls(sfd.FileName, dgvReturnHistory);

else if(tabControl1.SelectedIndex == 2)

Result = DataGridViewExportExcelClass.SaveXls(sfd.FileName, dgvTotalReturnHstory);

else if (tabControl1.SelectedIndex == 0)

Result = DataGridViewExportExcelClass.SaveXls(sfd.FileName, dgvReturn);

if (Result)

System.Diagnostics.Process.Start(sfd.FileName);

}

}

/// <summary>

/// 判断审核状态

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void dgvReturnHistory\_CellClick(object sender, DataGridViewCellEventArgs e)

{

DataTable dt = dgvReturnHistory.DataSource as DataTable;

if (dt != null)

{

var State = Convert.ToInt32(dt.Rows[dgvReturnHistory.CurrentRow.Index]["AState"]);

switch (State)

{

case 0:

btnAudit.Enabled = true;

btnAntiAudit.Enabled = true;

break;

case 1:

btnAudit.Enabled = true;

btnAntiAudit.Enabled = true;

break;

case 2:

btnAudit.Enabled = true;

btnAntiAudit.Enabled = true;

break;

}

}

}

/// <summary>

/// 审核

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnAudit\_Click(object sender, EventArgs e)

{

var dr = MessageBox.Show("是否确认该退款信息过审！", "系统提示", MessageBoxButtons.YesNo,

MessageBoxIcon.Warning, MessageBoxDefaultButton.Button1);

if (dr == DialogResult.Yes)

{

AuditReturn(1);

GetReturnHistory();

}

}

/// <summary>

/// 反审

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnAntiAudit\_Click(object sender, EventArgs e)

{

var dr = MessageBox.Show("是否确认反审该退款信息！", "系统提示", MessageBoxButtons.YesNo,

MessageBoxIcon.Warning, MessageBoxDefaultButton.Button1);

if (dr == DialogResult.Yes)

{

AuditReturn(2);

GetReturnHistory();

}

}

/// <summary>

/// 删除退款记录

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void 删除ToolStripMenuItem\_Click(object sender, EventArgs e)

{

if (dgvReturnHistory.CurrentRow != null)

{

DataTable dt = dgvReturnHistory.DataSource as DataTable;

if (dt != null)

{

string RId = dt.Rows[dgvReturnHistory.CurrentRow.Index]["RId"].ToString();

MessageBox.Show(fSOServiceClient.DeleteRefund(RId), "系统提示", MessageBoxButtons.OK);

GetReturnHistory();

}

}

}

//关闭窗口

private void DeductNoReturnRefundForm\_FormClosed(object sender, FormClosedEventArgs e)

{

instance = null;

Dispose();

}

/// <summary>

/// 全选

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCAll\_Click(object sender, EventArgs e)

{

ChooseAll(1);

}

/// <summary>

/// 全不选

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCNone\_Click(object sender, EventArgs e)

{

ChooseAll(2);

}

#endregion

#region 私有方法

/// <summary>

/// 获取回款列表

/// </summary>

private void GetReturn()

{

string DataCenter = GetDataCenter();

DeductNoReturnSelectModel model = new DeductNoReturnSelectModel();

model.MContractNo = txtContractNo.Text;

model.SalesId = txtSalesId.OperatorId;

model.IsZero = ck1.Checked;

model.SignAName = txtSignAName.Text;

model.Brand = txtBrand.Text;

model.CustomerName = txtCustomerName.Text;

model.DataCenter = DataCenter;

model.MustDeduct = ckMust.Checked;

model.IncludeHistory = ckHistory.Checked;

WaitFormService.CreateWaitForm();

DataTable dt = fSOServiceClient.GetRefundDetails(model);

dgvReturn.DataSource = dt;

dgvReturnDetails.DataSource = new DataTable();

if (dt.Rows.Count > 0)

{

double TotalAmount = 0;

foreach (DataRow i in dt.Rows)

{

TotalAmount += Convert.ToDouble(i["Deductamount"]);

}

txtTotalAmount.Text = Math.Round(TotalAmount, 2).ToString();

}

WaitFormService.CloseWaitForm();

}

/// <summary>

/// 获取退款历史记录

/// </summary>

private void GetReturnHistory()

{

string DataCenter = GetDataCenter();

DeductNoReturnSelectModel model = new DeductNoReturnSelectModel();

model.MContractNo = txtContractNo.Text;

model.SalesId = txtSalesId.OperatorId;

model.IsZero = ck1.Checked;

model.SignAName = txtSignAName.Text;

model.Brand = txtBrand.Text;

model.CustomerName = txtCustomerName.Text;

model.BeginDate = DateBegin.Value;

model.EndDate = DateEnd.Value;

model.DataCenter = DataCenter;

DataTable dt = fSOServiceClient.GetRefundHistoryDetails(model);

dgvReturnHistory.DataSource = dt;

}

/// <summary>

/// 按销售员和日期进行退款汇总

/// </summary>

private void GetReturnTotalHistory()

{

DateTime date = DateCalDate.Value;

DataTable dt = fSOServiceClient.GetRefundHistoryDetailsTotal(date);

dgvTotalReturnHstory.DataSource = dt;

}

/// <summary>

/// 读取文件

/// </summary>

private void ReadFiles()

{

DataTable dt = dgvReturnHistory.DataSource as DataTable;

if (dt != null)

{

var FileId = dt.Rows[dgvReturnHistory.CurrentRow.Index]["FileId"].ToString();

if (!string.IsNullOrEmpty(FileId))

{

if (Convert.ToInt32(FileId) > 0)

{

var FilePath = Global.BaseOperate.FSOFileRead(Convert.ToInt32(FileId));

System.Diagnostics.Process.Start(FilePath);

}

else

{

MessageBox.Show("没有上传文件", "提示", MessageBoxButtons.OK);

}

}

else

{

MessageBox.Show("没有上传文件", "提示", MessageBoxButtons.OK);

}

}

}

/// <summary>

/// 审核 或反审 1：审核 2:反审

/// </summary>

/// <param name="State"></param>

private void AuditReturn(int State)

{

string RIds = string.Empty;

DataTable dt = dgvReturnHistory.DataSource as DataTable;

if (dt != null)

{

int count = Convert.ToInt16(this.dgvReturnHistory.Rows.Count.ToString());

for (int i = 0; i < count; i++)

{

DataGridViewCheckBoxCell checkCell = (DataGridViewCheckBoxCell)dgvReturnHistory.Rows[i].Cells["choose"];

Boolean flag = Convert.ToBoolean(checkCell.Value);

if (flag == true) //查找被选择的数据行

{

checkCell.Value = false;

RIds += dgvReturnHistory.Rows[i].Cells["RId"].Value+",";

}

}

//var RId =Convert.ToInt32(dt.Rows[dgvReturnHistory.CurrentRow.Index]["RId"]);

if (RIds.Length == 0)

MessageBox.Show("审核时，请至少选中一行!", "系统提示", MessageBoxButtons.OK);

else

{

RIds = RIds.Substring(0, RIds.Length - 1);

MessageBox.Show(fSOServiceClient.AuditReturnRefund(State, RIds, UserId), "系统提示", MessageBoxButtons.OK);

}

}

}

//查询权限设置

private void SetRight()

{

btnReturn.Enabled =btnAudit.Visible=btnAntiAudit.Visible = Global.BaseOperate.CheckFuncRights(242302);

}

/// <summary>

/// 获取数据中心条件

/// </summary>

/// <returns></returns>

private string GetDataCenter()

{

string DataCenter = string.Empty;

if (ckall.Checked)

DataCenter = "";

else if (ckbj.Checked)

DataCenter = "北京数据中心";

else if (cksh.Checked)

DataCenter = "上海数据中心";

return DataCenter;

}

/// <summary>

/// 全选或全不选

/// </summary>

/// <param name="Type"></param>

private void ChooseAll(int Type)

{

DataTable dt = dgvReturnHistory.DataSource as DataTable;

int count = Convert.ToInt16(this.dgvReturnHistory.Rows.Count.ToString());

switch (Type)

{

case 1:

{

if (dt != null)

{

for (int i = 0; i < count; i++)

{

dgvReturnHistory.Rows[i].Cells["choose"].Value = true;

}

}

}

break;

case 2:

{

if (dt != null)

{

for (int i = 0; i < count; i++)

{

dgvReturnHistory.Rows[i].Cells["choose"].Value = false;

}

}

}

break;

}

}

#endregion

}

public partial class DeductNoReturnRuleFile : Form

{

#region 公有成员

public string CId = string.Empty;

#endregion

#region 私有成员

private static DeductNoReturnRuleFile instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

public DeductNoReturnRuleFile()

{

InitializeComponent();

dgvUpFile.AutoGenerateColumns = false;

}

#region 私有事件

//加载事件

private void DeductNoReturnRuleFile\_Load(object sender, EventArgs e)

{

GetFiles();

}

//添加文件

private void toolStripMenuItem1\_Click(object sender, EventArgs e)

{

OpenFileDialog fileDialog = new OpenFileDialog();

fileDialog.Title = "请选择文件";

if (fileDialog.ShowDialog() == DialogResult.OK)

{

string FileName = fileDialog.FileName;//返回文件的完整路径

int FileId = Global.BaseOperate.FSOFileUpload("DedcutNotReturn", FileName, "规则设置");

fSOServiceClient.AddDedcutNotReturnConfigFile(CId,FileId,FileName,UserId);

GetFiles();

}

}

//删除文件

private void toolStripMenuItem2\_Click(object sender, EventArgs e)

{

DataTable dt = dgvUpFile.DataSource as DataTable;

if (dt != null)

{

var FId = dgvUpFile.CurrentRow.Cells["Column1"].Value.ToString();

fSOServiceClient.DelDedcutNotReturnConfigFile(FId);

GetFiles();

}

}

//读取文件

private void toolStripMenuItem3\_Click(object sender, EventArgs e)

{

DataTable dt = dgvUpFile.DataSource as DataTable;

if (dt != null)

{

var FileId = dgvUpFile.CurrentRow.Cells["Column29"].Value.ToString();

if (!string.IsNullOrEmpty(FileId))

{

if (Convert.ToInt32(FileId) > 0)

{

var FilePath = Global.BaseOperate.FSOFileRead(Convert.ToInt32(FileId));

System.Diagnostics.Process.Start(FilePath);

}

else

{

MessageBox.Show("没有上传文件", "提示", MessageBoxButtons.OK);

}

}

else

{

MessageBox.Show("没有上传文件", "提示", MessageBoxButtons.OK);

}

}

}

#endregion

#region 私有方法

private void GetFiles()

{

dgvUpFile.DataSource = fSOServiceClient.GetDedcutNotReturnConfigFile(CId);

}

#endregion

}

public partial class DeductNoReturnRuleForm : Form

{

#region 公共成员

public string ContractNo = string.Empty;

public DateTime DefaultDate;

#endregion

#region 私有成员

private string UserId = Global.UserId;

private IFSOService fSOServiceClient = Global.FSOService;

private string FileName = string.Empty;

private int FileId = 0;

#endregion

public DeductNoReturnRuleForm()

{

InitializeComponent();

}

#region 私有事件

/// <summary>

/// 加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnRuleForm\_Load(object sender, EventArgs e)

{

txtSalesId.HostAddress = Global.LinkService.HostAddress;

txtContractNo.Text = ContractNo;

var listItem = new List<KeyValuePair<int, string>>();

listItem.Add(new KeyValuePair<int, string>(2, "特批不扣"));

listItem.Add(new KeyValuePair<int, string>(3, "其他免扣"));

listItem.Add(new KeyValuePair<int, string>(4, "永久赦免"));

txtRule.DataSource = listItem;

txtRule.DisplayMember = "Value";

txtRule.ValueMember = "Key";

txtRule.SelectedIndex = -1;

txtRule.SelectedIndexChanged += TxtRule\_SelectedIndexChanged;

DateBegin.Value = DateEnd.Value = DefaultDate;

}

private void TxtRule\_SelectedIndexChanged(object sender, EventArgs e)

{

if (txtRule.SelectedIndex == 0|| txtRule.SelectedIndex == 1)

{

txtSalesId.Enabled = true;

DateBegin.Enabled = true;

DateEnd.Enabled = true;

}

else

{

txtSalesId.Enabled = false;

DateBegin.Enabled = false;

DateEnd.Enabled = false;

}

}

/// <summary>

/// 确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSubmit\_Click(object sender, EventArgs e)

{

if (CheckData())

{

DateTime dateBegin = DateBegin.Value;

DateTime dateEnd = DateEnd.Value;

int DeductType = Convert.ToInt32(txtRule.SelectedValue);

string SalesVPId = txtSalesId.OperatorId;

string Remark = txtRemark.Text;

string result=fSOServiceClient.addDeductRule(ContractNo, DeductType, dateBegin, dateEnd, UserId,FileId, FileName, SalesVPId, Remark);

MessageBox.Show(result, "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

DialogResult = DialogResult.OK;

Close();

}

else

{

MessageBox.Show("数据格式有问题，请检查合同号,VP经理,扣款规则是否选填", "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

/// <summary>

/// 取消

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

/// <summary>

/// 上传文件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSupport\_Click(object sender, EventArgs e)

{

OpenFileDialog fileDialog = new OpenFileDialog();

fileDialog.Title = "请选择文件";

if (fileDialog.ShowDialog() == DialogResult.OK)

{

FileName = fileDialog.FileName;//返回文件的完整路径

FileId = Global.BaseOperate.FSOFileUpload("DedcutNotReturn", FileName, "规则设置");

txtFileName.Text = FileName;

}

}

#endregion

#region 私有方法

/// <summary>

/// 检查数据

/// </summary>

/// <returns></returns>

private bool CheckData()

{

bool result = true;

if (string.IsNullOrEmpty(txtContractNo.Text))

result = false;

if (txtRule.SelectedIndex == -1)

result = false;

if (txtRule.SelectedIndex == 0 && string.IsNullOrEmpty(txtSalesId.OperatorId))

result = false;

return result;

}

#endregion

}

public partial class DeductNoReturnRuleHistoryForm : Form

{

public DeductNoReturnRuleHistoryForm()

{

InitializeComponent();

dgvRuleHistory.AutoGenerateColumns = false;

}

#region 公共成员

public string ContractNo = string.Empty;

#endregion

#region 私有成员

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

#region 私有事件

/// <summary>

/// 画面加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnRuleHistoryForm\_Load(object sender, EventArgs e)

{

dgvRuleHistory.DataSource = fSOServiceClient.GetRuleHostory(ContractNo);

}

/// <summary>

/// 确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSubmit\_Click(object sender, EventArgs e)

{

Close();

}

#endregion

}

public partial class DeductNoReturnSalesVPEdit : Form

{

public DeductNoReturnSalesVPEdit()

{

InitializeComponent();

}

#region 公共成员

public string CId = string.Empty;

#endregion

#region 私有成员

private static DeductNotReturnPubRelaForm instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

#region 私有事件

//加载

private void DeductNoReturnSalesVPEdit\_Load(object sender, EventArgs e)

{

txtSalesVPId.HostAddress = Global.LinkService.HostAddress;

}

//确认

private void btnSubmit\_Click(object sender, EventArgs e)

{

string SalesVPId = txtSalesVPId.OperatorId;

if (!string.IsNullOrEmpty(SalesVPId))

{

fSOServiceClient.EditSalesVPId(CId, SalesVPId);

Close();

}

}

//取消

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

#endregion

}

public partial class DeductNotReturnPubRelaForm : Form

{

#region 公共成员

public string MContractNo = string.Empty;

public string OContractNo = string.Empty;

public string PId = string.Empty;

#endregion

#region 私有成员

private static DeductNotReturnPubRelaForm instance = null;

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

public DeductNotReturnPubRelaForm()

{

InitializeComponent();

dgvPubrRela.AutoGenerateColumns = false;

}

#region 私有事件

/// <summary>

/// 加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNotReturnPubRelaForm\_Load(object sender, EventArgs e)

{

dgvPubrRela.DataSource = fSOServiceClient.GetDeductNotReturnPubRela(MContractNo,OContractNo,PId);

}

//取消

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

//确认

private void BtnSubmit\_Click(object sender, EventArgs e)

{

DataTable dt = dgvPubrRela.DataSource as DataTable;

if (dt != null)

{

//获取所有被选中的合同号

int count = Convert.ToInt16(this.dgvPubrRela.Rows.Count.ToString());

var Modellist = new List<EditDeductNotReturnPubRelaModel>();

Boolean HasAmounOver = true;

for (int i = 0; i < count; i++)

{

double OriAmount = Convert.ToDouble(dgvPubrRela.Rows[i].Cells["Column4"].Value);

double AdjustAmount = Convert.ToDouble(dgvPubrRela.Rows[i].Cells["Column5"].Value);

if (OriAmount < AdjustAmount)

{

HasAmounOver = false;

}

Modellist.Add(new EditDeductNotReturnPubRelaModel

{

adjustAmount = AdjustAmount,

MContractNo = dgvPubrRela.Rows[i].Cells["Column3"].Value.ToString(),

SalesId = dgvPubrRela.Rows[i].Cells["Column7"].Value.ToString(),

Pid = Convert.ToInt32(dgvPubrRela.Rows[i].Cells["Column6"].Value)

});

}

if (HasAmounOver)

{

string result = fSOServiceClient.EditDeductNotReturnPubRela(MContractNo, PId, Modellist, UserId);

MessageBox.Show(result, "提示", MessageBoxButtons.OK);

Close();

}

else

{

MessageBox.Show("调整金额不能大于扣款金额", "提示", MessageBoxButtons.OK);

}

}

}

#endregion

}

public partial class DeductReturnAddForm : Form

{

#region 公共成员

public string ContractNo = string.Empty;

#endregion

#region 私有成员

private string UserId = Global.UserId;

private IFSOService fSOServiceClient = Global.FSOService;

private string FileName = string.Empty;

private int FileId = 0;

private double Deductamount = 0;

#endregion

public DeductReturnAddForm()

{

InitializeComponent();

}

#region 私有事件

/// <summary>

/// 加载

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void DeductNoReturnRuleForm\_Load(object sender, EventArgs e)

{

txtSalesId.HostAddress = Global.LinkService.HostAddress;

txtContractNo.Text = ContractNo;

txtSalesId.OperatorId = fSOServiceClient.GetDeductNotReturnSalesId(ContractNo);

Deductamount =fSOServiceClient.GetTotalDeductAmount(ContractNo);

txtDeductAmount.Text = Deductamount.ToString();

DateReturn.Value = Convert.ToDateTime(fSOServiceClient.GetCalDate().Rows[0]["FDATE"]);

}

/// <summary>

/// 确认

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSubmit\_Click(object sender, EventArgs e)

{

if (CheckData())

{

if (Convert.ToDouble(txtDeductAmount.Text) > Deductamount)

{

MessageBox.Show("退款金额大于该合同的暂扣金额", "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

DeductReturnBack model = new DeductReturnBack

{

MContractNo = txtContractNo.Text,

DateRe = DateReturn.Value,

ReturnAmount = Convert.ToDouble(txtDeductAmount.Text),

SignerId = txtSalesId.OperatorId,

FileName = FileName,

FileId = FileId,

UserId = UserId

};

fSOServiceClient.addRefundBackDetails(model);

MessageBox.Show("规则修改成功,修改后的规则将再下次未回合同扣款计算时生效.若有特殊金额设置,请自行对未回合同详情信息进行修改!", "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

DialogResult = DialogResult.OK;

Close();

}

}

else

{

MessageBox.Show("数据格式有问题，请检查", "提示", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

/// <summary>

/// 取消

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

/// <summary>

/// 选择合同号

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void button1\_Click(object sender, EventArgs e)

{

string contractId = string.Empty;

string contractNo = string.Empty;

SelectContractForm.Select(out contractId, out contractNo);

txtContractNo.Text = contractNo;

}

/// <summary>

/// 上传附件

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSupport\_Click(object sender, EventArgs e)

{

OpenFileDialog fileDialog = new OpenFileDialog();

fileDialog.Title = "请选择文件";

if (fileDialog.ShowDialog() == DialogResult.OK)

{

FileName = fileDialog.FileName;//返回文件的完整路径

FileId = Global.BaseOperate.FSOFileUpload("DedcutNotReturn", FileName, "规则设置");

txtFileName.Text = FileName;

}

}

#endregion

#region 私有方法

/// <summary>

/// 检查数据

/// </summary>

/// <returns></returns>

private bool CheckData()

{

bool result = true;

if (string.IsNullOrEmpty(txtContractNo.Text) ||string.IsNullOrEmpty(txtSalesId.OperatorId))

result = false;

try

{

decimal var1 = Convert.ToDecimal(txtDeductAmount.Text);

}

catch

{

result = false;

}

//if (string.IsNullOrEmpty(FileName) || FileId == 0)

// result = false;

return result;

}

#endregion

}

public partial class DepartmentForm : Form

{

#region 私有成员

private IFSOService fSOServiceClient = Global.FSOService;

string UserId = Global.UserId;

#endregion

public DepartmentForm()

{

InitializeComponent();

}

#region 公共成员

public int RId = -1;

public int iDo = 0; //1 修改 2 删除

#endregion

#region 私有事件

//加载

private void DepartmentForm\_Load(object sender, EventArgs e)

{

selectDepartment.HostAddress = Global.LinkService.HostAddress;

if (RId > -1)

{

DataTable dt = fSOServiceClient.GetDedcutNotReturnExcludeById(RId.ToString());

selectDepartment.DeptId = dt.Rows[0]["DeptId"].ToString();

DateBegin.Value=Convert.ToDateTime(dt.Rows[0]["BeginDate"]);

DateEnd.Value= Convert.ToDateTime(dt.Rows[0]["EndDate"]);

}

}

/// <summary>

/// 确定

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnSubmit\_Click(object sender, EventArgs e)

{

string DeptId = selectDepartment.DeptId;

DateTime Datebegin = DateBegin.Value;

DateTime Dateend = DateEnd.Value;

MessageBox.Show(fSOServiceClient.EditDedcutNotReturnExclude(RId, iDo, DeptId, Datebegin, Dateend, UserId), "提示", MessageBoxButtons.OK);

Close();

}

/// <summary>

/// 取消

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void btnCancel\_Click(object sender, EventArgs e)

{

Close();

}

#endregion

}