SCCM SQL Queries

Contents

1. A Specific System Is Part of what are all Collections

```sql
Declare @Name Varchar(255)
Set @Name = 'CLIENT01' --Provide Computer Name

Select vrs.Name0,fcm.CollectionID,Col.name as 'CollectionName',vrs.Client0
from v_R_System as VRS
inner join v_FullCollectionMembership as FCM on VRS.ResourceID = FCM.resourceID
inner join v_Collection as Col on fcm.CollectionID = col.CollectionID
where VRS.Name0 = @Name and col.CollectionID not like 'SMS%'
order by col.Name
```

2. All Users Names Part of specific AD Group

```sql
DECLARE @GroupName varchar(255)
set @GroupName = '%LAB-SCCM_Device_Adobe_Acrobat_Reader_11.0.03_EN%' -- Provide Group Name

Select
vrug.User_Group_Name0 as GroupName,
Vru.Name0 as Username from v_R_User as Vru
inner join v_RA_User_UserGroupName as Vrug on Vru.ResourceID=Vrug.ResourceID
where Vrug.User_Group_Name0 like @GroupName
```

3. Package Deployment Detailed status for specific Advertisement ID

```sql
Declare @AdvID varchar (255)
set @AdvID = '00020001' --Advertisement ID

Select
vca.AdvertisementID,
vc.Name as 'CollectionName',
vrs.Name0,
vrs.User_Name0,
vrs.AD_Site_Name0,
vca.LastAcceptanceStateName,
vca.LastAcceptanceStatusTime,
vca.LastStateName,
vca.LastStatusMessageIDName,
vca.LastExecutionResult
from v_Advertisement as Va
inner join v_ClientAdvertisementStatus as Vca on va.AdvertisementID=vca.AdvertisementID
inner join v_R_System as Vrs on vca.ResourceID=vrs.ResourceID
inner join v_Collection as Vc on va.CollectionID=vc.CollectionID
Where va.AdvertisementID = @AdvID
order by vca.LastStateName
```

4. All SQL Server Installed Version and Computer Details using Hardware Inventory

```sql
Declare @SoftwareName varchar (255)
Set @SoftwareName = '%Microsoft SQL Server%'

Select
distinct vrs.Name0,
vrs.User_Name0,
vga.DisplayName0,
vga.InstallDate0,
vga.Publisher0,
vga.ProdID0
from v_R_System as Vrs
inner join v_GS_ADD_REMOVE_PROGRAMS_64 as Vga on Vrs.ResourceID=Vga.ResourceID
where Vga.DisplayName0 like @SoftwareName
and vga.Publisher0 is not null
```

5. All Computers with Last Heartbeat Discovery Time Stamp

```sql
Select
vrs.Name0 as 'ComputerName',
vrs.Client0 as 'Client',
vrs.Operating_System_Name_and0 as 'Operating System',
Vad.AgentTime as 'LastHeartBeatTime'
from v_R_System as Vrs
inner join v_AgentDiscoveries as Vad on Vrs.ResourceID=Vad.ResourceId
where vad.AgentName like '%Heartbeat Discovery'
```

6. All IE Version using Software Inventory

```sql
Declare @EXEName varchar(255)
Declare @EXEPath varchar(255)

Set @EXEName = 'IExplore%'
Set @EXEPath = '_:\Program Files\Internet Explorer\'

Select
vrs.Name0,
vrs.User_Name0,
vrs.AD_Site_Name0,
vgs.FileName,
CASE
WHEN vgs.FileVersion LIKE '5.%' THEN 'Internet Explorer 5'
WHEN vgs.FileVersion LIKE '6.%' THEN 'Internet Explorer 6'
WHEN vgs.FileVersion LIKE '7.%' THEN 'Internet Explorer 7'
WHEN vgs.FileVersion LIKE '8.%' THEN 'Internet Explorer 8'
WHEN vgs.FileVersion LIKE '9.%' THEN 'Internet Explorer 9'
WHEN vgs.FileVersion LIKE '10.%' THEN 'Internet Explorer 10'
WHEN vgs.FileVersion LIKE '11.%' THEN 'Internet Explorer 11'
ELSE 'Other Version' END AS 'IE Version',
vgs.FilePath,
vgs.FileVersion
from v_R_System as Vrs
join v_GS_SoftwareFile as Vgs on vrs.ResourceID=vgs.ResourceID
Where vgs.FileName like @EXEName and vgs.FilePath like @EXEPath
--and vrs.Name0 in ('XXXXX','xxxxx1')
order by vrs.Name0
```

7. All Packages Total Targeted DP Counts

```sql
Select
vp.PackageID,
vp.Name,
Case vp.PackageType
When 0 Then 'Package'
When 3 Then 'Driver'
When 4 Then 'Task Sequence'
When 5 Then 'software Update'
When 7 Then 'Virtual'
When 8 Then 'Application'
When 257 Then 'Image'
When 258 Then 'Boot Image'
When 259 Then 'OS'
Else ' '
END AS 'PackageType',
vp.Manufacturer,
vp.Version,
vp.Language,
```

```sql
(pkgs.SourceSize/1024) as 'Package Size (MB)',
vp.PkgSourcePath,
vp.SourceVersion,
vp.SourceDate,
Pkgs.Targeted,
pkgs.Installed,
pkgs.Pending,
pkgs.Retrying,
pkgs.Failed,
pkgs.Unknown
from
vPkgStatusSummary as Pkgs
Inner Join v_Package as Vp on pkgs.PkgID=vp.PackageID
order by 3
```

8. All Packages Compare targeted Packages on Two DPS

```sql
select s.SiteCode,s.PackageID,p.Name,(p.sourcesize)/1024 as 'Size(MB)',
s.sourceversion as 'DPVersion',p.storedpkgversion as 'LastVersion',s.Installstatus as 'Package Status',
Case v_Package.PackageType
When 0 Then 'Package'
When 3 Then 'Driver'
When 4 Then 'Task Sequence'
When 5 Then 'software Update'
When 7 Then 'Virtual'
When 8 Then 'Application'
When 257 Then 'Image'
When 258 Then 'Boot Image'
When 259 Then 'OS'
Else ' '
END AS 'Type'
from v_PackageStatusDistPointsSumm s
inner join smspackages p on s.packageid = p.pkgid
inner join v_Package on v_Package.PackageID=p.[PkgID]
where s.PackageID not in
(select PackageID from v_DistributionPoint where ServerNALPath like '%DPServerName1%') and ServerNALPath like
'%DPServerName1%'
order by 8
```

9. Compare Packages with two DP'S

```sql
select
Pkg.PackageID,
Pkg.Name,
Case Pkg.PackageType
When 0 Then 'Package'
When 3 Then 'Driver'
When 4 Then 'TaskSequence'
When 5 Then 'softwareUpdate'
When 7 Then 'Virtual'
When 8 Then 'Application'
When 257 Then 'Image'
When 258 Then 'BootImage'
When 259 Then 'OS'
Else ' '
END AS 'Type'
from v_Package Pkg
where Pkg.PackageID in
(
select PackageID from v_DistributionPoint where ServerNALPath like  '%Master DP Name%'
and PackageID not in
(select PackageID from v_DistributionPoint where ServerNALPath like  '%Compare DP Name%')
)
order by 3
```

10. All Software Distribution Packages without Advertisements

```sql
select v_Package.PackageID, v_Package.Name,v_Package.SourceVersion,v_Package.SourceDate
from dbo.v_package
Where
packageID not in (select PackageID from dbo.v_Advertisement)and
PackageID not in (SELECT ReferencePackageID FROM v_TaskSequenceReferencesInfo) and
v_Package.name not like '%osd%' and V_package.PackageType ='0'
group by v_Package.PackageID, v_Package.Name,v_Package.SourceVersion,v_Package.SourceDate
order by v_Package.PackageID
```

11. All Client Settings Status

```sql
SELECT *
FROM v_CH_Settings
where SettingsID = 1
```

12. All Workstations Agent Health Status

```sql
Declare @CollectionID as Varchar(8)
Declare @Total as Numeric(8)
Declare @Healthy as Numeric(8)
Declare @Unhealthy as Numeric(8)
Declare @HWInventoryOK as Numeric(8)
Declare @HWInventoryNotOK as Numeric(8)
Declare @SWInventoryOK as Numeric(8)
Declare @SWInventoryNotOK as Numeric(8)
Declare @WSUSInventoryOK as Numeric(8)
Declare @WSUSInventoryNotOK as Numeric(8)

Set @CollectionID = 'SMS00001' -- specify scope collection ID

select @Total = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID and ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
)

select @Healthy = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
)

select @Unhealthy = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID Not in (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 ) and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
)

select @HWInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where DATEDIFF (day,LastHWScan,GetDate())<30)
)

select @HWInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in ( select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
and ResourceID Not in (select ResourceID from v_GS_WORKSTATION_STATUS where DATEDIFF (day,LastHWScan,GetDate())<30)
)
```

```sql
select @SWInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan where DATEDIFF (day,LastScanDate,GetDate())<30)
)

select @SWInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
and ResourceID Not in (select ResourceID from v_GS_LastSoftwareScan where DATEDIFF (day,LastScanDate,GetDate())<30)
)

select @WSUSInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and DATEDIFF
(day,LastScanTime,GetDate())<30)
)

select @WSUSInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
and ResourceID Not in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and DATEDIFF
(day,LastScanTime,GetDate())<30)
)

select
@Total as 'Total',
@Healthy as 'Healthy',
@Unhealthy as 'Unhealthy',
@HWInventoryOK as 'HW<30Days',
@HWInventoryNotOK as 'HW>30Days',
@SWInventoryOK as 'SW<30Days',
@SWInventoryNotOK as 'SW>30Days',
@WSUSInventoryOK as 'WSUS<30Days',
@WSUSInventoryNotOK as 'WSUS>30Days',
case when (@Total = 0) or (@Total is null) Then '100' Else (round(@Healthy/ convert (float,@Total)*100,2)) End as 'Healthy%',
case when (@Healthy = 0) or (@Healthy is null) Then '100' Else (round(@HWInventoryOK/ convert (float,@Healthy)*100,2)) End as
'HW%',
case when (@Healthy = 0) or (@Healthy is null) Then '100' Else (round(@SWInventoryOK/ convert (float,@Healthy)*100,2)) End as
'SW%',
case when (@Healthy = 0) or (@Healthy is null) Then '100' Else (round(@WSUSInventoryOK/ convert (float,@Healthy)*100,2)) End as
'WSUS%'
```

13. All Servers Agent Health Status

```sql
Declare @CollectionID as Varchar(8)
Declare @Total as Numeric(8)
Declare @Healthy as Numeric(8)
Declare @Unhealthy as Numeric(8)
Declare @HWInventoryOK as Numeric(8)
Declare @HWInventoryNotOK as Numeric(8)
Declare @SWInventoryOK as Numeric(8)
Declare @SWInventoryNotOK as Numeric(8)
Declare @WSUSInventoryOK as Numeric(8)
Declare @WSUSInventoryNotOK as Numeric(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID
```

```sql
select @Total = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID and ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
)
select @Healthy = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
)
select @Unhealthy = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID Not in (select ResourceID from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 ) and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
)
select @HWInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where DATEDIFF
(day,LastHWScan,GetDate())<30)
)
select @HWInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in ( select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID Not in (select ResourceID from v_GS_WORKSTATION_STATUS where DATEDIFF
(day,LastHWScan,GetDate())<30)
)
select @SWInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan where DATEDIFF
(day,LastScanDate,GetDate())<30)
)
select @SWInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID Not in (select ResourceID from v_GS_LastSoftwareScan where DATEDIFF
(day,LastScanDate,GetDate())<30)
)
select @WSUSInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and DATEDIFF
(day,LastScanTime,GetDate())<30)
)
select @WSUSInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
```

```sql
and ResourceID Not in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and DATEDIFF
(day,LastScanTime,GetDate())<30)
)
select
@Total as 'Total',
@Healthy as 'Healthy',
@Unhealthy as 'Unhealthy',
@HWInventoryOK as 'HW<30Days',
@HWInventoryNotOK as 'HW>30Days',
@SWInventoryOK as 'SW<30Days',
@SWInventoryNotOK as 'SW>30Days',
@WSUSInventoryOK as 'WSUS<30Days',
@WSUSInventoryNotOK as 'WSUS>30Days',
case when (@Total = 0) or (@Total is null) Then '0' Else (round(@Healthy/ convert
(float,@Total)*100,2)) End as 'Healthy%',
case when (@Healthy = 0) or (@Healthy is null) Then '0' Else (round(@HWInventoryOK/ convert
(float,@Healthy)*100,2)) End as 'HW%',
case when (@Healthy = 0) or (@Healthy is null) Then '0' Else (round(@SWInventoryOK/ convert
(float,@Healthy)*100,2)) End as 'SW%',
case when (@Healthy = 0) or (@Healthy is null) Then '0' Else (round(@WSUSInventoryOK/ convert
(float,@Healthy)*100,2)) End as 'WSUS%'

Declare @CollectionID as Varchar(8)
Declare @Total as Numeric(8)
Declare @Healthy as Numeric(8)
Declare @Unhealthy as Numeric(8)
Declare @HWInventoryOK as Numeric(8)
Declare @HWInventoryNotOK as Numeric(8)
Declare @SWInventoryOK as Numeric(8)
Declare @SWInventoryNotOK as Numeric(8)
Declare @WSUSInventoryOK as Numeric(8)
Declare @WSUSInventoryNotOK as Numeric(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID
select @Total = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID and ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
)
select @Healthy = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
)
select @Unhealthy = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID Not in (select ResourceID from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 ) and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
)
select @HWInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where DATEDIFF
(day,LastHWScan,GetDate())<30)
)
select @HWInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
```

```
ResourceID in ( select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID Not in (select ResourceID from v_GS_WORKSTATION_STATUS where DATEDIFF
(day,LastHWScan,GetDate())<30)
)
select @SWInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan where DATEDIFF
(day,LastScanDate,GetDate())<30)
)
select @SWInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID Not in (select ResourceID from v_GS_LastSoftwareScan where DATEDIFF
(day,LastScanDate,GetDate())<30)
)
select @WSUSInventoryOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and DATEDIFF
(day,LastScanTime,GetDate())<30)
)
select @WSUSInventoryNotOK = (
select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete = 0 and IsClient = 1 and
ResourceID in (
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
and ResourceID Not in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and DATEDIFF
(day,LastScanTime,GetDate())<30)
)
select
@Total as 'Total',
@Healthy as 'Healthy',
@Unhealthy as 'Unhealthy',
@HWInventoryOK as 'HW<30Days',
@HWInventoryNotOK as 'HW>30Days',
@SWInventoryOK as 'SW<30Days',
@SWInventoryNotOK as 'SW>30Days',
@WSUSInventoryOK as 'WSUS<30Days',
@WSUSInventoryNotOK as 'WSUS>30Days',
case when (@Total = 0) or (@Total is null) Then '0' Else (round(@Healthy/ convert
(float,@Total)*100,2)) End as 'Healthy%',
case when (@Healthy = 0) or (@Healthy is null) Then '0' Else (round(@HWInventoryOK/ convert
(float,@Healthy)*100,2)) End as 'HW%',
case when (@Healthy = 0) or (@Healthy is null) Then '0' Else (round(@SWInventoryOK/ convert
(float,@Healthy)*100,2)) End as 'SW%',
case when (@Healthy = 0) or (@Healthy is null) Then '0' Else (round(@WSUSInventoryOK/ convert
(float,@Healthy)*100,2)) End as 'WSUS%'
```

14. All Workstations Agent Detailed Health Status

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001'    --Specify the collection ID

select distinct(Name),Case when IsClient= 1 then 'Healthy' else 'Unhealthy' end as 'HealthStatus',
(select  case when count (v_GS_WORKSTATION_STATUS.ResourceID)=1 then 'Healthy' else  'Unhealthy' end
```

```sql
from v_GS_WORKSTATION_STATUS where DATEDIFF (day,LastHWScan,GetDate())<31 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'HWScanStatus',
(select case when count (v_GS_LastSoftwareScan.ResourceID)=1 then 'Healthy' else 'Unhealthy' end
from v_GS_LastSoftwareScan where DATEDIFF (day,LastScanDate,GetDate())<31 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'SWScanStatus',
(select case when count (v_UpdateScanStatus.ResourceID)=1 then 'Healthy' else 'Unhealthy' end
from v_UpdateScanStatus where DATEDIFF (day,LastScanTime,GetDate())<31 and LastErrorCode = 0 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'WSUSScanStatus',
(select DATEDIFF (day,LastHWScan,GetDate()) from v_GS_WORKSTATION_STATUS
where ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastHWScanDays',
(select DATEDIFF (day,LastScanDate,GetDate()) from v_GS_LastSoftwareScan
where ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastSWScanDays',
(select DATEDIFF (day,LastScanTime,GetDate()) from v_UpdateScanStatus
where LastErrorCode = 0 and ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastWSUSScanDays'
from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID in ( select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
order by 2 desc
```

15. All Servers Agent Detailed Health Status

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001'      --Specify the collection ID

select distinct(Name),Case when IsClient= 1 then 'Healthy' else 'Unhealthy' end as 'HealthStatus',
(select  case when count (v_GS_WORKSTATION_STATUS.ResourceID)=1 then 'Healthy' else  'Unhealthy' end
from v_GS_WORKSTATION_STATUS where DATEDIFF (day,LastHWScan,GetDate())<31 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'HWScanStatus',
(select case when count (v_GS_LastSoftwareScan.ResourceID)=1 then 'Healthy' else 'Unhealthy' end
from v_GS_LastSoftwareScan where DATEDIFF (day,LastScanDate,GetDate())<31 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'SWScanStatus',
(select case when count (v_UpdateScanStatus.ResourceID)=1 then 'Healthy' else 'Unhealthy' end
from v_UpdateScanStatus where DATEDIFF (day,LastScanTime,GetDate())<31 and LastErrorCode = 0 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'WSUSScanStatus',
(select DATEDIFF (day,LastHWScan,GetDate()) from v_GS_WORKSTATION_STATUS
where ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastHWScanDays',
(select DATEDIFF (day,LastScanDate,GetDate()) from v_GS_LastSoftwareScan
where ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastSWScanDays',
(select DATEDIFF (day,LastScanTime,GetDate()) from v_UpdateScanStatus
where LastErrorCode = 0 and ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastWSUSScanDays'
from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID in ( select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%')
order by 2 desc
```

16. All Workstations Client Health Summary Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalClient as Numeric(8)
```

```
Declare @ClientInstalled as Numeric(8)
Declare @ClientNotInstalled as Numeric(8)

Set @CollectionID = 'SMS00001'       --Specify the collection ID

select @TotalClient = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID and ResourceID in (
select ResourceID from v_R_System where (Client0 = 1 or Client0 = 0 or Client0 is null) and Unknown0 is null
and Operating_System_Name_and0 like '%Workstation%')
)

select @ClientInstalled = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID and ResourceID in (
select ResourceID from v_R_System where Client0 = 1  and Operating_System_Name_and0 like '%Workstation%')
)

select @ClientNotInstalled = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID and ResourceID in (
select ResourceID from v_R_System where (Client0 = 0 or Client0 is null) and Unknown0 is null
and Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalClient as 'TotalClient',
@ClientInstalled as 'ClientInstalled',
@ClientNotInstalled as 'ClientNotInstalled',
case when (@TotalClient = 0) or (@TotalClient is null) Then '100' Else (round(@ClientInstalled/ convert (float,@TotalClient)*100,2))
End as 'ClientInstalled%'
```

17. All Servers Client Health Summary Status

```
Declare @CollectionID as Varchar(8)
Declare @TotalClient as Numeric(8)
Declare @ClientInstalled as Numeric(8)
Declare @ClientNotInstalled as Numeric(8)

Set @CollectionID = 'SMS00001'        --Specify the collection ID

select @TotalClient = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID and
ResourceID in (
select ResourceID from v_R_System where (Client0 = 1 or Client0 = 0 or Client0 is null) and Unknown0 is
null
and Operating_System_Name_and0 like '%Server%')
)

select @ClientInstalled = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID and
ResourceID in (
select ResourceID from v_R_System where Client0 = 1  and Operating_System_Name_and0 like '%Server%')
)

select @ClientNotInstalled = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID and
ResourceID in (
select ResourceID from v_R_System where (Client0 = 0 or Client0 is null) and Unknown0 is null
and Operating_System_Name_and0 like '%Server%')
)
select
@TotalClient as 'TotalClient',
@ClientInstalled as 'ClientInstalled',
@ClientNotInstalled as 'ClientNotInstalled',
```

```sql
case when (@TotalClient = 0) or (@TotalClient is null) Then '100' Else (round(@ClientInstalled/ convert
(float,@TotalClient)*100,2)) End as 'ClientInstalled%'
```

18. All Active and Inactive Workstations Client Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalClientInstalled as Numeric(8)
Declare @ClientActive as Numeric(8)
Declare @ClientInActive as Numeric(8)

Set @CollectionID = 'SMS00001'          --Specify the collection ID

select @TotalClientInstalled = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1 or Ch.ClientActiveStatus = 0) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ClientActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ClientInActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 0) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalClientInstalled as 'TotalClientInstalled',
@ClientActive as 'ClientActive',
@ClientInActive as 'ClientInActive',
case when (@TotalClientInstalled = 0) or (@TotalClientInstalled is null) Then '100' Else (round(@ClientActive/ convert
(float,@TotalClientInstalled)*100,2)) End as 'ClientActive%'
```

19. All Active and Inactive Servers Client Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalClientInstalled as Numeric(8)
Declare @ClientActive as Numeric(8)
Declare @ClientInActive as Numeric(8)

Set @CollectionID = 'SMS00001'                  --Specify the collection ID

select @TotalClientInstalled = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1 or Ch.ClientActiveStatus = 0) and Vrs.Operating_System_Name_and0 like
'%Servers%')
)
```

```sql
select @ClientActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ClientInActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 0) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select
@TotalClientInstalled as 'TotalClientInstalled',
@ClientActive as 'ClientActive',
@ClientInActive as 'ClientInActive',
case when (@TotalClientInstalled = 0) or (@TotalClientInstalled is null) Then '100' Else
(round(@ClientActive/ convert (float,@TotalClientInstalled)*100,2)) End as 'ClientActive%'
```

20. All Active Workstations Client Health Evaluation Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveEvalPass as Numeric(8)
Declare @ActiveEvalFail as Numeric(8)
Declare @ActiveEvalUnknown as Numeric(8)

Set @CollectionID = 'SMS00001'          --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Pass' or Ch.ClientStateDescription = 'Active/Fail'
or Ch.ClientStateDescription = 'Active/Unknown') and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActiveEvalPass = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Pass') and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActiveEvalFail = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Fail') and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActiveEvalUnknown = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
```

```sql
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Unknown') and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalActive as 'TotalActive',
@ActiveEvalPass as 'ActiveEvalPass',
@ActiveEvalFail as 'ActiveEvalFail',
@ActiveEvalUnknown as 'ActiveEvalUnknown',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveEvalPass/ convert (float,@TotalActive)*100,2))
End as 'ActiveEvalPass%'
```

21. All Active Servers Client Health Evaluation Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveEvalPass as Numeric(8)
Declare @ActiveEvalFail as Numeric(8)
Declare @ActiveEvalUnknown as Numeric(8)

Set @CollectionID = 'SMS00001'                    --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Pass' or Ch.ClientStateDescription = 'Active/Fail'
or Ch.ClientStateDescription = 'Active/Unknown') and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActiveEvalPass = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Pass') and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActiveEvalFail = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Fail') and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActiveEvalUnknown = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientStateDescription = 'Active/Unknown') and Vrs.Operating_System_Name_and0 like
'%Servers%')
)

select
@TotalActive as 'TotalActive',
@ActiveEvalPass as 'ActiveEvalPass',
@ActiveEvalFail as 'ActiveEvalFail',
@ActiveEvalUnknown as 'ActiveEvalUnknown',
```

```
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveEvalPass/ convert
(float,@TotalActive)*100,2)) End as 'ActiveEvalPass%'
```

22. All Active Workstations Client Heartbeat (DDR) Status

```
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveHeartBeatDDR as Numeric(8)
Declare @InActiveHeartBeatDDR as Numeric(8)


Set @CollectionID = 'SMS00001'            --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActiveHeartBeatDDR = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveDDR = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @InActiveHeartBeatDDR = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveDDR = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalActive as 'TotalActive',
@ActiveHeartBeatDDR as 'ActiveHeartBeatDDR',
@InActiveHeartBeatDDR as 'InActiveHeartBeatDDR',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveHeartBeatDDR/ convert

(float,@TotalActive)*100,2)) End as 'ActiveHeartBeatDDR%'
```

23. All Active Servers Client Heartbeat (DDR) Status

```
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveHeartBeatDDR as Numeric(8)
Declare @InActiveHeartBeatDDR as Numeric(8)


Set @CollectionID = 'SMS00001'                --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActiveHeartBeatDDR = (
```

```sql
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveDDR = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @InActiveHeartBeatDDR = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveDDR = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select
@TotalActive as 'TotalActive',
@ActiveHeartBeatDDR as 'ActiveHeartBeatDDR',
@InActiveHeartBeatDDR as 'InActiveHeartBeatDDR',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveHeartBeatDDR/
convert (float,@TotalActive)*100,2)) End as 'ActiveHeartBeatDDR%'
```

24. All Active Workstations Client Hardware Inventory Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveHWInv as Numeric(8)
Declare @InActiveHWInv as Numeric(8)

Set @CollectionID = 'SMS00001'                --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActiveHWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveHW = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @InActiveHWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveHW = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalActive as 'TotalActive',
@ActiveHWInv as 'ActiveHWInv',
@InActiveHWInv as 'InActiveHWInv',
```

```
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveHWInv/ convert (float,@TotalActive)*100,2))
End as 'ActiveHWInv%'
```

25. All Active Servers Client Hardware Inventory Status

```
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveHWInv as Numeric(8)
Declare @InActiveHWInv as Numeric(8)

Set @CollectionID = 'SMS00001'                          --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActiveHWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveHW = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)


select @InActiveHWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveHW = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select
@TotalActive as 'TotalActive',
@ActiveHWInv as 'ActiveHWInv',
@InActiveHWInv as 'InActiveHWInv',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveHWInv/ convert
(float,@TotalActive)*100,2)) End as 'ActiveHWInv%'
```

26. All Active Workstations Client Software Inventory Status

```
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveSWInv as Numeric(8)
Declare @InActiveSWInv as Numeric(8)

Set @CollectionID = 'SMS00001'                --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActiveSWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
```

```sql
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveSW = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @InActiveSWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveSW = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalActive as 'TotalActive',
@ActiveSWInv as 'ActiveSWInv',
@InActiveSWInv as 'InActiveSWInv',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveSWInv/ convert (float,@TotalActive)*100,2))
End as 'ActiveSWInv%'
```

27. All Active Servers Client Software Inventory Status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActiveSWInv as Numeric(8)
Declare @InActiveSWInv as Numeric(8)

Set @CollectionID = 'SMS00001'                        --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActiveSWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveSW = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @InActiveSWInv = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActiveSW = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select
@TotalActive as 'TotalActive',
@ActiveSWInv as 'ActiveSWInv',
@InActiveSWInv as 'InActiveSWInv',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActiveSWInv/ convert
(float,@TotalActive)*100,2)) End as 'ActiveSWInv%'
```

28. All Active Workstations Client Policy Request Status

```sql
Declare @CollectionID as Varchar(8)
```

```
Declare @TotalActive as Numeric(8)
Declare @ActivePolicyRequest as Numeric(8)
Declare @InActivePolicyRequest as Numeric(8)

Set @CollectionID = 'SMS00001'          --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @ActivePolicyRequest = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActivePolicyRequest = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select @InActivePolicyRequest = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActivePolicyRequest = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Workstation%')
)

select
@TotalActive as 'TotalActive',
@ActivePolicyRequest as 'ActivePolicyRequest',
@InActivePolicyRequest as 'InActivePolicyRequest',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActivePolicyRequest/ convert
(float,@TotalActive)*100,2)) End as 'ActivePolicyRequest%'
```

29. All Active Servers Client Policy Request Status

```
Declare @CollectionID as Varchar(8)
Declare @TotalActive as Numeric(8)
Declare @ActivePolicyRequest as Numeric(8)
Declare @InActivePolicyRequest as Numeric(8)

Set @CollectionID = 'SMS00001'                --Specify the collection ID

select @TotalActive = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (Ch.ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like '%Servers%')
)

select @ActivePolicyRequest = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActivePolicyRequest = 1 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like
'%Servers%')
)
```

```
select @InActivePolicyRequest = (
select COUNT(*) as 'Count' from v_FullCollectionMembership where CollectionID = @CollectionID
and v_FullCollectionMembership.ResourceID in (
Select Vrs.ResourceID from v_R_System Vrs
inner join v_CH_ClientSummary Ch on Vrs.ResourceID = ch.ResourceID
where (IsActivePolicyRequest = 0 and ClientActiveStatus = 1) and Vrs.Operating_System_Name_and0 like
'%Servers%')
)


select
@TotalActive as 'TotalActive',
@ActivePolicyRequest as 'ActivePolicyRequest',
@InActivePolicyRequest as 'InActivePolicyRequest',
case when (@TotalActive = 0) or (@TotalActive is null) Then '100' Else (round(@ActivePolicyRequest/
convert (float,@TotalActive)*100,2)) End as 'ActivePolicyRequest%'
```

30. All PCs discovered from specific site

```
Select  agent.AgentSite,sys.Netbios_Name0, sys.Resource_Domain_OR_Workgr0, MAX(AgentTime) as AgentTime
from v_R_System sys
join v_AgentDiscoveries agent on sys.ResourceID=agent.ResourceId
where agent.AgentSite= 'A00'
group by  agent.AgentSite,sys.Netbios_Name0, sys.Resource_Domain_OR_Workgr0
order by agent.AgentSite, Netbios_Name0
```

31. All PCs Information with IP address and subnet details

```
Declare @CollectionID varchar(8)
set @CollectionID = 'SMS00001'
Select
vrs.Name0,
vrs.User_Name0 as 'LastUserName',
vos.Caption0,
vos.CSDVersion0,
vos.InstallDate0,
vpb.Manufacturer0,
vgc.Model0,
vpb.SerialNumber0,
vos.LastBootUpTime0,
(vd.Size0/1024) as 'HDDSize GB)',
(vpm.TotalPhysicalMemory0/1024/1024) as 'RAMSize  GB',
gna.IPAddress0,
gna.DefaultIPGateway0,
gna.DHCPServer0,
case gna.DHCPEnabled0
when 1 then 'Yes'
else 'No'
end as 'DHCPEnabled',
gna.IPSubnet0,
gna.MACAddress0,
vrs.AD_Site_Name0,
Case Vrs.client0
When 1 then 'Yes'
else 'No'
End as 'Client',
Case vrs.Active0
When 1 then 'Active'
else 'No'
End as 'Active'
from v_R_System as Vrs
Inner Join v_FullCollectionMembership as Vfc on vrs.ResourceID=vfc.ResourceID
left join v_GS_NETWORK_ADAPTER_CONFIGUR as GNA on vrs.ResourceID=gna.ResourceID
```

```sql
left join v_GS_OPERATING_SYSTEM as VOS on vrs.ResourceID=vos.ResourceID
left join v_GS_DISK as VD on vrs.ResourceID=vd.ResourceID
left join v_GS_X86_PC_MEMORY as VPM on vrs.ResourceID=vpm.ResourceID
left join v_GS_PC_BIOS as VPB on vrs.ResourceID=vpb.ResourceID
left join v_GS_COMPUTER_SYSTEM as VGC on vrs.ResourceID=vgc.ResourceID
where vfc.CollectionID = @CollectionID
and GNA.IPAddress0 is not null
and vd.MediaType0 like 'Fixed hard disk media'
```

32. All PCs with chassis type information

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001'      --Specify the collection ID

Select
distinct (v_R_System.ResourceID),
v_R_System.Name0  as 'Machine Name',
AD_Site_Name0 as 'AD Site',
v_R_System.Operating_System_Name_and0 as 'Operating System',
v_RA_System_SMSInstalledSites.SMS_Installed_Sites0 as 'Installed Site',
'Chassis Type' = CASE
when ChassisTypes0 = 1 THEN 'Virtual Machine'
when ChassisTypes0 = 2 THEN 'Unknown'
when ChassisTypes0 = 3 THEN 'Desktop'
when ChassisTypes0 = 4 THEN 'Low-profile Desktop'
when ChassisTypes0 = 5 THEN 'Pizza Box'
when ChassisTypes0 = 6 THEN 'Mini Tower'
when ChassisTypes0 = 7 THEN 'Tower'
when ChassisTypes0 = 8 THEN 'Portable'
when ChassisTypes0 = 9 THEN 'Laptop'
when ChassisTypes0 = 10 THEN 'Notebook'
when ChassisTypes0 = 11 THEN 'Handheld'
when ChassisTypes0 = 12 THEN 'Docking Station'
when ChassisTypes0 = 13 THEN 'All-in-One'
when ChassisTypes0 = 14 THEN 'Subnotebook'
when ChassisTypes0 = 15 THEN 'Space-Saving'
when ChassisTypes0 = 16 THEN 'Lunch Box'
when ChassisTypes0 = 17 THEN 'Main System chassis'
when ChassisTypes0 = 18 THEN 'Expansion chassis'
when ChassisTypes0 = 19 THEN 'Sub-Chassis'
when ChassisTypes0 = 20 THEN 'Bus-expansion chassis'
when ChassisTypes0 = 21 THEN 'Peripheral chassis'
when ChassisTypes0 = 22 THEN 'Storage chassis'
when ChassisTypes0 = 23 THEN 'Rack-mount chassis'
when ChassisTypes0 = 24 THEN 'Sealed-case computer'
END
from v_R_System
inner join v_GS_SYSTEM_ENCLOSURE on (v_GS_SYSTEM_ENCLOSURE.ResourceID = v_R_System.ResourceID)
inner join v_RA_System_SMSInstalledSites on (v_RA_System_SMSInstalledSites.ResourceID = v_R_System.ResourceID)
inner join v_FullCollectionMembership on (v_FullCollectionMembership.ResourceID = v_R_System.ResourceID)
where v_FullCollectionMembership.CollectionID = @CollectionID
```

33. All Desktops and Laptops counts details

```sql
SELECT
CASE
   WHEN ChassisTypes0 = '8' THEN 'Notebooks'
   WHEN ChassisTypes0 = '9' THEN 'Notebooks'
   WHEN ChassisTypes0 = '10' THEN 'Notebooks'
   ELSE 'Desktops'
END AS "Workstation Type", count(sys.name0) as ClientCount
from v_R_System SYS
LEFT  JOIN v_GS_SYSTEM_ENCLOSURE ENC ON ENC.ResourceID = SYS.ResourceID
LEFT  JOIN v_FullCollectionMembership FCM ON FCM.ResourceID = ENC.ResourceID
WHERE FCM.CollectionID = 'Collection ID' and sys.Obsolete0 = 0
GROUP BY
CASE ChassisTypes0
WHEN '8' THEN 'Notebooks'
```

```sql
WHEN '9' THEN 'Notebooks'
WHEN '10' THEN 'Notebooks'
ELSE 'Desktops'
END
ORDER BY 2 desc
```

34. All PCs Information with subnet and OU details

```sql
SELECT DISTINCT
SYS.Netbios_Name0,
SYS.User_Name0,OPSYS.InstallDate0 as InitialInstall,  BIOS.SerialNumber0,
CSYS.Model0,
MEM.TotalPhysicalMemory0, HWSCAN.LastHWScan,
ASSG.SMS_Installed_Sites0,
MAX(IPSub.IP_Subnets0) as 'Subnet',
OPSYS.Caption0 as 'OS Name',
MAX(SYSOU.System_OU_Name0) as 'OU'
FROM v_R_System  as SYS
JOIN v_RA_System_SMSInstalledSites as ASSG on SYS.ResourceID=ASSG.ResourceID
LEFT JOIN v_RA_System_IPSubnets IPSub on SYS.ResourceID = IPSub.ResourceID
LEFT JOIN  v_GS_X86_PC_MEMORY MEM on SYS.ResourceID = MEM.ResourceID
LEFT JOIN  v_GS_COMPUTER_SYSTEM CSYS on SYS.ResourceID = CSYS.ResourceID
LEFT JOIN  v_GS_PROCESSOR Processor  on Processor.ResourceID = SYS.ResourceID
LEFT JOIN  v_GS_OPERATING_SYSTEM OPSYS on SYS.ResourceID=OPSYS.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on SYS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_GS_LastSoftwareScan SWSCAN on SYS.ResourceID = SWSCAN.ResourceID
LEFT JOIN v_GS_PC_BIOS BIOS on SYS.ResourceID = BIOS.ResourceID
LEFT JOIN v_RA_System_SystemOUName SYSOU on SYS.ResourceID=SYSOU.ResourceID
LEFT JOIN v_R_User USR on SYS.User_Name0 = USR.User_Name0
LEFT JOIN v_FullCollectionMembership FCM on FCM.ResourceID = SYS.ResourceID
WHERE SYS.Obsolete0 = 0
--AND FCM.CollectionID = 'Collection ID'
GROUP BY SYS.Netbios_Name0, SYS.Obsolete0,SYS.Resource_Domain_OR_Workgr0,
CSYS.Manufacturer0, CSYS.Model0, BIOS.SerialNumber0,OPSYS.InstallDate0,HWSCAN.LastHWScan, MEM.TotalPhysicalMemory0,
SYS.User_Name0, SYS.User_Domain0,
ASSG.SMS_Installed_Sites0, SYS.Client_Version0, OPSYS.Caption0
ORDER BY OPSYS.InstallDate0 DESC
```

35. All PCs with particular application last used date

```sql
Declare @Monthold int
set @Monthold = 2

SELECT DISTINCT SYS.Netbios_Name0 as Name,
SF.FileName, SF.FileDescription, SF.FileVersion, SF.FileSize, SF.FileModifiedDate, SF.FilePath, max(apps.LastUsedTime0) as
LastUsedTime,
SYS.User_Name0 as Login,
CSYS.Manufacturer0 as Manufacturer, CSYS.Model0 as Model, BIOS.SerialNumber0 as SN,
MAX(IPSub.IP_Subnets0) as 'Subnet', sys.AD_Site_Name0 as ADSite, MAX(SYSOU.System_OU_Name0) as 'OU'
FROM v_GS_SoftwareFile SF
join v_R_System SYS on SYS.ResourceID = SF.ResourceID
LEFT JOIN v_RA_System_IPSubnets IPSub on SYS.ResourceID = IPSub.ResourceID
LEFT JOIN v_GS_COMPUTER_SYSTEM CSYS on SYS.ResourceID = CSYS.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM OPSYS on SYS.ResourceID=OPSYS.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on SYS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_GS_PC_BIOS BIOS on SYS.ResourceID = BIOS.ResourceID
LEFT JOIN v_RA_System_SystemOUName SYSOU on SYS.ResourceID=SYSOU.ResourceID
LEFT JOIN v_R_User USR on SYS.User_Name0 = USR.User_Name0
LEFT JOIN v_FullCollectionMembership FCM on SYS.ResourceID = FCM.ResourceID
LEFT JOIN (select * from v_GS_CCM_RECENTLY_USED_APPS where ExplorerFileName0 = 'notepad.exe') APPS on
SYS.ResourceID = APPS.ResourceID
Where SF.FileName LIKE 'notepad.exe'
GROUP BY SYS.Netbios_Name0, --apps.LastUsedTime0,
SF.FileName, SF.FileDescription, SF.FileVersion, SF.FileSize, SF.FileModifiedDate, SF.FilePath,
SYS.User_Name0, CSYS.Manufacturer0, CSYS.Model0, BIOS.SerialNumber0,
sys.AD_Site_Name0
HAVING max(apps.LastUsedTime0) < dateadd(month, -(@Monthold), dateadd(day,0,datediff(day,0,getdate()))) OR
max(apps.LastUsedTime0) IS NULL
```

ORDER BY SYS.Netbios_Name0

36. All PCs with particular software inventory Exe file

```
SELECT DISTINCT SYS.Netbios_Name0 as Name,
SF.FileName, SF.FileDescription, SF.FileVersion, SF.FileSize, SF.FileModifiedDate, SF.FilePath,
SYS.User_Name0 as Login, CSYS.Manufacturer0 as Manufacturer, CSYS.Model0 as Model, BIOS.SerialNumber0 as SN,
MAX(IPSub.IP_Subnets0) as 'Subnet', sys.AD_Site_Name0 as ADSite, MAX(SYSOU.System_OU_Name0) as 'OU'
FROM v_GS_SoftwareFile SF
join v_R_System SYS on SYS.ResourceID = SF.ResourceID
LEFT JOIN v_RA_System_IPSubnets IPSub on SYS.ResourceID = IPSub.ResourceID
LEFT JOIN v_GS_COMPUTER_SYSTEM CSYS on SYS.ResourceID = CSYS.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM OPSYS on SYS.ResourceID=OPSYS.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on SYS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_GS_PC_BIOS BIOS on SYS.ResourceID = BIOS.ResourceID
LEFT JOIN v_RA_System_SystemOUName SYSOU on SYS.ResourceID=SYSOU.ResourceID
LEFT JOIN v_R_User USR on SYS.User_Name0 = USR.User_Name0
LEFT JOIN v_FullCollectionMembership FCM on SYS.ResourceID = FCM.ResourceID
Where SF.FileName LIKE 'Microsoft.ConfigurationManagement.exe'
GROUP BY SYS.Netbios_Name0,SF.FileName, SF.FileDescription, SF.FileVersion, SF.FileSize, SF.FileModifiedDate,
SF.FilePath, SYS.User_Name0, CSYS.Manufacturer0, CSYS.Model0, BIOS.SerialNumber0, sys.AD_Site_Name0
ORDER BY SYS.Netbios_Name0
```

37. All PCs with Username and Email ID details

```
SELECT  SYS.User_Name0 as Login, USR.Mail0 as 'EMail ID', SYS.Netbios_Name0 as Machine,
Operating_System_Name_and0 as OS FROM v_R_System SYS
JOIN  v_R_User USR on USR.User_Name0 = SYS.User_Name0
--WHERE SYS.User_Name0 LIKE 'Username'
ORDER BY SYS.User_Name0, SYS.Netbios_Name0
```

38. All PCs with Configuration Manager Console installed details

```
Select sys.Name0 As Name, sys.user_name0 as UserName,
arp.DisplayName0 as DisplayName,
arp.Publisher0 as Publisher,
arp.Version0 as Version, max(arp.InstallDate0) AS InstallDate
FROM v_Add_Remove_Programs arp
JOIN  v_R_System sys ON arp.ResourceID = sys.ResourceID
LEFT JOIN v_FullCollectionMembership fcm on SYS.ResourceID=fcm.ResourceID
LEFT JOIN v_R_User USR on SYS.User_Name0 = USR.User_Name0
WHERE (arp.DisplayName0 like '%System Center%Configuration Manager Console%')
and arp.Publisher0 like '%Microsoft%'
GROUP BY sys.name0,sys.user_name0, arp.displayName0, arp.publisher0, arp.Version0
ORDER BY arp.displayName0, arp.publisher0
```

39. All PCs client assigned and installed site code details

```
Select v_R_System.Netbios_Name0 as 'NetBios Name',
v_R_System.AD_Site_Name0 as 'AD Site',
v_R_System.Active0 as 'isActive',
v_R_System.Obsolete0 as 'isObsolete',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'Assigned Site',
v_RA_System_SMSInstalledSites.SMS_Installed_Sites0 as 'Installed Site'
from v_R_System
inner join v_RA_System_SMSAssignedSites on (v_R_System.ResourceID = v_RA_System_SMSAssignedSites.ResourceID)
inner join v_RA_System_SMSInstalledSites on (v_R_System.ResourceID = v_RA_System_SMSInstalledSites.ResourceID)
Where SMS_Assigned_Sites0 like '%' and SMS_Installed_Sites0 like '%'
```

40. All PCs with No Clients based on OS Category status

```
Declare @CollectionID as Varchar(8)
Declare @TotalNoClientAgent as Numeric(8)
Declare @NoClientAgentWindowsOS as Numeric(8)
Declare @NoClientAgentWindowsOSLastLogonwithin7Days as Numeric(8)
Declare @NoClientAgentWindowsOSNOtLastLogonwithin7Days as Numeric(8)
Declare @NoClientAgentNonWindowsOS as Numeric(8)
```

```
Set @CollectionID = 'SMS00001' -- specify scope collection ID

select @TotalNoClientAgent = (
Select count(Vrs.ResourceID) as 'Count' from v_R_System Vrs
inner join v_FullCollectionMembership Vf on Vrs.ResourceID = Vf.ResourceID
where (Vrs.Client0 = 0 or Vrs.Client0 is null)
and Vf.CollectionID = @CollectionID
)
select @NoClientAgentWindowsOS = (
Select count(Vrs.ResourceID) as 'Count' from v_R_System Vrs
inner join v_FullCollectionMembership Vf on Vrs.ResourceID = Vf.ResourceID
where (Vrs.Client0 = 0 or Vrs.Client0 is null)
and Vrs.Unknown0 is null
and Vrs.Operating_System_Name_and0 like '%windows%'
and Vf.CollectionID = @CollectionID
)
select @NoClientAgentWindowsOSLastLogonwithin7Days = (
Select count(Vrs.ResourceID) as 'Count' from v_R_System Vrs
inner join v_FullCollectionMembership Vf on Vrs.ResourceID = Vf.ResourceID
where (Vrs.Client0 = 0 or Vrs.Client0 is null)
and Vrs.Unknown0 is null
and Vrs.Operating_System_Name_and0 like '%windows%'
and (DATEDIFF(day,Last_Logon_Timestamp0, GetDate())) < 7
and Vf.CollectionID = @CollectionID
)
select @NoClientAgentWindowsOSNOtLastLogonwithin7Days = (
Select count(Vrs.ResourceID) as 'Count' from v_R_System Vrs
inner join v_FullCollectionMembership Vf on Vrs.ResourceID = Vf.ResourceID
where (Vrs.Client0 = 0 or Vrs.Client0 is null)
and Vrs.Unknown0 is null
and Vrs.Operating_System_Name_and0 like '%windows%'
and (DATEDIFF(day,Last_Logon_Timestamp0, GetDate())) >= 7
and Vf.CollectionID = @CollectionID
)
select @NoClientAgentNonWindowsOS = (
Select count(Vrs.ResourceID) as 'Count' from v_R_System Vrs
inner join v_FullCollectionMembership Vf on Vrs.ResourceID = Vf.ResourceID
where (Vrs.Client0 = 0 or Vrs.Client0 is null)
and Vrs.Unknown0 is null
and Vrs.Operating_System_Name_and0 Not like '%windows%'
and Vf.CollectionID = @CollectionID
)

select
@TotalNoClientAgent as 'TotalNoClientAgent',
@NoClientAgentWindowsOS as 'NoClientAgentWindowsOS',
@NoClientAgentNonWindowsOS as 'NoClientAgentNonWindowsOS',
@NoClientAgentWindowsOSLastLogonwithin7Days as'NoClientAgentWindowsOSLastLogonwithin7Days',
@NoClientAgentWindowsOSNOtLastLogonwithin7Days as 'NoClientAgentWindowsOSNOtLastLogonwithin7Days',
case when (@NoClientAgentWindowsOS = 0) or (@NoClientAgentWindowsOS is null) Then '100' Else
(round(@NoClientAgentWindowsOSLastLogonwithin7Days/ convert (float,@NoClientAgentWindowsOS)*100,2)) End
as 'NoClientAgentWindowsOSLastLogonwithin7Days%'
```

41. All Workstations Client version status

```
Declare @CollectionID as Varchar(8)

Set @CollectionID = 'SMS00001'   -- specify scope collection ID

Select sys.Client_Version0 as 'Client Agent Version', count (sys.ResourceID) as 'Count' from v_R_System sys
inner join v_CH_ClientSummary ch on sys.ResourceID = ch.ResourceID
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
```

```
where (Ch.ClientActiveStatus = 1 and Sys.Operating_System_Name_and0 like '%Workstation%')
and Vf.CollectionID = @CollectionID
Group by sys.Client_Version0
Order by sys.Client_Version0 desc
```

42. All PCs with chassis type information

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001'      --Specify the collection ID

Select
distinct (v_R_System.ResourceID),
v_R_System.Name0 as 'Machine Name',
AD_Site_Name0 as 'AD Site',
v_R_System.Operating_System_Name_and0 as 'Operating System',
v_RA_System_SMSInstalledSites.SMS_Installed_Sites0 as 'Installed Site',
'Chassis Type' = CASE
when ChassisTypes0 = 1 THEN 'Virtual Machine'
when ChassisTypes0 = 2 THEN 'Unknown'
when ChassisTypes0 = 3 THEN 'Desktop'
when ChassisTypes0 = 4 THEN 'Low-profile Desktop'
when ChassisTypes0 = 5 THEN 'Pizza Box'
when ChassisTypes0 = 6 THEN 'Mini Tower'
when ChassisTypes0 = 7 THEN 'Tower'
when ChassisTypes0 = 8 THEN 'Portable'
when ChassisTypes0 = 9 THEN 'Laptop'
when ChassisTypes0 = 10 THEN 'Notebook'
when ChassisTypes0 = 11 THEN 'Handheld'
when ChassisTypes0 = 12 THEN 'Docking Station'
when ChassisTypes0 = 13 THEN 'All-in-One'
when ChassisTypes0 = 14 THEN 'Subnotebook'
when ChassisTypes0 = 15 THEN 'Space-Saving'
when ChassisTypes0 = 16 THEN 'Lunch Box'
when ChassisTypes0 = 17 THEN 'Main System chassis'
when ChassisTypes0 = 18 THEN 'Expansion chassis'
when ChassisTypes0 = 19 THEN 'Sub-Chassis'
when ChassisTypes0 = 20 THEN 'Bus-expansion chassis'
when ChassisTypes0 = 21 THEN 'Peripheral chassis'
when ChassisTypes0 = 22 THEN 'Storage chassis'
when ChassisTypes0 = 23 THEN 'Rack-mount chassis'
when ChassisTypes0 = 24 THEN 'Sealed-case computer'
END
from v_R_System
inner join v_GS_SYSTEM_ENCLOSURE on (v_GS_SYSTEM_ENCLOSURE.ResourceID = v_R_System.ResourceID)
inner join v_RA_System_SMSInstalledSites on (v_RA_System_SMSInstalledSites.ResourceID = v_R_System.ResourceID)
inner join v_FullCollectionMembership on (v_FullCollectionMembership.ResourceID = v_R_System.ResourceID)
where v_FullCollectionMembership.CollectionID = @CollectionID
```

43. All Workstations Client Installation Failure status

```
select count(cdr.MachineID) as 'Count',
cdr.CP_LastInstallationError as 'Error Code'
from v_CombinedDeviceResources cdr
where
cdr.IsClient = 0
and cdr.DeviceOS like '%Windows%'
group by cdr.CP_LastInstallationError
```

44. All Workstations with Last Boot up time status

```
Declare @CollectionID as Varchar(8)

Set @CollectionID = 'SMS00001'   --Specify the collection ID

select 'Last Reboot within 7 days' as TimePeriod,Count(sys.Name0) as 'Count',1 SortOrder
from v_R_System sys
```

```sql
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where os.LastBootUpTime0 < DATEADD(day,-7, GETDATE())
and ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
UNION
select 'Last Reboot within 14 days' as TimePeriod,Count(sys.Name0) as 'Count',2
from v_R_System sys
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where os.LastBootUpTime0 < DATEADD(day,-14, GETDATE())
and ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
UNION
select 'Last Reboot within 1 month' as TimePeriod,Count(sys.Name0) as 'Count',3
from v_R_System sys
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where os.LastBootUpTime0 < DATEADD(month,-1, GETDATE())
and ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
UNION
select 'Last Reboot within 3 months' as TimePeriod,Count(sys.Name0) as 'Count',4
from v_R_System sys
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where os.LastBootUpTime0 < DATEADD(month,-3, GETDATE())
and ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
UNION
select 'Last Reboot within 6 months' as TimePeriod,Count(sys.Name0) as 'Count',5
from v_R_System sys
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where os.LastBootUpTime0 < DATEADD(month,-6, GETDATE())
and ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
UNION
select 'Last Reboot within 12 months' as TimePeriod,Count(sys.Name0) as 'Count',6
from v_R_System sys
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where os.LastBootUpTime0 < DATEADD(month,-12, GETDATE())
and ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
UNION
select 'Total Machines Count' as TimePeriod,Count(sys.Name0) as 'Count',7
from v_R_System sys
inner join v_GS_OPERATING_SYSTEM os on os.ResourceId = sys.ResourceId
inner join v_FullCollectionMembership Vf on sys.ResourceID = Vf.ResourceID
inner join v_CH_ClientSummary ch on ch.ResourceID = sys.ResourceID
where ch.ClientActiveStatus = 1 and sys.Operating_System_Name_and0 like '%workstation%'
and Vf.CollectionID = @CollectionID
Order By SortOrder
```

45. All ConfigMgr Roles Status

```sql
select distinct
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Site System') as 'SiteSys',
```

```sql
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Component Server') as 'CompSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Site Server') as 'SiteSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Management Point') as 'MP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Distribution Point') as 'DP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS SQL Server') as 'SQL',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Software Update Point') as 'SUP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS SRS Reporting Point') as 'SSRS',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Reporting Point') as 'RPT',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Fallback Status Point') as 'FSP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Server Locator Point') as 'SLP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS PXE Service Point') as 'PXE',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS System Health Validator') as 'SysVal',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS State Migration Point') as 'SMP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Notification Server') as 'NotiSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Provider') as 'SMSPro',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Application Web Service') as 'WebSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Portal Web Site') as 'WebSite',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Branch distribution point') as 'BranDP'
from v_SystemResourceList
```

46. All IE Version using Software Inventory

```sql
SELECT distinct
Vrs.Netbios_Name0,
Vrs.User_Name0,
Vrs.AD_Site_Name0,
CASE
WHEN Sf.FileVersion LIKE '5.%' THEN 'Internet Explorer 5'
WHEN Sf.FileVersion LIKE '6.%' THEN 'Internet Explorer 6'
WHEN Sf.FileVersion LIKE '7.%' THEN 'Internet Explorer 7'
WHEN Sf.FileVersion LIKE '8.%' THEN 'Internet Explorer 8'
WHEN Sf.FileVersion LIKE '9.%' THEN 'Internet Explorer 9'
WHEN Sf.FileVersion LIKE '10.%' THEN 'Internet Explorer 10'
WHEN Sf.FileVersion LIKE '11.%' THEN 'Internet Explorer 11'
ELSE 'Other Version' END AS 'IE Version',
Sf.FileName,
Sf.FileVersion
FROM v_R_System Vrs
Inner Join v_GS_SoftwareFile Sf On Vrs.ResourceID = Sf.ResourceID
WHERE Sf.FileName = 'iexplore.exe'
and Sf.FilePath like '_:\Program%Internet Explorer%'
GROUP BY
Vrs.Netbios_Name0,
Vrs.User_Name0,
Vrs.AD_Site_Name0,
Sf.FileName,
Sf.FileVersion
ORDER BY Vrs.Netbios_Name0
```

47. All Packages which are waiting to distribute content to DPs

```sql
Select
SubString(dp.ServerNALPath, CHARINDEX('\\', dp.ServerNALPath)+2,(CHARINDEX('"]', dp.ServerNALPath) -
CHARINDEX('\\', dp.ServerNALPath))-3) as ServerName,
dp.SiteCode as 'SiteCode',
dp.PackageID as 'PackageID',
p.Name as 'PackageName',
P.SourceVersion as 'SourceVersion',
P.LastRefreshTime as 'LastRefreshTime',
stat.InstallStatus as 'InstallStatus'
from v_DistributionPoint dp left join v_PackageStatusDistPointsSumm stat on
dp.ServerNALPath=stat.ServerNALPath and dp.PackageID=stat.PackageID
```

```
left join v_PackageStatus pstat on dp.ServerNALPath=pstat.PkgServer and
dp.PackageID=pstat.PackageID left outer join v_Package p on dp.packageid = p.packageid
where stat.InstallStatus not in ('Package Installation complete')
ORDER BY 1
```

48. All Packages Content Distribution Status

```
select v_SystemResourceList.ServerName as 'ServerName',
v_SystemResourceList.SiteCode,
(select count(*)from v_PackageStatusDistPointsSumm where servernalpath = nalpath) as 'Targetted',
(select count(*)from v_PackageStatusDistPointsSumm where installstatus ='Package Installation
complete'and servernalpath = nalpath) as 'Installed',
(select count(*)from v_PackageStatusDistPointsSumm where (installstatus ='Content updating' or
installstatus ='Waiting to install package' or installstatus ='Content monitoring') and servernalpath =
nalpath) as 'Waiting',
(select count(*)from v_PackageStatusDistPointsSumm where installstatus like'%Retry%'and servernalpath =
nalpath) as 'Retrying',
(select count(*)from v_PackageStatusDistPointsSumm where installstatus ='Waiting to remove package' and
servernalpath = nalpath) as 'Removing',
(select count(*)from v_PackageStatusDistPointsSumm where installstatus like'%Fail%' and servernalpath =
nalpath) as 'Failed',
(select ROUND((100 * (select count(*)from v_PackageStatusDistPointsSumm where installstatus ='Package
Installation complete' and servernalpath = nalpath)/(select count(*)from v_PackageStatusDistPointsSumm
where servernalpath = nalpath)),2)) as 'Compliance %'
from v_SystemResourceList join v_PackageStatusDistPointsSumm
on v_SystemResourceList.nalpath = v_PackageStatusDistPointsSumm.servernalpath
where v_SystemResourceList.RoleName = 'SMS Distribution Point'
group by v_SystemResourceList.SiteCode, v_SystemResourceList.servername, v_SystemResourceList.nalpath
order by 1
```

49. All ConfigMgr Issue Servers Status

```
Select SiteStatus.SiteCode,
SiteInfo.ServerName,
SiteInfo.SiteName,
SiteStatus.Updated 'TimeStamp',
Case SiteInfo.Status When 1 Then 'Active' When 2 Then 'Pending' When 3 Then 'Failed' When 4 Then
'Deleted' When 5 Then 'Upgrade' Else ' ' END AS 'SiteState',
Case SiteStatus.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else ' ' End AS
'Status'
From V_SummarizerSiteStatus SiteStatus Join v_Site SiteInfo on SiteStatus.SiteCode = SiteInfo.SiteCode
where SiteInfo.Status <> 1 or SiteStatus.Status = 2 Order By SiteCode
```

50. All Software applications required deployments status within 30 days

```
Declare @SoftwareAppDeploymentsReportNeededDays as integer
Set @SoftwareAppDeploymentsReportNeededDays = 30     --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
```

```
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1 and Ds.DeploymentIntent = 1
and Ds.CreationTime > GETDATE()-@SoftwareAppDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

51. All Software applications available deployments status within 30 days

```
Declare @SoftwareAppDeploymentsReportNeededDays as integer
Set @SoftwareAppDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1 and Ds.DeploymentIntent = 2
and Ds.CreationTime > GETDATE()-@SoftwareAppDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

52. All Software applications simulate deployments status within 30 days

```
Declare @SoftwareAppDeploymentsReportNeededDays as integer
Set @SoftwareAppDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' when
Ds.DeploymentIntent = 3 Then 'Simulate' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
```

```
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1 and Ds.DeploymentIntent = 3
and Ds.CreationTime > GETDATE()-@SoftwareAppDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

53. All Software packages required deployments status within 30 days

```
Declare @SoftwarePkgDeploymentsReportNeededDays as integer
Set @SoftwarePkgDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1)as 'ApplicationName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2 and Ds.DeploymentIntent = 1
and Ds.ModificationTime > GETDATE()-@SoftwarePkgDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

54. All Software packages available deployments status within 30 days

```
Declare @SoftwarePkgDeploymentsReportNeededDays as integer
Set @SoftwarePkgDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1)as 'ApplicationName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
```

```
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2 and Ds.DeploymentIntent = 2
and Ds.ModificationTime > GETDATE()-@SoftwarePkgDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

55. All Software updates required deployments status within 30 days

```
Declare @PatchDeploymentsReportNeededDays as integer
Set @PatchDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
'Software Update' as 'PackageName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5 and Ds.DeploymentIntent = 1
and Vaa.LastModificationTime > GETDATE()-@PatchDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

56. All Software updates available deployments status within 30 days

```
Declare @PatchDeploymentsReportNeededDays as integer
Set @PatchDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
```

```
'Software Update' as 'PackageName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5 and Ds.DeploymentIntent = 2
and Vaa.LastModificationTime > GETDATE()-@PatchDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

57. All OSD required deployments status within 30 days

```
Declare @SoftwareOSDeploymentsReportNeededDays as integer
Set @SoftwareOSDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7 and Ds.DeploymentIntent = 1
and Ds.ModificationTime > GETDATE()-@SoftwareOSDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

58. All OSD available deployments status within 30 days

```
Declare @SoftwareOSDeploymentsReportNeededDays as integer
Set @SoftwareOSDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7 and Ds.DeploymentIntent = 2
and Ds.ModificationTime > GETDATE()-@SoftwareOSDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

59. All ConfigMgr Roles Detailed status

```
SELECT
srl.ServerName,
srl.SiteCode,
vs.SiteName,
vrs.AD_Site_Name0 as ADSite,
vs.ReportingSiteCode as Parent,
vs.Installdir,
MAX(CASE srl.rolename WHEN 'SMS Site System' THEN 'Yes' Else ' ' END) as SiteSys,
MAX(CASE srl.rolename WHEN 'SMS Component Server' THEN 'Yes' Else ' ' END) as CompSer,
MAX(CASE srl.rolename WHEN 'SMS Site Server' THEN 'Yes' Else ' ' END) as SiteSer,
MAX(CASE srl.rolename WHEN 'SMS Management Point' THEN 'Yes' Else ' ' END) as MP,
MAX(CASE srl.rolename WHEN 'SMS Distribution Point' THEN 'Yes' Else ' ' END) as DP,
MAX(CASE srl.rolename WHEN 'SMS SQL Server' THEN 'Yes' Else ' ' END) as 'SQL',
MAX(CASE srl.rolename WHEN 'SMS Software Update Point' THEN 'Yes' Else ' ' END) as SUP,
MAX(CASE srl.rolename WHEN 'SMS SRS Reporting Point' THEN 'Yes' Else ' ' END) as SSRS,
MAX(CASE srl.RoleName WHEN 'SMS Reporting Point' THEN 'Yes' Else ' ' END) as RPT,
MAX(CASE srl.rolename WHEN 'SMS Fallback Status Point' THEN 'Yes' Else ' ' END) as FSP,
MAX(CASE srl.rolename WHEN 'SMS ServerName Locator Point' THEN 'Yes' Else ' ' END) as SLP,
MAX(CASE srl.rolename WHEN 'SMS PXE Service Point' THEN 'Yes' Else ' ' END) as PXE,
MAX(CASE srl.rolename WHEN 'AI Update Service Point' THEN 'Yes' Else ' ' END) as AssI,
MAX(CASE srl.rolename WHEN 'SMS State Migration Point' THEN 'Yes' Else ' ' END) as SMP,
MAX(CASE srl.rolename WHEN 'SMS System Health Validator' THEN 'Yes' Else ' ' END) as SysVal,
MAX(CASE srl.rolename WHEN 'SMS Notification Server' THEN 'Yes' Else ' ' END) as NotiSer,
MAX(CASE srl.rolename WHEN 'SMS Provider' THEN 'Yes'  Else ' ' END) as SMSPro,
MAX(CASE srl.rolename WHEN 'SMS Application Web Service' THEN 'Yes' Else ' ' END) as WebSer,
MAX(CASE srl.rolename WHEN 'SMS Portal Web Site' THEN 'Yes' Else ' ' END) as WebSite,
MAX(CASE srl.rolename WHEN 'SMS Branch distribution point' THEN 'Yes'  Else ' ' END) as BranDP
FROM
v_SystemResourceList as srl
LEFT JOIN v_site vs on srl.ServerName = vs.ServerName
```

```
LEFT JOIN v_R_System_Valid vrs on LEFT(srl.ServerName, CHARINDEX('.', srl.ServerName) - 1)  =
vrs.Netbios_Name0
GROUP BY
srl.ServerName,
srl.SiteCode,
vs.SiteName,
vs.ReportingSiteCode,
vrs.AD_Site_Name0,
vs.InstallDir
ORDER BY srl.sitecode,srl.ServerName
```

60. All SCCM Server Software Update Sync status

```
SELECT US.SiteCode, S.ServerName, S.SiteName, US.ContentVersion, US.SyncTime, US.LastSyncState,
US.LastSyncStateTime, US.LastErrorCode
FROM update_syncstatus US, v_Site S where US.SiteCode=S.SiteCode
ORDER BY SyncTime
```

61. All Software applications required deployments status within 5 days

```
Declare @CurrentDeploymentsReportNeededDays as integer
Set @CurrentDeploymentsReportNeededDays = 5      --Specify the Days

Select
CONVERT(VARCHAR(11),GETDATE(),106) as 'Date',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'ReqDays'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1 and Ds.DeploymentIntent = 1
and DateDiff(D,Ds.EnforcementDeadline, GetDate()) between 0 and @CurrentDeploymentsReportNeededDays
and Ds.NumberTotal > 0
order by Ds.EnforcementDeadline desc
```

62. All Software packages required deployments status within 5 days

```
Declare @CurrentDeploymentsReportNeededDays as integer
Set @CurrentDeploymentsReportNeededDays = 5      --Specify the Days

Select
CONVERT(VARCHAR(11),GETDATE(),106) as 'Date',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1)as 'ApplicationName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
```

```sql
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailDays'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2 and Ds.DeploymentIntent = 1
and DateDiff(D,Ds.DeploymentTime, GetDate()) between 0 and @CurrentDeploymentsReportNeededDays
and Ds.NumberTotal > 0
order by Ds.DeploymentTime desc
```

63. All Software updates required deployments status within 5 days

```sql
Declare @CurrentDeploymentsReportNeededDays as integer
Set @CurrentDeploymentsReportNeededDays = 500      --Specify the Days

Select
CONVERT(VARCHAR(11),GETDATE(),106) as 'Date',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.AssignmentName as 'DeploymentName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'ReqDays'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5 and Ds.DeploymentIntent = 1
and DateDiff(D,Ds.EnforcementDeadline, GetDate()) between 0 and @CurrentDeploymentsReportNeededDays
and Ds.NumberTotal > 0
order by Ds.EnforcementDeadline desc
```

64. All Software applications deployments status within 30 days

```sql
Declare @AppDeploymentsReportNeededDays as integer
Set @AppDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
```

```
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' when
Ds.DeploymentIntent = 3 Then 'Simulate' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1
and Ds.CreationTime > GETDATE()-@AppDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

65. All Software packages deployments status within 30 days

```
Declare @PKGDeploymentsReportNeededDays as integer
Set @PKGDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1)as 'ApplicationName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2
and Ds.ModificationTime > GETDATE()-@PKGDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

66. All Software updates deployments status within 30 days

```sql
Declare @PatchDeploymentsReportNeededDays as integer
Set @PatchDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
'Software Update' as 'PackageName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5
and Vaa.LastModificationTime > GETDATE()-@PatchDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

67. All OS deployments status within 30 days

```sql
Declare @OSDeploymentsReportNeededDays as integer
Set @OSDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
```

```sql
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7
and Ds.ModificationTime > GETDATE()-@OSDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

68. All Software updates deployments status within 30 days

```sql
Declare @PatchDeploymentsReportNeededDays as integer
Set @PatchDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AssignmentName as 'DeploymentName',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5
and Vaa.LastModificationTime > GETDATE()-@PatchDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

69. All OS deployments status within 30 days

```sql
Declare @OSDeploymentsReportNeededDays as integer
Set @OSDeploymentsReportNeededDays = 30      --Specify the Days

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
```

```
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7
and Ds.ModificationTime > GETDATE()-@OSDeploymentsReportNeededDays
order by Ds.DeploymentTime desc
```

70. All Site Servers Issue MP components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_MP_CONTROL_MANAGER' and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

71. All Site Servers Issue DP components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_DISTRIBUTION_MANAGER' and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

72. All Site Servers Issue DDR components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
```

```
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_DISCOVERY_DATA_MANAGER'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

73. All Site Servers Issue CCR components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_CLIENT_CONFIG_MANAGER'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

74. All Site Servers Issue WSUS components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and (ComponentName ='SMS_WSUS_CONFIGURATION_MANAGER' or ComponentName ='SMS_WSUS_SYNC_MANAGER')
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

75. All Site Servers Issue Discovery components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
```

```
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and (ComponentName ='SMS_AD_SYSTEM_GROUP_DISCOVERY_AGENT' or ComponentName
='SMS_AD_SYSTEM_DISCOVERY_AGENT' or
ComponentName ='SMS_NETWORK_DISCOVERY' or ComponentName ='SMS_AD_SECURITY_GROUP_DISCOVERY_AGENT')
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

76. All Site Servers Issue Collection Evaluator components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_COLLECTION_EVALUATOR'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

77. All Site Servers Issue Hardware Inventory components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_componentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_INVENTORY_DATA_LOADER'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

78. All Site Servers Issue Despooler components status

```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
```

```sql
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_DESPOOLER'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

79. All Site Servers Issue Inbox Monitor components status

```sql
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_INBOX_MONITOR'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

80. All Site Servers Issue Component Monitor components status

```sql
SELECT distinct  SiteCode,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName ='SMS_COMPONENT_MONITOR'
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

81. All Site Servers Issue Others components status

```sql
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
```

```sql
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and ComponentName Not in
('SMS_MP_CONTROL_MANAGER','SMS_DISTRIBUTION_MANAGER','SMS_DISCOVERY_DATA_MANAGER',
'SMS_CLIENT_CONFIG_MANAGER','SMS_WSUS_CONFIGURATION_MANAGER','SMS_WSUS_SYNC_MANAGER',
'SMS_AD_SECURITY_GROUP_DISCOVERY_AGENT','SMS_AD_SYSTEM_GROUP_DISCOVERY_AGENT',
'SMS_AD_SYSTEM_DISCOVERY_AGENT','SMS_NETWORK_DISCOVERY','SMS_COLLECTION_EVALUATOR',
'SMS_INVENTORY_DATA_LOADER','SMS_DESPOOLER','SMS_INBOX_MONITOR','SMS_COMPONENT_MONITOR'
)
and v_ComponentSummarizer.Status = 2 Order By SiteCode
```

82. All Workstations Not Assigned Clients detailed status

```sql
Declare @SCCMManagedWorkstationsScopeCollectionID as varchar(8)
Set @SCCMManagedWorkstationsScopeCollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct (VRS.Netbios_Name0) as 'Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_CH_ClientSummary.ClientActiveStatus = 1 Then 'Yes' Else 'No' End 'ClientHealthActive',
v_CH_ClientSummary.ClientStateDescription as 'ClientHealthDescription',
System_Disc.AD_Site_Name0 as 'ADSiteName',
Vrs.Operating_System_Name_and0 as 'OSType',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
v_CH_ClientSummary.LastActiveTime as 'LastCommunicated',
DateDiff(D, v_CH_ClientSummary.LastActiveTime, GetDate()) 'LastCommunicatedDays',
Vrs.Creation_Date0 as 'ClientCreation',
DateDiff(D, Vrs.Creation_Date0, GetDate()) 'ClientCreationDays',
System_Disc.Last_Logon_Timestamp0 as 'LastLogon',
DateDiff(D, System_Disc.Last_Logon_Timestamp0, GetDate()) 'LastLogonDays',
System_Disc.Distinguished_Name0 as 'OUName'
from V_R_System Vrs
Left Outer join  v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID =
VRS.ResourceId
Left Outer join  v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
left outer join  v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
left outer join  v_CH_ClientSummary on v_CH_ClientSummary.ResourceID = VRS.ResourceId
left outer join  System_Disc on System_Disc.ItemKey = VRS.ResourceId
where (Vrs.Client0 = 0 or Vrs.Client0 is null ) and
(VRS.Obsolete0 = 0 or VRS.Obsolete0 is null) and
v_FullCollectionMembership.CollectionID = @SCCMManagedWorkstationsScopeCollectionID and
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 is null and
Vrs.Operating_System_Name_and0 like '%Workstation%'
order by System_Disc.Last_Logon_Timestamp0 desc
```

83. All Workstations Unhealthy Clients detailed status

```sql
Declare @SCCMManagedWorkstationsScopeCollectionID as varchar(8)
Set @SCCMManagedWorkstationsScopeCollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct (VRS.Netbios_Name0) as 'Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_CH_ClientSummary.ClientActiveStatus = 1 Then 'Yes' Else 'No' End 'ClientHealthActive',
v_CH_ClientSummary.ClientStateDescription as 'ClientHealthDescription',
System_Disc.AD_Site_Name0 as 'ADSiteName',
Vrs.Operating_System_Name_and0 as 'OSType',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
```

```sql
v_CH_ClientSummary.LastActiveTime as 'LastCommunicated',
DateDiff(D, v_CH_ClientSummary.LastActiveTime, GetDate()) 'LastCommunicatedDays',
Vrs.Creation_Date0 as 'ClientCreation',
DateDiff(D, Vrs.Creation_Date0, GetDate()) 'ClientCreationDays',
System_Disc.Last_Logon_Timestamp0 as 'LastLogon',
DateDiff(D, System_Disc.Last_Logon_Timestamp0, GetDate()) 'LastLogonDays',
System_Disc.Distinguished_Name0 as 'OUName'
from V_R_System Vrs
Left Outer join  v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID =
VRS.ResourceId
Left Outer join  v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
left outer join  v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
left outer join  v_CH_ClientSummary on v_CH_ClientSummary.ResourceID = VRS.ResourceId
left outer join  System_Disc on System_Disc.ItemKey = VRS.ResourceId
where (Vrs.Client0 = 0 or Vrs.Client0 is null ) and
(VRS.Obsolete0 = 0 or VRS.Obsolete0 is null) and
v_FullCollectionMembership.CollectionID = @SCCMManagedWorkstationsScopeCollectionID and
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 is not null and
Vrs.Operating_System_Name_and0 like '%Workstation%'
order by System_Disc.Last_Logon_Timestamp0 desc
```

84. All Workstations Inactive Clients detailed status

```sql
Declare @SCCMManagedWorkstationsScopeCollectionID as varchar(8)
Set @SCCMManagedWorkstationsScopeCollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct (VRS.Netbios_Name0) as 'Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_CH_ClientSummary.ClientActiveStatus = 1 Then 'Yes' Else 'No' End 'ClientHealthActive',
v_CH_ClientSummary.ClientStateDescription as 'ClientHealthDescription',
System_Disc.AD_Site_Name0 as 'ADSiteName',
Vrs.Operating_System_Name_and0 as 'OSType',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
v_CH_ClientSummary.LastActiveTime as 'LastCommunicated',
DateDiff(D, v_CH_ClientSummary.LastActiveTime, GetDate()) 'LastCommunicatedDays',
Vrs.Creation_Date0 as 'ClientCreation',
DateDiff(D, Vrs.Creation_Date0, GetDate()) 'ClientCreationDays',
System_Disc.Last_Logon_Timestamp0 as 'LastLogon',
DateDiff(D, System_Disc.Last_Logon_Timestamp0, GetDate()) 'LastLogonDays',
System_Disc.Distinguished_Name0 as 'OUName'
from V_R_System Vrs
Left Outer join  v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID =
VRS.ResourceId
Left Outer join  v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
left outer join  v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
left outer join  v_CH_ClientSummary on v_CH_ClientSummary.ResourceID = VRS.ResourceId
left outer join  System_Disc on System_Disc.ItemKey = VRS.ResourceId
where (Vrs.Client0 = 1 and v_CH_ClientSummary.ClientActiveStatus <> 1 ) and
(VRS.Obsolete0 = 0 or VRS.Obsolete0 is null) and
v_FullCollectionMembership.CollectionID = @SCCMManagedWorkstationsScopeCollectionID and
Vrs.Operating_System_Name_and0 like '%Workstation%'
order by System_Disc.Last_Logon_Timestamp0 desc
```

85. All Obsolete Clients detailed status

```sql
Select
ResourceID as 'ResourceID',
Name0 as 'MachineName',
User_Name0 as 'LogonUserName',
Case When Client0 = 1 Then 'Yes' Else 'No' End as 'Client',
Case When Obsolete0 = 1 Then 'Yes' Else 'No' End as 'Obsolete',
```

```
Case When Active0 = 1 Then 'Yes' Else 'No' End as 'Active',
AD_Site_Name0 as 'ADSiteName',
Operating_System_Name_and0 as 'OSType',
Client_Version0 as 'ClientVersion',
Creation_Date0 as 'CreationDateinSCCM'
from v_R_system
where
(Obsolete0 =1 and Active0 = 0)
and Name0 not like '%Unknown%'
```

86. All Packages available in SCCM

```
SELECT DISTINCT p.PackageID, p.Name,p.Version,p.Language,p.Manufacturer,
PackageType = Case Packagetype
When 0 Then 'Software Distribution Package'
When 3 Then 'Driver Package'
When 4 Then 'Task Sequence Package'
When 5 Then 'Software Update Package'
When 6 Then 'Device Settings Package'
When 7 Then 'Virtual Package'
When 8 Then 'Software Distribution Application'
When 257 Then 'Image Package'
When 258 Then 'Boot Image Package'
When 259 Then 'OS Install Package'
End,
p.PkgSourcePath,p.SourceDate,p.LastRefreshTime ,p.SourceVersion,
p.SourceSite,n.Targeted,n.Installed,n.Retrying,n.Failed,(n.SourceSize/1024)as 'SourceSize(MB)',
(n.SourceCompressedSize/1024) as 'SourceCompressedSize(MB)'
FROM v_Package p LEFT JOIN v_PackageStatusRootSummarizer n
ON p.PackageID = n.PackageID
WHERE p.PackageType <> '4' and p.PackageType <> '258'
order by 1 desc
```

87. All Collections available in SCCM

```
SELECT
dbo.v_Collection.CollectionID as 'CollectionID',
dbo.v_Collections_G.LimitToCollectionID  as 'LimitToCollectionID',
dbo.v_Collection.Name as 'CollectionName',
dbo.Collections.LimitToCollectionName as 'LimitingCollectionName',
Case when dbo.v_Collections_G.CollectionType = 1 Then 'User' when dbo.v_Collections_G.CollectionType = 2 Then 'Device' Else
'Others' End as 'CollectionType',
dbo.v_Collection.MemberCount as 'MembersCount',
dbo.v_Collections_G.CollectionComment as 'CollectionComment',
dbo.Collections.BeginDate as 'CreatedDate',
dbo.Collections.LastMemberChangeTime as 'LastMemberChangeDate',
dbo.v_CollectionRuleQuery.RuleName as 'CollectionRuleName',
dbo.v_CollectionRuleQuery.QueryID as 'QueryID',
dbo.v_CollectionRuleQuery.QueryExpression as 'QueryExpression'
FROM dbo.v_Collection
left join dbo.v_CollectionRuleQuery ON dbo.v_Collection.CollectionID = dbo.v_CollectionRuleQuery.CollectionID
left join dbo.Collections on dbo.v_Collection.CollectionID = dbo.Collections.SiteID
left join dbo.v_Collections_G on dbo.v_Collection.CollectionID = dbo.v_Collections_G.SiteID
ORDER BY dbo.v_Collection.Name, dbo.v_Collections_G.CollectionType
```

88. All Managed Workstations details status

```
Declare @ProjectName as varchar(25)
Set @ProjectName = 'LAB' -- specify Project Name

SELECT
DISTINCT Vrs.resourceid as 'ResourceID',
Vrs.Name0 as 'MachineName',
'SCCM' as 'SourceName',
getdate() as 'ReportingDate',
bios.SerialNumber0 as 'SerialNumber',
cs.Manufacturer0 as 'Manufacturer',
encl.PartNumber0 as 'ManufacturerPartNumber',
cs.model0 as 'DeviceModel',
```

```sql
case when p.caption0 is null then p.name0 else p.caption0 end as 'ProcessorType',
(mem.TotalPhysicalMemory0 / 1024/1000) as 'MemorySize(GB)',
disk.Size0 / 1024 as 'HDDSize',
bios.Version0 as 'BIOSVersion',
@ProjectName as 'AccountCode',
encl.chassistypes0 as 'EquipmentType',
os.caption0 as 'OperatingSystem',
os.csdversion0 as 'ServicePackVersion',
os.InstallDate0 as 'InstallationDate',
Vrs.Creation_Date0 as 'CreationDateinSCCM'
FROM v_r_system Vrs
LEFT OUTER JOIN v_gs_pc_bios bios on Vrs.resourceid = bios.resourceid
LEFT OUTER JOIN v_gs_computer_system cs on Vrs.resourceid = cs.resourceid
LEFT OUTER JOIN v_gs_system_enclosure encl on Vrs.resourceid = encl.resourceid
LEFT OUTER JOIN v_gs_processor p on Vrs.resourceid = p.resourceid
LEFT OUTER JOIN v_gs_x86_pc_memory mem on Vrs.resourceid = mem.resourceid
LEFT OUTER JOIN v_gs_logical_disk disk on Vrs.resourceid = disk.resourceid
LEFT OUTER JOIN v_gs_operating_system os on Vrs.resourceid = os.resourceid
WHERE
disk.name0 = 'C:'
AND encl.tag0 = 'System Enclosure 0'
AND os.caption0 not like '%server%'
AND Vrs.Active0 = 1
AND Vrs.Client0 = 1
AND Vrs.Obsolete0 = 0
```

89. All Workstations Assets Inventory details status

```sql
Declare @CollectionID as varchar(8)
Declare @ProjectName as varchar(25)

Set @CollectionID = 'SMS00001' -- specify scope collection ID
Set @ProjectName = 'LAB' -- specify Project Name

Select
Distinct (VRS.Netbios_Name0) as 'Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 1 Then 'VMWare'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('3','4')Then 'Desktop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('8','9','10','11','12','14') Then 'Laptop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 6 Then 'Mini Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 7 Then 'Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 13 Then 'All in One'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 15 Then 'Space-Saving'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 17 Then 'Main System Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 21 Then 'Peripheral Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 22 Then 'Storage Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 23 Then 'Rack Mount Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 24 Then 'Sealed-Case PC'
Else 'Others' End 'CaseType',
LEFT(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0),
ISNULL(NULLIF(CHARINDEX(',',MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0)) - 1, -
1),LEN(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0))))as 'IPAddress',
MAX (v_GS_NETWORK_ADAPTER_CONFIGUR.MACAddress0) as 'MACAddress',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
VRS.Client_Version0 as 'ClientVersion',
VRS.Creation_Date0 as 'ClientCreationDate',
VRS.AD_Site_Name0 as 'ADSiteName',
dbo.v_GS_OPERATING_SYSTEM.InstallDate0 AS 'OSInstallDate',
DateDiff(D, dbo.v_GS_OPERATING_SYSTEM.InstallDate0, GetDate()) 'OSInstallDateAge',
Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100) as 'LastBootDate',
DateDiff(D, Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100), GetDate()) as 'LastBootDateAge',
PC_BIOS_DATA.SerialNumber00 as 'SerialNumber',
v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0 as 'AssetTag',
PC_BIOS_DATA.ReleaseDate00 as 'ReleaseDate',
PC_BIOS_DATA.Name00 as 'BiosName',
```

```sql
PC_BIOS_DATA.SMBIOSBIOSVersion00 as 'BiosVersion',
v_GS_PROCESSOR.Name0 as 'ProcessorName',
case when Computer_System_DATA.Manufacturer00 like 'VMware%' Then 'VMWare'
when Computer_System_DATA.Manufacturer00 like 'Gigabyte%' Then 'Gigabyte'
when Computer_System_DATA.Manufacturer00 like 'VIA Technologies%' Then 'VIA Technologies'
when Computer_System_DATA.Manufacturer00 like 'MICRO-STAR%' Then 'MICRO-STAR'
Else Computer_System_DATA.Manufacturer00 End 'Manufacturer',
Computer_System_DATA.Model00 as 'Model',
Computer_System_DATA.SystemType00 as 'OSType',
v_GS_COMPUTER_SYSTEM.Domain0 as 'DomainName',
VRS.User_Domain0+'\'+ VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
Case when v_GS_COMPUTER_SYSTEM.domainrole0 = 0 then 'Standalone Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 1 Then 'Member Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 2 Then 'Standalone Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 3 Then 'Member Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 4 Then 'Backup Domain Controller'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 5 Then 'Primary Domain Controller'
End 'Role',

case when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Enterprise Edition' Then 'Microsoft(R)
Windows(R) Server 2003 Enterprise Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Standard Edition' Then 'Microsoft(R) Windows(R)
Server 2003 Standard Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Web Edition' Then 'Microsoft(R) Windows(R)
Server 2003 Web Edition'
Else Operating_System_DATA.Caption00 End 'OSName',
Operating_System_DATA.CSDVersion00 as 'ServicePack',
Operating_System_DATA.Version00 as 'Version',
((v_GS_X86_PC_MEMORY.TotalPhysicalMemory0/1024)/1000) as 'TotalRAMSize(GB)',
max(v_GS_LOGICAL_DISK.Size0 / 1024) AS 'TotalHDDSize(GB)',
v_GS_WORKSTATION_STATUS.LastHWScan as 'LastHWScan',
DateDiff(D, v_GS_WORKSTATION_STATUS.LastHwScan, GetDate()) as 'LastHWScanAge',
@ProjectName as 'AccountName'
from V_R_System VRS
Left Outer join PC_BIOS_DATA on PC_BIOS_DATA.MachineID = VRS.ResourceId
Left Outer join Operating_System_DATA on Operating_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_WORKSTATION_STATUS on v_GS_WORKSTATION_STATUS.ResourceID = VRS.ResourceId
Left Outer join Computer_System_DATA on Computer_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_X86_PC_MEMORY on v_GS_X86_PC_MEMORY.ResourceID = VRS.ResourceId
Left Outer join v_GS_PROCESSOR on v_GS_PROCESSOR.ResourceID = VRS.ResourceId
Left Outer join v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = VRS.ResourceId
Left Outer join v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
Left Outer join v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID = VRS.ResourceId
Left Outer join v_GS_COMPUTER_SYSTEM on v_GS_COMPUTER_SYSTEM.ResourceID = VRS.ResourceId
Left Outer join v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
Left Outer join v_GS_NETWORK_ADAPTER_CONFIGUR on v_GS_NETWORK_ADAPTER_CONFIGUR.ResourceID =
VRS.ResourceId
left outer join v_GS_LOGICAL_DISK on v_GS_LOGICAL_DISK.ResourceID = Vrs.ResourceId AND
v_GS_LOGICAL_DISK.DriveType0 = 3
Left Outer join v_R_User on VRS.User_Name0 = v_R_User.User_Name0
where VRS.Operating_System_Name_and0 like '%Workstation%'
and (VRS.Obsolete0 = 0 or VRS.Obsolete0 is null)
and VRS.Client0 = 1
and v_FullCollectionMembership.CollectionID = @CollectionID
--and VRS.Netbios_Name0 = 'CLIENT01'
GROUP BY VRS.Netbios_Name0,VRS.Client0,VRS.Active0,v_GS_SYSTEM_ENCLOSURE.ChassisTypes0,
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0,VRS.Client_Version0,Vrs.Creation_Date0,
Vrs.AD_Site_Name0,v_Gs_Operating_System.InstallDate0,v_Gs_Operating_System.LastBootUpTime0,
PC_BIOS_DATA.SerialNumber00,v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0,PC_BIOS_DATA.ReleaseDate00,
PC_BIOS_DATA.Name00,PC_BIOS_DATA.SMBIOSBIOSVersion00,v_GS_PROCESSOR.Name0,Computer_System_DATA.Manufact
urer00,
Computer_System_DATA.Model00,Computer_System_DATA.SystemType00,v_GS_COMPUTER_SYSTEM.Domain0,
Vrs.User_Domain0,Vrs.User_Name0,v_R_User.Mail0,v_GS_COMPUTER_SYSTEM.DomainRole0,Operating_System_DATA.Caption0
0,Operating_System_DATA.CSDVersion00, Operating_System_DATA.Version00, v_GS_X86_PC_MEMORY.TotalPhysicalMemory0,
v_GS_WORKSTATION_STATUS.LastHWScan
order by VRS.Netbios_Name0
```

90. All Workstations Assets Inventory details status

```sql
Declare @CollectionID as varchar(8)
Declare @ProjectName as varchar(25)

Set @CollectionID = 'SMS00001' -- specify scope collection ID
Set @ProjectName = 'LAB' -- specify Project Name

Select
Distinct (VRS.Netbios_Name0) as 'Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 1 Then 'VMWare'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('3','4')Then 'Desktop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('8','9','10','11','12','14') Then 'Laptop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 6 Then 'Mini Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 7 Then 'Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 13 Then 'All in One'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 15 Then 'Space-Saving'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 17 Then 'Main System Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 21 Then 'Peripheral Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 22 Then 'Storage Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 23 Then 'Rack Mount Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 24 Then 'Sealed-Case PC'
Else 'Others' End 'CaseType',
LEFT(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0),
ISNULL(NULLIF(CHARINDEX(',',MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0)) - 1, -
1),LEN(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0))))as 'IPAddress',
MAX (v_GS_NETWORK_ADAPTER_CONFIGUR.MACAddress0) as 'MACAddress',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
VRS.Client_Version0 as 'ClientVersion',
VRS.Creation_Date0 as 'ClientCreationDate',
VRS.AD_Site_Name0 as 'ADSiteName',
dbo.v_GS_OPERATING_SYSTEM.InstallDate0 AS 'OSInstallDate',
DateDiff(D, dbo.v_GS_OPERATING_SYSTEM.InstallDate0, GetDate()) 'OSInstallDateAge',
Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100) as 'LastBootDate',
DateDiff(D, Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100), GetDate()) as
'LastBootDateAge',
PC_BIOS_DATA.SerialNumber00 as 'SerialNumber',
v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0 as 'AssetTag',
PC_BIOS_DATA.ReleaseDate00 as 'ReleaseDate',
PC_BIOS_DATA.Name00 as 'BiosName',
PC_BIOS_DATA.SMBIOSBIOSVersion00 as 'BiosVersion',
v_GS_PROCESSOR.Name0 as 'ProcessorName',
case when Computer_System_DATA.Manufacturer00 like 'VMware%' Then 'VMWare'
when Computer_System_DATA.Manufacturer00 like 'Gigabyte%' Then 'Gigabyte'
when Computer_System_DATA.Manufacturer00 like 'VIA Technologies%' Then 'VIA Technologies'
when Computer_System_DATA.Manufacturer00 like 'MICRO-STAR%' Then 'MICRO-STAR'
Else Computer_System_DATA.Manufacturer00 End 'Manufacturer',
Computer_System_DATA.Model00 as 'Model',
Computer_System_DATA.SystemType00 as 'OSType',
v_GS_COMPUTER_SYSTEM.Domain0 as 'DomainName',
VRS.User_Domain0+'\'+ VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
Case when v_GS_COMPUTER_SYSTEM.domainrole0 = 0 then 'Standalone Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 1 Then 'Member Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 2 Then 'Standalone Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 3 Then 'Member Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 4 Then 'Backup Domain Controller'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 5 Then 'Primary Domain Controller'
End 'Role',
```

```sql
case when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Enterprise Edition'
Then 'Microsoft(R) Windows(R) Server 2003 Enterprise Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Standard Edition' Then
'Microsoft(R) Windows(R) Server 2003 Standard Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Web Edition' Then
'Microsoft(R) Windows(R) Server 2003 Web Edition'
Else Operating_System_DATA.Caption00 End 'OSName',
Operating_System_DATA.CSDVersion00 as 'ServicePack',
Operating_System_DATA.Version00 as 'Version',
((v_GS_X86_PC_MEMORY.TotalPhysicalMemory0/1024)/1000) as 'TotalRAMSize(GB)',
max(v_GS_LOGICAL_DISK.Size0 / 1024) AS 'TotalHDDSize(GB)',
v_GS_WORKSTATION_STATUS.LastHWScan as 'LastHWScan',
DateDiff(D, v_GS_WORKSTATION_STATUS.LastHwScan, GetDate()) as 'LastHWScanAge',
@ProjectName as 'AccountName'
from V_R_System VRS
Left Outer join PC_BIOS_DATA on PC_BIOS_DATA.MachineID = VRS.ResourceId
Left Outer join Operating_System_DATA on Operating_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_WORKSTATION_STATUS on v_GS_WORKSTATION_STATUS.ResourceID = VRS.ResourceId
Left Outer join Computer_System_DATA on Computer_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_X86_PC_MEMORY on v_GS_X86_PC_MEMORY.ResourceID = VRS.ResourceId
Left Outer join v_GS_PROCESSOR on v_GS_PROCESSOR.ResourceID = VRS.ResourceId
Left Outer join v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = VRS.ResourceId
Left Outer join v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
Left Outer join v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID =
VRS.ResourceId
Left Outer join v_GS_COMPUTER_SYSTEM on v_GS_COMPUTER_SYSTEM.ResourceID = VRS.ResourceId
Left Outer join v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
Left Outer join v_GS_NETWORK_ADAPTER_CONFIGUR on v_GS_NETWORK_ADAPTER_CONFIGUR.ResourceID =
VRS.ResourceId
left outer join v_GS_LOGICAL_DISK on v_GS_LOGICAL_DISK.ResourceID = Vrs.ResourceId AND
v_GS_LOGICAL_DISK.DriveType0 = 3
Left Outer join v_R_User on VRS.User_Name0 = v_R_User.User_Name0
where VRS.Operating_System_Name_and0 like '%Server%'
and (VRS.Obsolete0 = 0 or VRS.Obsolete0 is null)
and VRS.Client0 = 1
and v_FullCollectionMembership.CollectionID = @CollectionID
--and VRS.Netbios_Name0 = 'CLIENT01'
GROUP BY VRS.Netbios_Name0,VRS.Client0,VRS.Active0,v_GS_SYSTEM_ENCLOSURE.ChassisTypes0,
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0,VRS.Client_Version0,Vrs.Creation_Date0,
Vrs.AD_Site_Name0,v_Gs_Operating_System.InstallDate0,v_Gs_Operating_System.LastBootUpTime0,
PC_BIOS_DATA.SerialNumber00,v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0,PC_BIOS_DATA.ReleaseDate00,
PC_BIOS_DATA.Name00,PC_BIOS_DATA.SMBIOSBIOSVersion00,v_GS_PROCESSOR.Name0,Computer_System_DATA.Manufact
urer00,
Computer_System_DATA.Model00,Computer_System_DATA.SystemType00,v_GS_COMPUTER_SYSTEM.Domain0,
Vrs.User_Domain0,Vrs.User_Name0,v_R_User.Mail0,v_GS_COMPUTER_SYSTEM.DomainRole0,Operating_System_DATA.C
aption00,Operating_System_DATA.CSDVersion00, Operating_System_DATA.Version00,
v_GS_X86_PC_MEMORY.TotalPhysicalMemory0, v_GS_WORKSTATION_STATUS.LastHWScan
order by VRS.Netbios_Name0
```

91. All PCs with Office 365 Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
App.ARPDisplayName0 as 'DisplayName',
```

```sql
App.InstallDate0  as 'InstalledDate',
App.ProductVersion0 as 'Version'
from V_R_System VRS
LEFT JOIN v_GS_INSTALLED_SOFTWARE App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_FullCollectionMembership as Col on VRS.ResourceID = Col.ResourceID
LEFT JOIN v_R_User on VRS.User_Name0 = v_R_User.User_Name0
Where App.ARPDisplayName0 like 'Microsoft Office 365%'
and App.ProductVersion0 like '15.%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and VRS.Client0 = 1 and VRS.Obsolete0 = 0
order by VRS.Name0, App.ProductVersion0
```

92. All PCs without Office 365 Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
distinct(vs.Name0)  as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vs.AD_Site_Name0 as 'ADSite',
vs.Full_Domain_Name0 as 'Domain',
vs.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge'
from v_R_System vs
LEFT JOIN v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = vs.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vs.ResourceID = Os.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = vs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on vs.ResourceID = HWSCAN.ResourceID
LEFT JOIN Computer_System_DATA St on vs.ResourceID = st.MachineID
LEFT JOIN v_R_User on vs.User_Name0 = v_R_User.User_Name0
LEFT JOIN v_GS_INSTALLED_SOFTWARE  App ON vs.ResourceID = App.ResourceID
where Vs.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @Collection
and Vs.Client0 = 1 and Vs.Obsolete0 = 0
and vs.ResourceID Not in (
Select Vrs.ResourceID from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE  App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
Where App.ARPDisplayName0 like 'Microsoft Office 365%'
and App.ProductVersion0 like '15.%'
)
```

93. All PCs with SEP Antivirus Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
App.ARPDisplayName0 as 'DisplayName',
App.InstallDate0  as 'InstalledDate',
App.ProductVersion0 as 'Version'
```

```sql
from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT Join v_FullCollectionMembership as Col on VRS.ResourceID = Col.ResourceID
LEFT JOIN v_R_User on VRS.User_Name0 = v_R_User.User_Name0
Where App.ARPDisplayName0 like 'Symantec%Endpoint%Protection%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and VRS.Client0 = 1 and VRS.Obsolete0 = 0
order by VRS.Name0, App.ProductVersion0
```

94. All PCs without SEP Antivirus Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
distinct(vs.Name0)  as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vs.AD_Site_Name0 as 'ADSite',
vs.Full_Domain_Name0 as 'Domain',
vs.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge'
from v_R_System vs
LEFT JOIN v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = vs.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vs.ResourceID = Os.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = vs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on vs.ResourceID = HWSCAN.ResourceID
LEFT JOIN Computer_System_DATA St on vs.ResourceID = st.MachineID
LEFT JOIN v_R_User on vs.User_Name0 = v_R_User.User_Name0
LEFT JOIN v_GS_INSTALLED_SOFTWARE  App ON vs.ResourceID = App.ResourceID
where Vs.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @Collection
and Vs.Client0 = 1 and Vs.Obsolete0 = 0
and vs.ResourceID Not in (
Select Vrs.ResourceID from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE  App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
Where App.ARPDisplayName0 like 'Symantec%Endpoint%Protection%'
)
```

95. All Workstations Client Agent Detailed Report

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

select distinct(Name),Case when IsClient= 1 then 'Healthy' else 'UnHealthy' end as 'AgentStatus',
(select  case when count (v_GS_WORKSTATION_STATUS.ResourceID)=1 then 'Healthy' else  'UnHealthy' end
from v_GS_WORKSTATION_STATUS where DATEDIFF (day,LastHWScan,GetDate())<30 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'HWScanStatus',
(select case when count (v_GS_LastSoftwareScan.ResourceID)=1 then 'Healthy' else 'UnHealthy' end
from v_GS_LastSoftwareScan where DATEDIFF (day,LastScanDate,GetDate())<30 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'SWScanStatus',
(select case when count (v_UpdateScanStatus.ResourceID)=1 then 'Healthy' else 'UnHealthy' end
from v_UpdateScanStatus where DATEDIFF (day,LastScanTime,GetDate())<30 and LastErrorCode = 0 and
ResourceID=v_FullCollectionMembership.ResourceID)
as 'WSUSScanStatus',
(select DATEDIFF (day,LastHWScan,GetDate()) from v_GS_WORKSTATION_STATUS
where ResourceID=v_FullCollectionMembership.ResourceID)
```

```sql
as 'LastHWScanDays',
(select DATEDIFF (day,LastScanDate,GetDate()) from v_GS_LastSoftwareScan
where ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastSWScanDays',
(select DATEDIFF (day,LastScanTime,GetDate()) from v_UpdateScanStatus
where LastErrorCode = 0 and ResourceID=v_FullCollectionMembership.ResourceID)
as 'LastWSUSScanDays'
from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID in ( select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%')
order by 2 desc
```

96. All Workstations Low Free Disk Space Report

```sql
Declare @CollectionID as Varchar(8)
Declare @FreeSpace as Integer
Set @CollectionID = 'SMS00001' -- specify scope collection ID
Set @FreeSpace = '5000' -- specify MB Size

Select
distinct (Vrs.Name0) as 'Machine',
Vrs.AD_Site_Name0 as 'ADSiteName',
Vrs.User_Name0 as 'UserName',
USR.Mail0 as 'EMailID',
Os.Caption00 as 'OSName',
Csd.SystemType00 as 'OSType',
LD.DeviceID00 as 'Drive',
LD.FileSystem00 as 'FileSystem',
LD.Size00 / 1024 as 'TotalSpace (GB)',
LD.FreeSpace00 / 1024 as 'FreeSpace (GB)',
Ws.LastHWScan as 'LastHWScan',
DateDiff(D, Ws.LastHwScan, GetDate()) as 'LastHWScanAge'
FROM v_R_System Vrs
Join  v_R_User USR on USR.User_Name0 = Vrs.User_Name0
Join v_FullCollectionMembership  Fc on Fc.ResourceID = Vrs.ResourceID
Join Operating_System_DATA Os on Os.MachineID = Vrs.ResourceID
Join Computer_System_DATA Csd on Csd.MachineID = Vrs.ResourceID
Join Logical_Disk_Data Ld on Ld.MachineID = Vrs.ResourceID
Join v_GS_WORKSTATION_STATUS Ws on Ws.ResourceID = Vrs.ResourceId
where CollectionID = @CollectionID
and LD.Description00 = 'Local Fixed Disk'
and LD.FreeSpace00 < @FreeSpace
and Vrs.Operating_System_Name_and0 like '%Workstation%'
Order By Vrs.Name0 asc
```

97. All Servers Low Free Disk Space Report

```sql
Declare @CollectionID as Varchar(8)
Declare @FreeSpace as Integer
Set @CollectionID = 'SMS00001' -- specify scope collection ID
Set @FreeSpace = '5000' -- specify MB Size

Select
distinct (Vrs.Name0) as 'Machine',
Vrs.AD_Site_Name0 as 'ADSiteName',
Vrs.User_Name0 as 'UserName',
USR.Mail0 as 'EMailID',
Os.Caption00 as 'OSName',
Csd.SystemType00 as 'OSType',
LD.DeviceID00 as 'Drive',
LD.FileSystem00 as 'FileSystem',
LD.Size00 / 1024 as 'TotalSpace (GB)',
LD.FreeSpace00 / 1024 as 'FreeSpace (GB)',
Ws.LastHWScan as 'LastHWScan',
```

```
DateDiff(D, Ws.LastHwScan, GetDate()) as 'LastHWScanAge'
FROM v_R_System Vrs
Join  v_R_User USR on USR.User_Name0 = Vrs.User_Name0
Join v_FullCollectionMembership  Fc on Fc.ResourceID = Vrs.ResourceID
Join Operating_System_DATA Os on Os.MachineID = Vrs.ResourceID
Join Computer_System_DATA Csd on Csd.MachineID = Vrs.ResourceID
Join Logical_Disk_Data Ld on Ld.MachineID = Vrs.ResourceID
Join v_GS_WORKSTATION_STATUS Ws on Ws.ResourceID = Vrs.ResourceId
where CollectionID = @CollectionID
and LD.Description00 = 'Local Fixed Disk'
and LD.FreeSpace00 < @FreeSpace
and Vrs.Operating_System_Name_and0 like '%Server%'
Order By Vrs.Name0 asc
```

98. All Workstations Machines Names Last Logon with Serial No Report

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct (VRS.Netbios_Name0) as 'Name',
PC_BIOS_DATA.SerialNumber00 as 'SerialNumber',
VRS.User_Domain0+'\'+ VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID'
from V_R_System VRS
Left Outer join PC_BIOS_DATA on PC_BIOS_DATA.MachineID = VRS.ResourceId
Left Outer join Operating_System_DATA on Operating_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
left outer join v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
Left Outer join v_R_User on VRS.User_Name0 = v_R_User.User_Name0
where VRS.Operating_System_Name_and0 like '%Workstation%'
and (VRS.Obsolete0 = 0 or VRS.Obsolete0 is null)
and VRS.Client0 = 1
and v_FullCollectionMembership.CollectionID = @CollectionID
GROUP BY VRS.Netbios_Name0, VRS.Client0, PC_BIOS_DATA.SerialNumber00, Vrs.User_Domain0,
Vrs.User_Name0, v_R_User.Mail0
Order by VRS.Netbios_Name0
```

99. All Workstations with Adobe Acrobat Reader Installed Machines Report

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
Vru.Mail0 as 'EMailID',
VRS.AD_Site_Name0 as 'ADSite',
ARP.Publisher0 as 'Publisher',
ARP.ARPDisplayName0 as 'DisplayName',
ARP.PackageCode0 as 'ProductID',
ARP.InstallDate0 as 'InstalledDate',
ARP.ProductVersion0 as 'Version'
from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
LEFT JOIN v_R_User Vru on VRS.User_Name0 = Vru.User_Name0
where ARP.Publisher0 like '%Adobe%'
and
ARP.ARPDisplayName0 like '%Acrobat%Reader%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
```

```
and v_FullCollectionMembership.CollectionID = @CollectionID
and VRS.Obsolete0 = 0
order by 12
```

100.   All Workstations with Adobe Acrobat Reader Last Usage Machines Report

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

Declare @days float
Declare @__timezoneoffset int
Select @__timezoneoffset = DateDiff (ss,GetUTCDate(),Getdate())
Select @days=DATEDIFF(day,IntervalStart,DATEADD(month,1,IntervalStart))from v_SummarizationInterval
If IsNULL(@days,0) > 0
Select
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vrs.Full_Domain_Name0 as 'Domain',
Vrs.User_Name0 as 'UserName',
Vrs.AD_Site_Name0 as 'ADSite',
ARP.Publisher0 as 'Publisher',
ARP.DisplayName0 as 'DisplayName',
ARP.ProdID0 as 'ProductID',
ARP.InstallDate0  as 'InstalledDate',
ARP.Version0 as 'Version',
DATEADD(ss,@__timezoneoffset,MAX(mus.LastUsage)) as 'LastUsage',
DateDiff(D, DATEADD(ss,@__timezoneoffset,MAX(mus.LastUsage)), GetDate()) as 'LastUsageDays'
from V_R_System Vrs
LEFT JOIN v_GS_ADD_REMOVE_PROGRAMS  ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_MonthlyUsageSummary mus on Vrs.ResourceID=mus.ResourceID
LEFT JOIN v_MeteredFiles mf on mus.FileID=mf.MeteredFileID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
Where ARP.Publisher0 like '%Adobe%'
and ARP.DisplayName0 like '%Acrobat%Reader%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @CollectionID
and Vrs.Obsolete0 = 0 and Vrs.Client0 = 1
Group by
Vrs.Name0,Os.Caption0,St.SystemType00,Vrs.Full_Domain_Name0,Vrs.User_Name0,Vrs.AD_Site_Name0,ARP.Publisher0,ARP.Inst
allDate0,ARP.DisplayName0,ARP.ProdID0,ARP.Version0
Having SUM (UsageCount) + SUM (TSUsageCount) > 0
Order by Vrs.Name0
```

101.   All Workstations with Adobe Products Not Used More Than 90 Days Machines Report

```
Declare @RequirdDays as Integer
Declare @days float
Declare @__timezoneoffset int
Select @__timezoneoffset = DateDiff (ss,GetUTCDate(),Getdate())
Select @days=DATEDIFF(day,IntervalStart,DATEADD(month,1,IntervalStart))from v_SummarizationInterval

Set @RequirdDays = 90 -- specify Days

If IsNULL(@days,0) > 0
Select
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vrs.Full_Domain_Name0 as 'Domain',
Vrs.User_Name0 as 'UserName',
Vrs.AD_Site_Name0 as 'ADSite',
ARP.Publisher0 as 'Publisher',
```

```sql
ARP.DisplayName0 as 'DisplayName',
ARP.ProdID0 as 'ProductID',
ARP.InstallDate0 as 'InstalledDate',
ARP.Version0 as 'Version',
DATEADD(ss,@__timezoneoffset,MAX(mus.LastUsage)) as 'LastUsage',
DateDiff(D, DATEADD(ss,@__timezoneoffset,MAX(mus.LastUsage)), GetDate()) as 'LastUsageDays'
from V_R_System Vrs
LEFT JOIN v_GS_ADD_REMOVE_PROGRAMS  ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_MonthlyUsageSummary mus on Vrs.ResourceID=mus.ResourceID
LEFT JOIN v_MeteredFiles mf on mus.FileID=mf.MeteredFileID
where
ARP.Publisher0 like '%Adobe%'
and (DATEDIFF(d,LastUsage, GETDATE()) > @RequirdDays)
and
(
ARP.DisplayName0 like '%Adobe%Acrobat%Pro%'
or ARP.DisplayName0 like '%Adobe%InDesign%'
or ARP.DisplayName0 like '%Adobe%Illustrator%'
or ARP.DisplayName0 like '%Adobe%Illustrator%'
or ARP.DisplayName0 like '%Adobe%Photoshop%'
)
and VRS.Operating_System_Name_and0 like '%Workstation%'
and Vrs.Obsolete0=0 and Vrs.Client0=1
Group by
Vrs.Name0,Os.Caption0,St.SystemType00,Vrs.Full_Domain_Name0,Vrs.User_Name0,Vrs.AD_Site_Name0,ARP.Publisher0,ARP.Inst
allDate0,ARP.DisplayName0,ARP.ProdID0,ARP.Version0
having SUM(UsageCount) + SUM(TSUsageCount) > 0
order by DateDiff(D, DATEADD(ss,@__timezoneoffset,MAX(mus.LastUsage)), GetDate())
```

102.   All Client and Inventory Health Report

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalMachines as Numeric(5)
Declare @Healthy as Numeric(5)
Declare @UnHealthy as Numeric(5)
Declare @HWInventorySuccess as Numeric(5)
Declare @HWInventoryNotRun as Numeric(5)
Declare @SWInventorySuccess as Numeric(5)
Declare @SWInventoryNotRun as Numeric(5)
Declare @WSUSScanSuccess as Numeric(5)
Declare @WSUSScanNotRun as Numeric(5)

Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select @TotalMachines = (select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID)

select @Healthy = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete != 1 and IsClient = 1
)

select @UnHealthy = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID Not in (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete != 1 and IsClient = 1 )
)

select @HWInventorySuccess = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where (DATEDIFF (day,LastHWScan,GetDate())<30)
and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
)
select @HWInventoryNotRun = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
```

```sql
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where (DATEDIFF (day,LastHWScan,GetDate())<30)
and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
)
)

select @SWInventorySuccess = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan where (DATEDIFF (day,LastScanDate,GetDate())<30) and
ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
)
select @SWInventoryNotRun = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan where (DATEDIFF (day,LastScanDate,GetDate())<30) and
ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
)
)
select @WSUSScanSuccess = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and (DATEDIFF
(day,LastScanTime,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat
Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)

select @WSUSScanNotRun = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and (DATEDIFF
(day,LastScanTime,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat
Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)
)

select
@TotalMachines as 'TotalMachines',
@Healthy as 'Healthy',
@UnHealthy as 'UnHealthy',
(Select(@Healthy/@TotalMachines)*100) as 'Healthy%',
@HWInventorySuccess as 'HWInventorySuccess',
@HWInventoryNotRun as 'HWInventoryNotRun',
(Select(@HWInventorySuccess/@Healthy)*100) as 'HWInventorySuccess%',
@SWInventorySuccess as 'SWInventorySuccess',
@SWInventoryNotRun as 'SWInventoryNotRun',
(Select(@SWInventorySuccess/@Healthy)*100) as 'SWInventorySuccess%',
@WSUSScanSuccess as 'WSUSScanSuccess',
@WSUSScanNotRun as 'WSUSScanNotRun',
(Select(@WSUSScanSuccess/@Healthy)*100) as 'WSUSScanSuccess%'
```

103.   All Total Scope machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID)
```

104.   All Total Healthy machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID
```

```sql
select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete != 1 and IsClient = 1)
```

105.   All Total Unhealthy machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and ResourceID Not in (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete != 1 and IsClient = 1))
```

106.   All Total Hardware Inventory within 30 Days machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where (DATEDIFF (day,LastHWScan,GetDate())<30)
and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
)
```

107.   All Total Hardware Inventory not within 30 Days machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where (DATEDIFF (day,LastHWScan,GetDate())<30)
and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
))
```

108.   All Total Software Inventory within 30 Days machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan
where (DATEDIFF (day, LastScanDate,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where
AgentName in ('Heartbeat Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)
```

109.   All Total Software Inventory not within 30 Days machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID
```

```sql
select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_LastSoftwareScan where (DATEDIFF (day,LastScanDate,GetDate())<30) and
ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat Discovery') and DATEDIFF
(day,AgentTime,GetDate())<30))
))
```

## 110.	All Total WSUS Scan within 30 Days machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and (DATEDIFF
(day,LastScanTime,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat
Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)
```

## 111.	All Total WSUS Scan not within 30 Days machines details

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- Specify Scope Collection ID

select
Name0 as 'MachineName'
from v_R_system where V_R_System.ResourceID
in ( select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and (DATEDIFF
(day,LastScanTime,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where AgentName in ('Heartbeat
Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
))
```

## 112.	All Deployments status for Specific Application

```sql
Declare @ApplicationName as Varchar(255)
Set @ApplicationName = 'Adobe DIO Illustrator Extension 3.0.13.2'     --Specify Application Name

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else 'Others' End as
'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' when Ds.DeploymentIntent = 3
Then 'Simulate' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
```

```sql
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess + Ds.NumberOther) /
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1
and Vaa.ApplicationName = @ApplicationName
order by Ds.DeploymentTime desc
```

113.   All Deployments status for Specific Package

```sql
Declare @PackageName as Varchar(255)
Set @PackageName = 'SAP Client GUI 7.30' --Specify Package Name

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1) as 'PackageName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess + Ds.NumberOther) /
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2
and Ds.SoftwareName like @PackageName + '%'
order by Ds.DeploymentTime desc
```

114.   All Deployments status for Specific Software Update Group

```sql
Declare @SoftwareUpdateGroupName as varchar(255)
Set @SoftwareUpdateGroupName = 'All Updates SRWP2'        --Specify Software Update Group Name

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Li.Title as 'SUGroupName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
```

```sql
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
left join v_AuthListInfo LI on LI.ModelID = Ds.ModelID
Where Ds.FeatureType = 5
and Li.Title like @SoftwareUpdateGroupName
order by Ds.DeploymentTime desc
```

115.   All Deployments status for Specific Task Sequence

```sql
Declare @TaskSequenceName as varchar(255)
Set @TaskSequenceName = 'Windows 7 x64-CoreImage'    --Specify TaskSequenceName

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess + Ds.NumberOther) /
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7
and Ds.SoftwareName = @TaskSequenceName
order by Ds.DeploymentTime desc
```

116.   Deployment Detailed status for specific application with specific collection

```sql
Declare @ApplicationName as Varchar(255)
Declare @CollectionName as Varchar(255)
Set @ApplicationName = 'Adobe Creative-Cloud Design-Standard 2015 (64-bit)' -- Specify Application name
Set @CollectionName = '%Adobe Creative-Cloud Design-Standard 2015%' -- Specify Application name

select
aa.ApplicationName as 'Application Name',
aa.CollectionName as 'Target Collection',
ae.descript as 'DeploymentTypeName',
```

```sql
s1.netbios_name0 as 'ComputerName',
s1.AD_Site_Name0 as 'ADSiteName',
case when ae.AppEnforcementState = 1000 then 'Success'
when ae.AppEnforcementState = 1001 then 'Already Compliant'
when ae.AppEnforcementState = 1002 then 'Simulate Success'
when ae.AppEnforcementState = 2000 then 'In Progress'
when ae.AppEnforcementState = 2001 then 'Waiting for Content'
when ae.AppEnforcementState = 2002 then 'Installing'
when ae.AppEnforcementState = 2003 then 'Restart to Continue'
when ae.AppEnforcementState = 2004 then 'Waiting for maintenance window'
when ae.AppEnforcementState = 2005 then 'Waiting for schedule'
when ae.AppEnforcementState = 2006 then 'Downloading dependent content'
when ae.AppEnforcementState = 2007 then 'Installing dependent content'
when ae.AppEnforcementState = 2008 then 'Restart to complete'
when ae.AppEnforcementState = 2009 then 'Content downloaded'
when ae.AppEnforcementState = 2010 then 'Waiting for update'
when ae.AppEnforcementState = 2011 then 'Waiting for user session reconnect'
when ae.AppEnforcementState = 2012 then 'Waiting for user logoff'
when ae.AppEnforcementState = 2013 then 'Waiting for user logon'
when ae.AppEnforcementState = 2014 then 'Waiting to install'
when ae.AppEnforcementState = 2015 then 'Waiting retry'
when ae.AppEnforcementState = 2016 then 'Waiting for presentation mode'
when ae.AppEnforcementState = 2017 then 'Waiting for Orchestration'
when ae.AppEnforcementState = 2018 then 'Waiting for network'
when ae.AppEnforcementState = 2019 then 'Pending App-V Virtual Environment'
when ae.AppEnforcementState = 2020 then 'Updating App-V Virtual Environment'
when ae.AppEnforcementState = 3000 then 'Requirements not met'
when ae.AppEnforcementState = 3001 then 'Host platform not applicable'
when ae.AppEnforcementState = 4000 then 'Unknown'
when ae.AppEnforcementState = 5000 then 'Deployment failed'
when ae.AppEnforcementState = 5001 then 'Evaluation failed'
when ae.AppEnforcementState = 5002 then 'Deployment failed'
when ae.AppEnforcementState = 5003 then 'Failed to locate content'
when ae.AppEnforcementState = 5004 then 'Dependency installation failed'
when ae.AppEnforcementState = 5005 then 'Failed to download dependent content'
when ae.AppEnforcementState = 5006 then 'Conflicts with another application deployment'
when ae.AppEnforcementState = 5007 then 'Waiting retry'
when ae.AppEnforcementState = 5008 then 'Failed to uninstall superseded deployment type'
when ae.AppEnforcementState = 5009 then 'Failed to download superseded deployment type'
when ae.AppEnforcementState = 5010 then 'Failed to updating App-V Virtual Environment'
End as 'State Message',
case when ae.AppEnforcementState like '10%' then 'Success'
when ae.AppEnforcementState like '20%' then 'Progress'
when ae.AppEnforcementState like '30%' then 'ReqNotMet'
when ae.AppEnforcementState like '40%' then 'Unknown'
when ae.AppEnforcementState like '50%' then 'Failed'
End as 'Status',
LastComplianceMessageTime as 'LastMessageTime'
from v_R_System_Valid s1
join vAppDTDeploymentResultsPerClient ae on ae.ResourceID = s1.ResourceID
join v_CICurrentComplianceStatus ci2 on ci2.CI_ID = ae.CI_ID AND ci2.ResourceID=s1.ResourceID
join v_ApplicationAssignment aa on ae.AssignmentID = aa.AssignmentID
where ae.AppEnforcementState is not null
and (aa.ApplicationName = @ApplicationName
and aa.CollectionName like @CollectionName)
order by ae.AppEnforcementState, LastComplianceMessageTime Desc
```

117.   Deployment Detailed status for specific application

```sql
Declare @ApplicationName as Varchar(255)
Set @ApplicationName = 'Adobe Creative-Cloud Design-Standard 2015 (64-bit)' -- Specify Application name

select
aa.ApplicationName as 'Application Name',
aa.CollectionName as 'Target Collection',
ae.descript as 'DeploymentTypeName',
```

```sql
s1.netbios_name0 as 'ComputerName',
s1.AD_Site_Name0 as 'ADSiteName',
case when ae.AppEnforcementState = 1000 then 'Success'
when ae.AppEnforcementState = 1001 then 'Already Compliant'
when ae.AppEnforcementState = 1002 then 'Simulate Success'
when ae.AppEnforcementState = 2000 then 'In Progress'
when ae.AppEnforcementState = 2001 then 'Waiting for Content'
when ae.AppEnforcementState = 2002 then 'Installing'
when ae.AppEnforcementState = 2003 then 'Restart to Continue'
when ae.AppEnforcementState = 2004 then 'Waiting for maintenance window'
when ae.AppEnforcementState = 2005 then 'Waiting for schedule'
when ae.AppEnforcementState = 2006 then 'Downloading dependent content'
when ae.AppEnforcementState = 2007 then 'Installing dependent content'
when ae.AppEnforcementState = 2008 then 'Restart to complete'
when ae.AppEnforcementState = 2009 then 'Content downloaded'
when ae.AppEnforcementState = 2010 then 'Waiting for update'
when ae.AppEnforcementState = 2011 then 'Waiting for user session reconnect'
when ae.AppEnforcementState = 2012 then 'Waiting for user logoff'
when ae.AppEnforcementState = 2013 then 'Waiting for user logon'
when ae.AppEnforcementState = 2014 then 'Waiting to install'
when ae.AppEnforcementState = 2015 then 'Waiting retry'
when ae.AppEnforcementState = 2016 then 'Waiting for presentation mode'
when ae.AppEnforcementState = 2017 then 'Waiting for Orchestration'
when ae.AppEnforcementState = 2018 then 'Waiting for network'
when ae.AppEnforcementState = 2019 then 'Pending App-V Virtual Environment'
when ae.AppEnforcementState = 2020 then 'Updating App-V Virtual Environment'
when ae.AppEnforcementState = 3000 then 'Requirements not met'
when ae.AppEnforcementState = 3001 then 'Host platform not applicable'
when ae.AppEnforcementState = 4000 then 'Unknown'
when ae.AppEnforcementState = 5000 then 'Deployment failed'
when ae.AppEnforcementState = 5001 then 'Evaluation failed'
when ae.AppEnforcementState = 5002 then 'Deployment failed'
when ae.AppEnforcementState = 5003 then 'Failed to locate content'
when ae.AppEnforcementState = 5004 then 'Dependency installation failed'
when ae.AppEnforcementState = 5005 then 'Failed to download dependent content'
when ae.AppEnforcementState = 5006 then 'Conflicts with another application deployment'
when ae.AppEnforcementState = 5007 then 'Waiting retry'
when ae.AppEnforcementState = 5008 then 'Failed to uninstall superseded deployment type'
when ae.AppEnforcementState = 5009 then 'Failed to download superseded deployment type'
when ae.AppEnforcementState = 5010 then 'Failed to updating App-V Virtual Environment'
End as 'State Message',
case when ae.AppEnforcementState like '10%' then 'Success'
when ae.AppEnforcementState like '20%' then 'Progress'
when ae.AppEnforcementState like '30%' then 'ReqNotMet'
when ae.AppEnforcementState like '40%' then 'Unknown'
when ae.AppEnforcementState like '50%' then 'Failed'
End as 'Status',
LastComplianceMessageTime as 'LastMessageTime'
from v_R_System_Valid s1
join vAppDTDeploymentResultsPerClient ae on ae.ResourceID = s1.ResourceID
join v_CICurrentComplianceStatus ci2 on ci2.CI_ID = ae.CI_ID AND ci2.ResourceID=s1.ResourceID
join v_ApplicationAssignment aa on ae.AssignmentID = aa.AssignmentID
where ae.AppEnforcementState is not null
and aa.ApplicationName = @ApplicationName
order by ae.AppEnforcementState, LastComplianceMessageTime Desc
```

118.   Deployment Detailed status for specific package with specific collection

```sql
Declare @PackageName as Varchar(255)
Declare @CollectionName as Varchar(255)

Set @PackageName = 'Lync 2013' -- Specify Application name
Set @CollectionName = '%Lync%' -- Specify Collection name

select distinct
pack.Name as 'Package Name',
```

```sql
COLL.CollectionName as 'Target Collection',
adv.ProgramName as 'DeploymentTypeName',
Vrs.Name0 as 'ComputerName',
Vrs.AD_Site_Name0 as 'ADSiteName',
ADV.AdvertisementID as 'AdvertisementID',
ADVS.LastStateName as 'Status',
ADVS.LastStatusTime as 'Status Time'
from v_R_System Vrs
INNER JOIN vSMS_ClientAdvertisementStatus ADVS ON Vrs.ResourceID = ADVS.ResourceID
INNER JOIN v_Advertisement ADV ON ADV.AdvertisementID = ADVS.AdvertisementID
INNER JOIN v_FullCollectionMembership CM ON Vrs.ResourceID = CM.ResourceID
LEFT JOIN v_Package Pack on adv.PackageID = pack.PackageID
LEFT JOIN v_Collections COLL ON ADV.CollectionID = COLL.SiteID
where pack.Name = @PackageName
and COLL.CollectionName like @CollectionName
Order by Vrs.Name0
```

119.   Deployment Detailed status for specific package

```sql
Declare @PackageName as Varchar(255)
Set @PackageName = 'Lync 2013' -- Specify Application name

select distinct
pack.Name as 'Package Name',
COLL.CollectionName as 'Target Collection',
adv.ProgramName as 'DeploymentTypeName',
Vrs.Name0 as 'ComputerName',
Vrs.AD_Site_Name0 as 'ADSiteName',
ADV.AdvertisementID as 'AdvertisementID',
ADVS.LastStateName as 'Status',
ADVS.LastStatusTime as 'Status Time'
from v_R_System Vrs
INNER JOIN vSMS_ClientAdvertisementStatus ADVS ON Vrs.ResourceID = ADVS.ResourceID
INNER JOIN v_Advertisement ADV ON ADV.AdvertisementID = ADVS.AdvertisementID
INNER JOIN v_FullCollectionMembership CM ON Vrs.ResourceID = CM.ResourceID
LEFT JOIN v_Package Pack on adv.PackageID = pack.PackageID
LEFT JOIN v_Collections COLL ON ADV.CollectionID = COLL.SiteID
where pack.Name = @PackageName
Order by Vrs.Name0
```

120.   Deployment Detailed status for specific software Update deployment

```sql
Declare @DeploymentName as Varchar(255)
Set @DeploymentName = 'SU WKS-All Production Computers' -- Specify Software update deployment name

select
vrs.Name0,
vrs.Active0,
vrs.AD_Site_Name0,
vrs.User_Name0,
vrs.Operating_System_Name_and0,
a.Assignment_UniqueID as DeploymentID,
a.AssignmentName as DeploymentName,
a.StartTime as Available,
a.EnforcementDeadline as Deadline,
sn.StateName as LastEnforcementState,
wsus.LastErrorCode as 'LasErrorCode',
wsus.LastScanTime as 'LastWSUSScan',
DateDiff(D, wsus.LastScanTime, GetDate()) as 'LastWSUSScan Age',
wks.LastHWScan,
DateDiff(D, wks.LastHwScan, GetDate()) as 'LastHWScan Age'
from v_CIAssignment a
join v_AssignmentState_Combined assc on a.AssignmentID=assc.AssignmentID
join v_StateNames sn on assc.StateType = sn.TopicType and sn.StateID=isnull(assc.StateID,0)
join v_R_System vrs on vrs.ResourceID = assc.ResourceID
join v_GS_WORKSTATION_STATUS wks on wks.ResourceID = assc.ResourceID
join v_UpdateScanStatus wsus on wsus.ResourceID = assc.ResourceID
where a.AssignmentName = @DeploymentName
```

```sql
and assc.StateType in (300,301)
order by 11 desc
```

121. All Collections with RefreshType

```sql
Select
CollectionID as 'ColletionID',
Name as 'ColletionName',
case Refreshtype
when 1 then 'NoScheduled Update'
when 2 then 'Full Scheduled Update'
when 4 then 'Incremental Update Update(Only)'
when 6 then 'Incremental and Full Scheduled Update'
End as 'Refreshtype'
from v_Collection
order by refreshtype
```

122. All Software Inventory Report for Specific Computer Based on Installed Software Class

```sql
Select sys.Netbios_Name0,
vos.Caption0 AS [Operating System],
varp.ARPDisplayName0 AS [Software Name],
varp.Publisher0,
varp.ProductVersion0,
varp.TimeStamp,
varp.InstallDate0,
Varp.ProductID0,
fcm.SiteCode,
sys.User_Name0,
sys.User_Domain0
FROM v_R_System sys
JOIN v_GS_INSTALLED_SOFTWARE varp ON sys.ResourceID = varp.ResourceID
JOIN v_FullCollectionMembership fcm on sys.ResourceID=fcm.ResourceID
JOIN v_GS_OPERATING_SYSTEM vos on sys.ResourceID=vos.ResourceID
WHERE ARPDisplayName0 LIKE '%' and sys.Name0 in ('Client01')
```

123. All Applications Deployments Status for Specific Computers

```sql
select
s1.netbios_name0 as 'ComputerName',
aa.ApplicationName as 'ApplicationName',
'Application' as 'ApplicationType',
case when ae.AppEnforcementState like '10%' then 'Success'
when ae.AppEnforcementState like '20%' then 'Progress'
when ae.AppEnforcementState like '30%' then 'ReqNotMet'
when ae.AppEnforcementState like '40%' then 'Unknown'
when ae.AppEnforcementState like '50%' then 'Failed'
End as 'Status',
LastComplianceMessageTime as 'LastMessageTime'
from v_R_System_Valid s1
join vAppDTDeploymentResultsPerClient ae on ae.ResourceID = s1.ResourceID
join v_CICurrentComplianceStatus ci2 on ci2.CI_ID = ae.CI_ID AND ci2.ResourceID=s1.ResourceID
join v_ApplicationAssignment aa on ae.AssignmentID = aa.AssignmentID
where ae.AppEnforcementState is not null
and (
s1.netbios_name0 in ('Client01',' Client02')
)
order by ae.AppEnforcementState, LastComplianceMessageTime Desc
```

124. Specific Software Update Deployment Failed Errors with Description

```sql
Declare @DeploymentName as Varchar(255)
Set @DeploymentName = 'SU WKS-All Production Computers'

Select
Vrs.Name0 as 'MachineName',
Vrs.User_Name0 as 'UserName',
```

```sql
assc.LastEnforcementMessageTime as 'LastEnforcementTime',
assc.LastEnforcementErrorID&0x0000FFFF as 'ErrorStatusID',
isnull(master.dbo.fn_varbintohexstr(CONVERT(VARBINARY(8),CONVERT(int,assc.LastEnforcementErrorCode))), 0) as 'ErrorCode',
assc.LastEnforcementErrorCode as 'ErrorCodeInt',
Asi.MessageName as 'MessageName'
from V_CIAssignment  cia
join V_UpdateAssignmentStatus_Live  assc on assc.AssignmentID = cia.AssignmentID
inner join v_AssignmentState_Combined Ac on Ac.ResourceID = assc.ResourceID
inner join V_AdvertisementStatusInformation Asi on Asi.MessageID = Ac.LastStatusMessageID
and isnull(assc.IsCompliant, 0) = 0
and assc.LastEnforcementMessageID in (6,9)
and assc.LastEnforcementErrorCode Not in (0)
join v_R_System Vrs on assc.ResourceID = Vrs.ResourceID and isnull (Vrs.Obsolete0,0) = 0
where cia.AssignmentName =  @DeploymentName
```

125.   AL_Designer Exe File and Path with version - Based on Software Inventory

```sql
Declare @EXEName as Varchar(255)
Set @EXEName = 'AL_Designer.exe'

Select distinct
Vr.Netbios_Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vr.Full_Domain_Name0 as 'Domain',
Vr.User_Name0 as 'UserName',
USR.Mail0 as 'EMail ID',
Vr.AD_Site_Name0 as 'ADSite',
SWSCAN.LastScanDate as 'LastSWScan',
DateDiff(D, SWSCAN.LastScanDate, GetDate()) as 'LastSWScanAge',
Sw.FileName as 'FileName',
Sw.FilePath as 'FilePath',
Sw.FileDescription as 'FileDescription',
Sw.FileModifiedDate as 'FileModified',
Sw.FileVersion as 'FileVersion'
FROM
v_R_System Vr
Left JOIN v_R_User Usr on Vr.User_Name0 = Usr.User_Name0
Left JOIN v_GS_SoftwareFile Sw ON Vr.ResourceID = Sw.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on Vr.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_GS_LastSoftwareScan SWSCAN on Vr.ResourceID = SWSCAN.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vr.ResourceID = Os.ResourceID
LEFT JOIN Computer_System_DATA St on Vr.ResourceID = st.MachineID
LEFT Join v_FullCollectionMembership as Col on Vr.ResourceID = Col.ResourceID
WHERE FileName like @EXEName
--and Col.CollectionID = '01A0023D'
and VR.Operating_System_Name_and0 like '%Workstation%'
GROUP BY Vr.Netbios_Name0, Os.Caption0, St.SystemType00, Vr.Full_Domain_Name0, Vr.User_Name0,
USR.Mail0, Vr.AD_Site_Name0, SWSCAN.LastScanDate,Sw.FileName, Sw.FilePath,
Sw.FileDescription, Sw.FileModifiedDate, Sw.FileVersion
ORDER BY Sw.FileName, Vr.Netbios_Name0
```

126.   Lync Exe File and Path with version - Based on Software Inventory

```sql
Declare @EXEName as Varchar(255)
Set @EXEName = 'lync.exe'

Select distinct
Vr.Netbios_Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vr.Full_Domain_Name0 as 'Domain',
Vr.User_Name0 as 'UserName',
USR.Mail0 as 'EMail ID',
Vr.AD_Site_Name0 as 'ADSite',
SWSCAN.LastScanDate as 'LastSWScan',
```

```
DateDiff(D, SWSCAN.LastScanDate, GetDate()) as 'LastSWScanAge',
Sw.FileName as 'FileName',
Sw.FilePath as 'FilePath',
Sw.FileDescription as 'FileDescription',
Sw.FileModifiedDate as 'FileModified',
Sw.FileVersion as 'FileVersion'
FROM
v_R_System Vr
Left JOIN v_R_User Usr on Vr.User_Name0 = Usr.User_Name0
Left JOIN v_GS_SoftwareFile Sw ON Vr.ResourceID = Sw.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on Vr.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_GS_LastSoftwareScan SWSCAN on Vr.ResourceID = SWSCAN.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vr.ResourceID = Os.ResourceID
LEFT JOIN Computer_System_DATA St on Vr.ResourceID = st.MachineID
LEFT Join v_FullCollectionMembership as Col on Vr.ResourceID = Col.ResourceID
WHERE FileName like @EXEName
--and Col.CollectionID = '01A0023D'
and VR.Operating_System_Name_and0 like '%Workstation%'
GROUP BY Vr.Netbios_Name0, Os.Caption0, St.SystemType00, Vr.Full_Domain_Name0, Vr.User_Name0,
USR.Mail0, Vr.AD_Site_Name0, SWSCAN.LastScanDate,Sw.FileName, Sw.FilePath,
Sw.FileDescription, Sw.FileModifiedDate, Sw.FileVersion
ORDER BY Sw.FileName, Vr.Netbios_Name0
```

127.   All PCs with McAfee Antivirus Installed Machines Report Based on Installed Software

```
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
App.ARPDisplayName0 as 'DisplayName',
App.InstallDate0  as 'InstalledDate',
App.ProductVersion0 as 'Version'
from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT Join v_FullCollectionMembership as Col on VRS.ResourceID = Col.ResourceID
LEFT JOIN v_R_User on VRS.User_Name0 = v_R_User.User_Name0
Where App.ARPDisplayName0 like '%McAfee%VirusScan%Enterprise%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and VRS.Client0 = 1 and VRS.Obsolete0 = 0
order by VRS.Name0, App.ProductVersion0
```

128.   All PCs without McAfee Antivirus Installed Machines Report Based on Installed Software

```
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
distinct(vs.Name0)  as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vs.AD_Site_Name0 as 'ADSite',
vs.Full_Domain_Name0 as 'Domain',
vs.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
```

```sql
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge'
from v_R_System vs
LEFT JOIN v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = vs.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vs.ResourceID = Os.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = vs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on vs.ResourceID = HWSCAN.ResourceID
LEFT JOIN Computer_System_DATA St on vs.ResourceID = st.MachineID
LEFT JOIN v_R_User on vs.User_Name0 = v_R_User.User_Name0
LEFT JOIN v_GS_INSTALLED_SOFTWARE  App ON vs.ResourceID = App.ResourceID
where Vs.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @Collection
and Vs.Client0 = 1 and Vs.Obsolete0 = 0
and vs.ResourceID Not in (
Select Vrs.ResourceID from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE  App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
Where App.ARPDisplayName0 like '%McAfee%VirusScan%Enterprise%'
)
```

129.   All PCs with McAfee DLP Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
App.ARPDisplayName0 as 'DisplayName',
App.InstallDate0  as 'InstalledDate',
App.ProductVersion0 as 'Version'
from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT Join v_FullCollectionMembership as Col on VRS.ResourceID = Col.ResourceID
LEFT JOIN v_R_User on VRS.User_Name0 = v_R_User.User_Name0
Where App.ARPDisplayName0 like '%McAfee%VirusScan%Enterprise%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and VRS.Client0 = 1 and VRS.Obsolete0 = 0
order by VRS.Name0, App.ProductVersion0
```

130.   All PCs without McAfee DLP Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
distinct(vs.Name0)  as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vs.AD_Site_Name0 as 'ADSite',
vs.Full_Domain_Name0 as 'Domain',
vs.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge'
```

```
from v_R_System vs
LEFT JOIN v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = vs.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vs.ResourceID = Os.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = vs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on vs.ResourceID = HWSCAN.ResourceID
LEFT JOIN Computer_System_DATA St on vs.ResourceID = st.MachineID
LEFT JOIN v_R_User on vs.User_Name0 = v_R_User.User_Name0
LEFT JOIN v_GS_INSTALLED_SOFTWARE  App ON vs.ResourceID = App.ResourceID
where Vs.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @Collection
and Vs.Client0 = 1 and Vs.Obsolete0 = 0
and vs.ResourceID Not in (
Select Vrs.ResourceID from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE  App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
Where App.ARPDisplayName0 like '%McAfee%VirusScan%Enterprise%'
)
```

131.   All Workstations Adobe Products Installed Machines Report

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
Vru.Mail0 as 'EMailID',
VRS.AD_Site_Name0 as 'ADSite',
ARP.Publisher0 as 'Publisher',
ARP.ARPDisplayName0 as 'DisplayName',
ARP.PackageCode0 as 'ProductID',
ARP.InstallDate0 as 'InstalledDate',
ARP.ProductVersion0 as 'Version',
Case when ARP.ARPDisplayName0 like '%Acrobat%Reader%' Then 'Acrobat Reader Installed'
when ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%' Then 'Acrobat Pro Installed'
when ARP.ARPDisplayName0 like '%After%Effect%' Then 'After Effect Installed'
when ARP.ARPDisplayName0 like '%Adobe%AIR%' Then 'Air Installed'
when ARP.ARPDisplayName0 like '%Adobe%Bridge%' Then 'Bridge Installed'
when ARP.ARPDisplayName0 like '%Design%Standard%' Then 'Creative Cloud Installed'
when ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' Then 'Dreamweaver Installed'
when ARP.ARPDisplayName0 like '%Adobe%Flash%Professional%' Then 'Flash Professional Installed'
when ARP.ARPDisplayName0 like '%Adobe%Illustrator%' Then 'Illustrator Installed'
when ARP.ARPDisplayName0 like '%Adobe%InDesign%' Then 'InDesign Installed'
when ARP.ARPDisplayName0 like '%Adobe%Lightroom%' Then 'Lightroom Installed'
when ARP.ARPDisplayName0 like '%Adobe%Photoshop%' Then 'Photoshop Installed'
when ARP.ARPDisplayName0 like '%Adobe%Premiere%' Then 'Premiere Installed'
Else 'Others' End 'AdobeProductName'
from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
LEFT JOIN v_R_User Vru on VRS.User_Name0 = Vru.User_Name0
where ARP.Publisher0 like '%Adobe%'
and
(ARP.ARPDisplayName0 like '%Acrobat%Reader%' or ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%'
or ARP.ARPDisplayName0 like '%After%Effect%' or ARP.ARPDisplayName0 like '%Adobe%AIR%'
or ARP.ARPDisplayName0 like '%Adobe%Bridge%' or ARP.ARPDisplayName0 like '%Design%Standard%'
or ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' or ARP.ARPDisplayName0 like '%Adobe%Flash%Professional%'
or ARP.ARPDisplayName0 like '%Adobe%Illustrator%' or ARP.ARPDisplayName0 like '%Adobe%InDesign%'
or ARP.ARPDisplayName0 like '%Adobe%Lightroom%' or ARP.ARPDisplayName0 like '%Adobe%Photoshop%'
```

```
or ARP.ARPDisplayName0 like '%Adobe%Premiere%'
)
and VRS.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @CollectionID
and VRS.Obsolete0 = 0
order by 13
```

132.   All Servers Adobe Products Installed Machines Report

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
Vru.Mail0 as 'EMailID',
VRS.AD_Site_Name0 as 'ADSite',
ARP.Publisher0 as 'Publisher',
ARP.ARPDisplayName0 as 'DisplayName',
ARP.PackageCode0 as 'ProductID',
ARP.InstallDate0 as 'InstalledDate',
ARP.ProductVersion0 as 'Version',
Case when ARP.ARPDisplayName0 like '%Acrobat%Reader%' Then 'Acrobat Reader Installed'
when ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%' Then 'Acrobat Pro Installed'
when ARP.ARPDisplayName0 like '%After%Effect%' Then 'After Effect Installed'
when ARP.ARPDisplayName0 like '%Adobe%AIR%' Then 'Air Installed'
when ARP.ARPDisplayName0 like '%Adobe%Bridge%' Then 'Bridge Installed'
when ARP.ARPDisplayName0 like '%Design%Standard%' Then 'Creative Cloud Installed'
when ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' Then 'Dreamweaver Installed'
when ARP.ARPDisplayName0 like '%Adobe%Flash%Professional%' Then 'Flash Professional Installed'
when ARP.ARPDisplayName0 like '%Adobe%Illustrator%' Then 'Illustrator Installed'
when ARP.ARPDisplayName0 like '%Adobe%InDesign%' Then 'InDesign Installed'
when ARP.ARPDisplayName0 like '%Adobe%Lightroom%' Then 'Lightroom Installed'
when ARP.ARPDisplayName0 like '%Adobe%Photoshop%' Then 'Photoshop Installed'
when ARP.ARPDisplayName0 like '%Adobe%Premiere%' Then 'Premiere Installed'
Else 'Others' End 'AdobeProductName'
from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
LEFT JOIN v_R_User Vru on VRS.User_Name0 = Vru.User_Name0
where ARP.Publisher0 like '%Adobe%'
and
(ARP.ARPDisplayName0 like '%Acrobat%Reader%' or ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%'
or ARP.ARPDisplayName0 like '%After%Effect%' or ARP.ARPDisplayName0 like '%Adobe%AIR%'
or ARP.ARPDisplayName0 like '%Adobe%Bridge%' or ARP.ARPDisplayName0 like '%Design%Standard%'
or ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' or ARP.ARPDisplayName0 like '%Adobe%Flash%Professional%'
or ARP.ARPDisplayName0 like '%Adobe%Illustrator%' or ARP.ARPDisplayName0 like '%Adobe%InDesign%'
or ARP.ARPDisplayName0 like '%Adobe%Lightroom%' or ARP.ARPDisplayName0 like '%Adobe%Photoshop%'
or ARP.ARPDisplayName0 like '%Adobe%Premiere%'
)
and VRS.Operating_System_Name_and0 like '%Server%'
and v_FullCollectionMembership.CollectionID = @CollectionID
and VRS.Obsolete0 = 0
order by 13
```

133.   All Workstations Adobe Products Installed Machines with Disk Space Report

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID

Select
```

```sql
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
Vru.Mail0 as 'EMailID',
VRS.AD_Site_Name0 as 'ADSite',
LD.DeviceID00 as 'Drive',
LD.FileSystem00 as 'FileSystem',
LD.Size00 / 1024 as 'TotalSpace (GB)',
LD.FreeSpace00 / 1024 as 'FreeSpace (GB)',
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge',
ARP.Publisher0 as 'Publisher',
ARP.ARPDisplayName0 as 'DisplayName',
ARP.PackageCode0 as 'ProductID',
ARP.InstallDate0 as 'InstalledDate',
ARP.ProductVersion0 as 'Version',
Case when ARP.ARPDisplayName0 like '%Acrobat%Reader%' Then 'Acrobat Reader Installed'
when ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%' Then 'Acrobat Pro Installed'
when ARP.ARPDisplayName0 like '%After%Effect%' Then 'After Effect Installed'
when ARP.ARPDisplayName0 like '%Adobe%AIR%' Then 'Air Installed'
when ARP.ARPDisplayName0 like '%Adobe%Bridge%' Then 'Bridge Installed'
when ARP.ARPDisplayName0 like '%Design%Standard%' Then 'Creative Cloud Installed'
when ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' Then 'Dreamweaver Installed'
when ARP.ARPDisplayName0 like '%Adobe%Flash%Professional%' Then 'Flash Professional Installed'
when ARP.ARPDisplayName0 like '%Adobe%Illustrator%' Then 'Illustrator Installed'
when ARP.ARPDisplayName0 like '%Adobe%InDesign%' Then 'InDesign Installed'
when ARP.ARPDisplayName0 like '%Adobe%Lightroom%' Then 'Lightroom Installed'
when ARP.ARPDisplayName0 like '%Adobe%Photoshop%' Then 'Photoshop Installed'
when ARP.ARPDisplayName0 like '%Adobe%Premiere%' Then 'Premiere Installed'
Else 'Others' End 'AdobeProductName'
from V_R_System VRS
LEFT JOIN v_GS_INSTALLED_SOFTWARE ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN Logical_Disk_Data Ld on Ld.MachineID = Vrs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
LEFT JOIN v_R_User Vru on VRS.User_Name0 = Vru.User_Name0
where ARP.Publisher0 like '%Adobe%'
and
(ARP.ARPDisplayName0 like '%Acrobat%Reader%' or ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%'
or ARP.ARPDisplayName0 like '%After%Effect%' or ARP.ARPDisplayName0 like '%Adobe%AIR%'
or ARP.ARPDisplayName0 like '%Adobe%Bridge%' or ARP.ARPDisplayName0 like '%Design%Standard%'
or ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' or ARP.ARPDisplayName0 like
'%Adobe%Flash%Professional%'
or ARP.ARPDisplayName0 like '%Adobe%Illustrator%' or ARP.ARPDisplayName0 like '%Adobe%InDesign%'
or ARP.ARPDisplayName0 like '%Adobe%Lightroom%' or ARP.ARPDisplayName0 like '%Adobe%Photoshop%'
or ARP.ARPDisplayName0 like '%Adobe%Premiere%'
)
and VRS.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @CollectionID
and LD.Description00 = 'Local Fixed Disk'
and VRS.Obsolete0 = 0
order by 13
```

134.   All Servers Adobe Products Installed Machines with Disk Space Report

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID
```

```sql
Select
Distinct VRS.Name0 as 'Name',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
Vru.Mail0 as 'EMailID',
VRS.AD_Site_Name0 as 'ADSite',
LD.DeviceID00 as 'Drive',
LD.FileSystem00 as 'FileSystem',
LD.Size00 / 1024 as 'TotalSpace (GB)',
LD.FreeSpace00 / 1024 as 'FreeSpace (GB)',
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge',
ARP.Publisher0 as 'Publisher',
ARP.ARPDisplayName0 as 'DisplayName',
ARP.PackageCode0 as 'ProductID',
ARP.InstallDate0 as 'InstalledDate',
ARP.ProductVersion0 as 'Version',
Case when ARP.ARPDisplayName0 like '%Acrobat%Reader%' Then 'Acrobat Reader Installed'
when ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%' Then 'Acrobat Pro Installed'
when ARP.ARPDisplayName0 like '%After%Effect%' Then 'After Effect Installed'
when ARP.ARPDisplayName0 like '%Adobe%AIR%' Then 'Air Installed'
when ARP.ARPDisplayName0 like '%Adobe%Bridge%' Then 'Bridge Installed'
when ARP.ARPDisplayName0 like '%Design%Standard%' Then 'Creative Cloud Installed'
when ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' Then 'Dreamweaver Installed'
when ARP.ARPDisplayName0 like '%Adobe%Flash%Professional%' Then 'Flash Professional Installed'
when ARP.ARPDisplayName0 like '%Adobe%Illustrator%' Then 'Illustrator Installed'
when ARP.ARPDisplayName0 like '%Adobe%InDesign%' Then 'InDesign Installed'
when ARP.ARPDisplayName0 like '%Adobe%Lightroom%' Then 'Lightroom Installed'
when ARP.ARPDisplayName0 like '%Adobe%Photoshop%' Then 'Photoshop Installed'
when ARP.ARPDisplayName0 like '%Adobe%Premiere%' Then 'Premiere Installed'
Else 'Others' End 'AdobeProductName'
from V_R_System VRS
LEFT JOIN v_GS_INSTALLED_SOFTWARE ARP ON VRS.ResourceID = ARP.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN Logical_Disk_Data Ld on Ld.MachineID = Vrs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
LEFT JOIN v_R_User Vru on VRS.User_Name0 = Vru.User_Name0
where ARP.Publisher0 like '%Adobe%'
and
(ARP.ARPDisplayName0 like '%Acrobat%Reader%' or ARP.ARPDisplayName0 like '%Adobe%Acrobat%Pro%'
or ARP.ARPDisplayName0 like '%After%Effect%' or ARP.ARPDisplayName0 like '%Adobe%AIR%'
or ARP.ARPDisplayName0 like '%Adobe%Bridge%' or ARP.ARPDisplayName0 like '%Design%Standard%'
or ARP.ARPDisplayName0 like '%Adobe%Dreamweaver%' or ARP.ARPDisplayName0 like
'%Adobe%Flash%Professional%'
or ARP.ARPDisplayName0 like '%Adobe%Illustrator%' or ARP.ARPDisplayName0 like '%Adobe%InDesign%'
or ARP.ARPDisplayName0 like '%Adobe%Lightroom%' or ARP.ARPDisplayName0 like '%Adobe%Photoshop%'
or ARP.ARPDisplayName0 like '%Adobe%Premiere%'
)
and VRS.Operating_System_Name_and0 like '%Server%'
and v_FullCollectionMembership.CollectionID = @CollectionID
and LD.Description00 = 'Local Fixed Disk'
and VRS.Obsolete0 = 0
order by 13
```

135. Deployments status for specific applications

```sql
select
aa.ApplicationName as 'Application Name',
aa.CollectionName as 'Target Collection',
ae.descript as 'DeploymentTypeName',
s1.netbios_name0 as 'ComputerName',
s1.AD_Site_Name0 as 'ADSiteName',
case when ae.AppEnforcementState = 1000 then 'Success'
when ae.AppEnforcementState = 1001 then 'Already Compliant'
when ae.AppEnforcementState = 1002 then 'Simulate Success'
when ae.AppEnforcementState = 2000 then 'In Progress'
when ae.AppEnforcementState = 2001 then 'Waiting for Content'
when ae.AppEnforcementState = 2002 then 'Installing'
when ae.AppEnforcementState = 2003 then 'Restart to Continue'
when ae.AppEnforcementState = 2004 then 'Waiting for maintenance window'
when ae.AppEnforcementState = 2005 then 'Waiting for schedule'
when ae.AppEnforcementState = 2006 then 'Downloading dependent content'
when ae.AppEnforcementState = 2007 then 'Installing dependent content'
when ae.AppEnforcementState = 2008 then 'Restart to complete'
when ae.AppEnforcementState = 2009 then 'Content downloaded'
when ae.AppEnforcementState = 2010 then 'Waiting for update'
when ae.AppEnforcementState = 2011 then 'Waiting for user session reconnect'
when ae.AppEnforcementState = 2012 then 'Waiting for user logoff'
when ae.AppEnforcementState = 2013 then 'Waiting for user logon'
when ae.AppEnforcementState = 2014 then 'Waiting to install'
when ae.AppEnforcementState = 2015 then 'Waiting retry'
when ae.AppEnforcementState = 2016 then 'Waiting for presentation mode'
when ae.AppEnforcementState = 2017 then 'Waiting for Orchestration'
when ae.AppEnforcementState = 2018 then 'Waiting for network'
when ae.AppEnforcementState = 2019 then 'Pending App-V Virtual Environment'
when ae.AppEnforcementState = 2020 then 'Updating App-V Virtual Environment'
when ae.AppEnforcementState = 3000 then 'Requirements not met'
when ae.AppEnforcementState = 3001 then 'Host platform not applicable'
when ae.AppEnforcementState = 4000 then 'Unknown'
when ae.AppEnforcementState = 5000 then 'Deployment failed'
when ae.AppEnforcementState = 5001 then 'Evaluation failed'
when ae.AppEnforcementState = 5002 then 'Deployment failed'
when ae.AppEnforcementState = 5003 then 'Failed to locate content'
when ae.AppEnforcementState = 5004 then 'Dependency installation failed'
when ae.AppEnforcementState = 5005 then 'Failed to download dependent content'
when ae.AppEnforcementState = 5006 then 'Conflicts with another application deployment'
when ae.AppEnforcementState = 5007 then 'Waiting retry'
when ae.AppEnforcementState = 5008 then 'Failed to uninstall superseded deployment type'
when ae.AppEnforcementState = 5009 then 'Failed to download superseded deployment type'
when ae.AppEnforcementState = 5010 then 'Failed to updating App-V Virtual Environment'
End as 'State Message',
case when ae.AppEnforcementState like '10%' then 'Success'
when ae.AppEnforcementState like '20%' then 'Progress'
when ae.AppEnforcementState like '30%' then 'ReqNotMet'
when ae.AppEnforcementState like '40%' then 'Unknown'
when ae.AppEnforcementState like '50%' then 'Failed'
End as 'Status',
LastComplianceMessageTime as 'LastMessageTime'
from v_R_System_Valid s1
join vAppDTDeploymentResultsPerClient ae on ae.ResourceID = s1.ResourceID
join v_CICurrentComplianceStatus ci2 on ci2.CI_ID = ae.CI_ID AND ci2.ResourceID=s1.ResourceID
join v_ApplicationAssignment aa on ae.AssignmentID = aa.AssignmentID
where ae.AppEnforcementState is not null
and aa.ApplicationName in ('ApplicationName1', 'ApplicationName2', 'ApplicationName3')
order by ae.AppEnforcementState, LastComplianceMessageTime Desc
```

136. Deployments status for specific packages

```sql
select distinct
pack.Name as 'Package Name',
COLL.CollectionName as 'Target Collection',
adv.ProgramName as 'DeploymentTypeName',
Vrs.Name0 as 'ComputerName',
Vrs.AD_Site_Name0 as 'ADSiteName',
ADV.AdvertisementID as 'AdvertisementID',
ADVS.LastStateName as 'Status',
ADVS.LastStatusTime as 'Status Time'
from v_R_System Vrs
INNER JOIN vSMS_ClientAdvertisementStatus ADVS ON Vrs.ResourceID = ADVS.ResourceID
INNER JOIN v_Advertisement ADV ON ADV.AdvertisementID = ADVS.AdvertisementID
INNER JOIN v_FullCollectionMembership CM ON Vrs.ResourceID = CM.ResourceID
LEFT JOIN v_Package Pack on adv.PackageID = pack.PackageID
LEFT JOIN v_Collections COLL ON ADV.CollectionID = COLL.SiteID
where pack.Name in ('PackageName1', 'PackageName2', 'PackageName3')
Order by Vrs.Name0
```

137. Deployment status for specific software Update deployments

```sql
select
vrs.Name0,
vrs.Active0,
vrs.AD_Site_Name0,
vrs.User_Name0,
vrs.Operating_System_Name_and0,
a.Assignment_UniqueID as DeploymentID,
a.AssignmentName as DeploymentName,
a.StartTime as Available,
a.EnforcementDeadline as Deadline,
sn.StateName as LastEnforcementState,
wsus.LastErrorCode as 'LasErrorCode',
wsus.LastScanTime as 'LastWSUSScan',
DateDiff(D, wsus.LastScanTime, GetDate()) as 'LastWSUSScan Age',
wks.LastHWScan,
DateDiff(D, wks.LastHwScan, GetDate()) as 'LastHWScan Age'
from v_CIAssignment a
join v_AssignmentState_Combined assc on a.AssignmentID=assc.AssignmentID
join v_StateNames sn on assc.StateType = sn.TopicType and sn.StateID=isnull(assc.StateID,0)
join v_R_System vrs on vrs.ResourceID = assc.ResourceID
join v_GS_WORKSTATION_STATUS wks on wks.ResourceID = assc.ResourceID
join v_UpdateScanStatus wsus on wsus.ResourceID = assc.ResourceID
where a.AssignmentName in ('UpdateGroupDeploymentName1', ' UpdateGroupDeploymentName 2',
' UpdateGroupDeploymentName 3')
and assc.StateType in (300,301)
order by 11 desc
```

138. Find SQL Server Installed Version

```sql
SELECT @@VERSION as 'SQL Server Installed Version'
```

139. Find SCCM SQL Database Size with Database File Path

```sql
select
Sys.FILEID as 'FileID',
left(Sys.NAME,15) as 'DBName',
left(Sys.FILENAME,60) as 'DBFilePath',
convert(decimal(12,2),round(Sys.size/128.000,2)) as 'Filesize (MB)',
convert(decimal(12,2),round(fileproperty(Sys.name,'SpaceUsed')/128.000,2)) as 'UsedSpace (MB)',
convert(decimal(12,2),round((Sys.size-fileproperty(Sys.name,'SpaceUsed'))/128.000,2)) as 'FreeSpace
(MB)',
convert(decimal(12,2),round(Sys.growth/128.000,2)) as 'GrowthSpace (MB)'
from dbo.sysfiles Sys
```

140. Find Overall SCCM Site Hierarchy Information

```sql
select distinct
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Site System') as 'SCCM SVR Counts',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Site Server') as 'SCCM Site Counts',
(
select COUNT(*) from v_SystemResourceList Vrl
Inner Join V_site Vs on Vs.ServerName = Vrl.ServerName
Where Vrl.RoleName = 'SMS Site Server' and Vs.Type = 3 and Vs.ReportingSiteCode is not null
) as 'CAS Site Counts',
(
select COUNT(*) from v_SystemResourceList Vrl
Inner Join V_site Vs on Vs.ServerName = Vrl.ServerName
Where Vrl.RoleName = 'SMS Site Server' and Vs.Type = 2 and Vs.ReportingSiteCode is not null
) as 'PRI Site Counts',
(
select COUNT(*) from v_SystemResourceList Vrl
Inner Join V_site Vs on Vs.ServerName = Vrl.ServerName
Where Vrl.RoleName = 'SMS Site Server' and Vs.Type = 1 and Vs.ReportingSiteCode is not null
) as 'SEC Site Counts',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Management Point') as 'MP SVR Counts',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Distribution Point') as 'DP SVR
Counts',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Software Update Point') as 'SUP SVR
Counts',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS SRS Reporting Point') as 'SSRS SVR
Counts',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Provider') as 'SMSPro SVR Counts'
from v_SystemResourceList
```

141. Find SCCM Site Hierarchy Detailed Information

```sql
Select
V_Site.SiteCode as 'SiteCode',
V_Site.ReportingSiteCode as 'ReportTo',
V_Site.ServerName as 'ServerName',
V_Site.SiteName as 'SiteName',
CASE
WHEN V_Site.Type = 3 and V_Site.ReportingSiteCode is not null THEN 'CAS Site Server'
WHEN V_Site.Type = 2 and V_Site.ReportingSiteCode ='' THEN 'Standalone Primary Site Server'
WHEN V_Site.Type = 2 and V_Site.ReportingSiteCode is not null THEN 'Primary Site Server'
WHEN V_Site.Type = 1 and V_Site.ReportingSiteCode is not null THEN 'Secondary Site Server'
ELSE 'Others' END as 'Site Server Detail',
V_Site.InstallDir as 'Installed Directory',
CASE
WHEN V_Site.BuildNumber = '7711' THEN '2012 RTM'
WHEN V_Site.BuildNumber = '7804' THEN '2012 SP1'
WHEN V_Site.BuildNumber = '7958' THEN '2012 R2'
WHEN V_Site.BuildNumber = '8239' THEN '2012 R2 SP1'
WHEN V_Site.BuildNumber = '8325' THEN '1511'
WHEN V_Site.BuildNumber = '8355' THEN '1602'
WHEN V_Site.BuildNumber = '8412' THEN '1606'
WHEN V_Site.BuildNumber = '8458' THEN '1610'
ELSE 'Others' END as 'SCCM Version',
V_Site.Version as 'Build Version'
from V_Site
```

142. Find Overall Windows Workstations Client Machines OS category with counts

```sql
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID
Select
Os.Caption0 as 'Operating System',
```

```
COUNT(*) AS 'Total'
from v_FullCollectionMembership Vf
inner join v_GS_OPERATING_SYSTEM Os on Os.ResourceID = Vf.ResourceID
where Vf.CollectionID = @CollectionID and Vf.ResourceID in
(
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Workstation%'
)
GROUP BY Os.Caption0
ORDER BY Os.Caption0 Desc
```

143. Find Overall Windows Servers Client Machines OS category with counts

```
Declare @CollectionID as Varchar(8)
Set @CollectionID = 'SMS00001' -- specify scope collection ID
Select
Os.Caption0 as 'Operating System',
COUNT(*) AS 'Total'
from v_FullCollectionMembership Vf
inner join v_GS_OPERATING_SYSTEM Os on Os.ResourceID = Vf.ResourceID
where Vf.CollectionID = @CollectionID and Vf.ResourceID in
(
select ResourceID from v_R_System where Operating_System_Name_and0 like '%Server%'
)
GROUP BY Os.Caption0
ORDER BY Os.Caption0 Desc
```

144. Find Overall WSUS Server Configurations and Ports

```
select * from wsusserverlocations
```

145. SEP Antivirus with Specific Version Installed Machines Report Based on Installed Software

```
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
App.ARPDisplayName0 as 'DisplayName',
App.InstallDate0  as 'InstalledDate',
App.ProductVersion0 as 'Version'
from V_R_System VRS
LEFT JOIN v_GS_INSTALLED_SOFTWARE App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT Join v_FullCollectionMembership as Col on VRS.ResourceID = Col.ResourceID
LEFT JOIN v_R_User on VRS.User_Name0 = v_R_User.User_Name0
Where App.ARPDisplayName0 like 'Symantec%Endpoint%Protection%'
and App.ProductVersion0 like '%14.0%'
and VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and VRS.Client0 = 1 and VRS.Obsolete0 = 0
order by VRS.Name0, App.ProductVersion0
```

146. SEP Antivirus without Specific Version Installed Machines Report Based on Installed Software

```sql
Declare @Collection varchar(8)
Declare @ApplicationName varchar(255)
Declare @ApplicationVersion varchar(255)

Set @Collection = 'SMS00001' -- specify scope collection ID
Set @ApplicationName = 'Symantec%Endpoint%Protection%' -- specify Application Name
Set @ApplicationVersion = '%14.0%' -- specify Application Version

Select
distinct(vs.Name0)  as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
Vs.AD_Site_Name0 as 'ADSite',
vs.Full_Domain_Name0 as 'Domain',
vs.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
HWSCAN.LastHWScan as 'LastHWScan',
DateDiff(D, HWSCAN.LastHwScan, GetDate()) as 'LastHWScanAge'
from v_R_System vs
LEFT JOIN v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = vs.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on Vs.ResourceID = Os.ResourceID
LEFT JOIN v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = vs.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on vs.ResourceID = HWSCAN.ResourceID
LEFT JOIN Computer_System_DATA St on vs.ResourceID = st.MachineID
LEFT JOIN v_R_User on vs.User_Name0 = v_R_User.User_Name0
LEFT JOIN v_GS_INSTALLED_SOFTWARE  App ON vs.ResourceID = App.ResourceID
where Vs.Operating_System_Name_and0 like '%Workstation%'
and v_FullCollectionMembership.CollectionID = @Collection
and Vs.Client0 = 1 and Vs.Obsolete0 = 0
and vs.ResourceID Not in (
Select Vrs.ResourceID from V_R_System VRS
Left JOIN v_GS_INSTALLED_SOFTWARE  App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
Where App.ARPDisplayName0 like @ApplicationName
and App.ProductVersion0 like @ApplicationVersion
)
```

147. All Application Installed on Specific Collection

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
App.ARPDisplayName0 as 'DisplayName',
App.InstallDate0  as 'InstalledDate',
App.ProductVersion0 as 'Version'
from V_R_System VRS
LEFT JOIN v_GS_INSTALLED_SOFTWARE App ON VRS.ResourceID = App.ResourceID
LEFT JOIN Computer_System_DATA St on VRS.ResourceID = st.MachineID
LEFT JOIN v_GS_OPERATING_SYSTEM Os on VRS.ResourceID = Os.ResourceID
LEFT JOIN v_GS_WORKSTATION_STATUS HWSCAN on VRS.ResourceID = HWSCAN.ResourceID
LEFT Join v_FullCollectionMembership as Col on VRS.ResourceID = Col.ResourceID
```

```sql
LEFT JOIN v_R_User on VRS.User_Name0 = v_R_User.User_Name0
Where Col.CollectionID = @Collection
and VRS.Client0 = 1 and VRS.Obsolete0 = 0
order by VRS.Name0, App.ProductVersion0
```

148.   All Client Health Last Month SLA and KPI Data status

```sql
Declare @CollectionID as Varchar(8)
Declare @TotalMachines as Numeric(8)
Declare @Healthy as Numeric(8)
Declare @UnHealthy as Numeric(8)
Declare @HWInventorySuccess as Numeric(8)
Declare @HWInventoryNotRun as Numeric(8)
Declare @WSUSScanSuccess as Numeric(8)
Declare @WSUSScanNotRun as Numeric(8)

Set @CollectionID = 'SMS00001'

select @TotalMachines = (select COUNT(*) from v_FullCollectionMembership where CollectionID =
@CollectionID)

select @Healthy = ( select COUNT(*) from v_FullCollectionMembership where CollectionID = @CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete != 1 and IsClient = 1
)

select @UnHealthy = ( select COUNT(*) from v_FullCollectionMembership where CollectionID =
@CollectionID
and ResourceID Not in (select ResourceID from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsActive = 1 and IsObsolete != 1 and IsClient = 1 )
)

select @HWInventorySuccess = ( select COUNT(*) from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where (DATEDIFF
(day,LastHWScan,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where
AgentName in ('Heartbeat Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)
select @HWInventoryNotRun = ( select COUNT(*) from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_GS_WORKSTATION_STATUS where (DATEDIFF
(day,LastHWScan,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where
AgentName in ('Heartbeat Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)
)

select @WSUSScanSuccess = ( select COUNT(*) from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and (DATEDIFF
(day,LastScanTime,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where
AgentName in ('Heartbeat Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)

select @WSUSScanNotRun = ( select COUNT(*) from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
```

```sql
and ResourceID Not In (select ResourceID from v_FullCollectionMembership where CollectionID =
@CollectionID
and IsAssigned = 1 and IsClient = 1 and IsActive = 1 and IsObsolete != 1
and ResourceID in (select ResourceID from v_UpdateScanStatus where lastErrorCode = 0 and (DATEDIFF
(day,LastScanTime,GetDate())<30) and ResourceID in (select ResourceID from v_AgentDiscoveries Where
AgentName in ('Heartbeat Discovery') and DATEDIFF (day,AgentTime,GetDate())<30))
)
)

select
@TotalMachines as 'TotalMachines',
@Healthy as 'Healthy',
@UnHealthy as 'UnHealthy',
(Select(@Healthy/@TotalMachines)*100) as 'Healthy%',
@HWInventorySuccess as 'HWInventorySuccess',
@HWInventoryNotRun as 'HWInventoryNotRun',
(Select(@HWInventorySuccess/@Healthy)*100) as 'HWInventorySuccess%',
@WSUSScanSuccess as 'WSUSScanSuccess',
@WSUSScanNotRun as 'WSUSScanNotRun',
(Select(@WSUSScanSuccess/@Healthy)*100) as 'WSUSScanSuccess%'
```

149.   All Software Applications Last Month deployments SLA and KPI Data status

```sql
Declare @AppDeploymentsReportNeededMonths as integer
Set @AppDeploymentsReportNeededMonths = 1      --Specify the no of Months

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
 'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' when Ds
.DeploymentIntent = 3 Then 'Simulate' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1
and (DATEDIFF(m,Ds.CreationTime, GETDATE()) = @AppDeploymentsReportNeededMonths)
and (DATEDIFF(m,Ds.DeploymentTime, GETDATE()) = @AppDeploymentsReportNeededMonths)
order by Ds.DeploymentTime desc
```

150. All Software Packages Last Month deployments SLA and KPI Data status

```sql
Declare @PKGDeploymentsReportNeededMonths as integer
Set @PKGDeploymentsReportNeededMonths = 1    --Specify the no of Months

Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1)as 'ApplicationName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2
and (DATEDIFF(m,Ds.ModificationTime, GETDATE()) = @PKGDeploymentsReportNeededMonths)
and ((Ds.ProgramName not like 'Shop-Install%') and (Ds.ProgramName not like 'Shop-Uninst%'))
order by Ds.DeploymentTime desc
```

151. All Software Updates Last Month deployments SLA and KPI Data status

```sql
Declare @PatchDeploymentsReportNeededMonths as integer
Set @PatchDeploymentsReportNeededMonths = 1     --Specify the no of Months

Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
'Software Update' as 'PackageName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
```

```
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5
and (DATEDIFF(m,Vaa.CreationTime, GETDATE()) = @PatchDeploymentsReportNeededMonths)
order by Ds.DeploymentTime desc
```

152.   All OS Last Month deployments SLA and KPI Data status

```
Declare @OSDeploymentsReportNeededMonths as integer
Set @OSDeploymentsReportNeededMonths = 1      --Specify the no of Months
Select
Vaa.AdvertisementName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7
and (DATEDIFF(m,Ds.ModificationTime, GETDATE()) = @OSDeploymentsReportNeededMonths)
order by Ds.DeploymentTime desc
```

153.   Selected KB Article ID patch required or installed status for Specific Collection ID

```
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
UI.ArticleID as 'ArticleID',
UI.BulletinID as 'BulletinID',
UI.Title as 'Title',
CASE WHEN UCS.Status = 2 THEN 'Required'
```

```sql
WHEN UCS.Status = 3 THEN 'Installed'
ELSE 'Unknown' END AS 'KBStatus',
UI.InfoURL as 'InformationURL'
FROM v_UpdateComplianceStatus UCS
INNER JOIN v_UpdateInfo UI ON UCS.CI_ID = UI.CI_ID
INNER JOIN v_CICategories_All CIC ON UI.CI_ID = CIC.CI_ID
INNER JOIN v_CategoryInfo CI ON CIC.CategoryInstanceID = CI.CategoryInstanceID
INNER JOIN v_R_System VRS ON UCS.ResourceID = VRS.ResourceID
INNER JOIN v_GS_OPERATING_SYSTEM Os on UCS.ResourceID = Os.ResourceID
INNER JOIN Computer_System_DATA St on UCS.ResourceID = st.MachineID
INNER Join v_FullCollectionMembership Col on UCS.ResourceID = Col.ResourceID
WHERE VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and UI.articleid in ('4012212','4012213','4012214','4012598')
--and UI.BulletinID in ('ms17-010','ms17-008')
and active0 = 1 and client0 = 1
Order by 10
```

154.   Selected KB Article ID patch required or installed status for Specific Collection ID

```sql
Declare @Collection varchar(8)
Set @Collection = 'SMS00001' -- specify scope collection ID

Select
Distinct VRS.Name0 as 'MachineName',
Os.Caption0 as 'OperatingSystem',
St.SystemType00 as 'OSType',
VRS.AD_Site_Name0 as 'ADSite',
VRS.Full_Domain_Name0 as 'Domain',
VRS.User_Name0 as 'UserName',
UI.ArticleID as 'ArticleID',
UI.BulletinID as 'BulletinID',
UI.Title as 'Title',
CASE WHEN UCS.Status = 2 THEN 'Required'
WHEN UCS.Status = 3 THEN 'Installed'
ELSE 'Unknown' END AS 'KBStatus',
UI.InfoURL as 'InformationURL'
FROM v_UpdateComplianceStatus UCS
INNER JOIN v_UpdateInfo UI ON UCS.CI_ID = UI.CI_ID
INNER JOIN v_CICategories_All CIC ON UI.CI_ID = CIC.CI_ID
INNER JOIN v_CategoryInfo CI ON CIC.CategoryInstanceID = CI.CategoryInstanceID
INNER JOIN v_R_System VRS ON UCS.ResourceID = VRS.ResourceID
INNER JOIN v_GS_OPERATING_SYSTEM Os on UCS.ResourceID = Os.ResourceID
INNER JOIN Computer_System_DATA St on UCS.ResourceID = st.MachineID
INNER Join v_FullCollectionMembership Col on UCS.ResourceID = Col.ResourceID
WHERE VRS.Operating_System_Name_and0 like '%Workstation%'
and Col.CollectionID = @Collection
and UI.articleid in ('4012212','4012213','4012214','4012598')
--and UI.BulletinID in ('ms17-010','ms17-008')
and active0 = 1 and client0 = 1
Order by 10
```

155.   Site Status Overview Status

```sql
Select
SiteStatus.SiteCode, SiteInfo.SiteName, SiteStatus.Updated 'Time Stamp',
Case SiteStatus.Status
When 0 Then 'OK'
When 1 Then 'Warning'
When 2 Then 'Critical'
```

```
Else ' '
End AS 'Site Status',
Case SiteInfo.Status
When 1 Then 'Active'
When 2 Then 'Pending'
When 3 Then 'Failed'
When 4 Then 'Deleted'
When 5 Then 'Upgrade'
Else ' '
END AS 'Site State'
From V_SummarizerSiteStatus SiteStatus Join v_Site SiteInfo on SiteStatus.SiteCode = SiteInfo.SiteCode
where SiteInfo.Status <> 1
Order By SiteCode
```

156.   Site Status Status

```
SELECT distinct
SiteCode 'Site Code',
SUBSTRING(SiteSystem, CHARINDEX('\\', SiteSystem) + 2, CHARINDEX('"]', SiteSystem) - CHARINDEX('\\',
SiteSystem) - 3 ) AS 'Site System',
REPLACE(Role, 'SMS', 'ConfigMgr') 'Role',
Case ObjectType
When 0 Then 'Directory'
When 1 Then 'SQL Database'
When 2 Then 'SQL Transaction Log'
Else ' '
END AS 'Object Type',
CAST(BytesTotal/1024 AS VARCHAR(49)) + 'MB' 'Total',
CAST(BytesFree/1024 AS VARCHAR(49)) + 'MB' 'Free',
CASE PercentFree
When -1 Then 'Unknown'
When -2 Then 'Automatically grow'
ELSE CAST(PercentFree AS VARCHAR(49)) + '%'
END AS '%Free',
Case v_SiteSystemSummarizer.Status
When 0 Then 'OK'
When 1 Then 'Warning'
When 2 Then 'Critical'
Else ' '
End As 'Status'
FROM v_SiteSystemSummarizer
--where v_SiteSystemSummarizer.Status <> 0
Order By 1
```

157.   Site Components Status
```
SELECT distinct  SiteCode ,
MachineName 'ServerName',
ComponentName ,
Case v_componentSummarizer.State When 0 Then 'Stopped' When 1 Then 'Started' When 2 Then 'Paused' When
3 Then 'Installing' When 4 Then 'Re-Installing' When 5 Then 'De-Installing' Else ' ' END AS 'Thread
State',
Errors,
Warnings,
Infos,
Case v_componentSummarizer.Type When 0 Then 'Autostarting' When 1 Then 'Scheduled' When 2 Then 'Manual'
ELSE ' ' END AS 'StartupType',
CASE AvailabilityState When 0 Then 'Online' When 3 Then 'Offline' ELSE ' ' END AS 'State',
Case v_ComponentSummarizer.Status When 0 Then 'OK' When 1 Then 'Warning' When 2 Then 'Critical' Else '
' End As 'Status'
```

```
from v_ComponentSummarizer Where TallyInterval = '0001128000100008'
and (v_ComponentSummarizer.Status = 2 or v_ComponentSummarizer.Status = 1)
Order By ComponentName,SiteCode
```

158.   Overall Content Distribution Status

```
SELECT DPSI.Name,
(DPSI.NumberInstalled+DPSI.NumberInProgress+DPSI.NumberErrors+DPSI.NumberUnknown) as 'NumberTotal',
DPSI.NumberInstalled, DPSI.NumberInProgress, DPSI.NumberErrors, DPSI.NumberUnknown FROM
vSMS_DPStatusInfo DPSI
ORDER BY 2 desc
```

159.   Compare Two DP's Packages Status

```
Select
Pkg.PackageID,Pkg.Name
from v_Package Pkg
where Pkg.PackageID in
(
select PackageID from v_DistributionPoint where ServerNALPath like  '%DPName1%'
and PackageID not in (select PackageID from v_DistributionPoint where ServerNALPath like  '% DPName2%')
)
```

160.   Top Users for Specific Computer Status

```
SELECT DISTINCT
Vrs.Name0 AS [Computer Name],
Os.Caption0 AS [Operating System],
Enc.SerialNumber0 AS [Service Tag],
Vgs.Manufacturer0 AS [Make],
Vrs.User_Name0 AS [Last Logged on User],
Con.TopConsoleUser0 AS [Top User]
FROM v_r_system Vrs
LEFT JOIN v_gs_computer_system Vgs ON Vrs.ResourceID = Vgs.ResourceID
LEFT JOIN v_gs_system_enclosure Enc ON Vrs.ResourceID = Enc.ResourceID
LEFT JOIN v_GS_SYSTEM_CONSOLE_USAGE Con ON Vrs.ResourceID = Con.ResourceID
LEFT JOIN v_GS_OPERATING_SYSTEM Os ON Vrs.ResourceID = Os.ResourceID
ORDER BY Vrs.Name0
```

161.   Software Update Group Created, Modified or Deleted Properties

```
Select
rsm.Severity,
rsm.MessageTypeString as 'Type',
rsm.SiteCode,
rsm.Timestamp as 'Date/Time',
rsm.System,
rsm.Component,
rsm.MessageID,
'User "' + rsm.InsStrValue1 + '"' +
CASE
when rsm.MessageID = 30196 Then ' created updates assignment '
when rsm.MessageID = 30197 Then ' modified updates assignment '
when rsm.MessageID = 30198 Then ' deleted updates assignment '
when rsm.MessageID = 30219 Then ' created authorization list '
when rsm.MessageID = 30220 Then ' modified authorization list '
when rsm.MessageID = 30221 Then ' deleted authorization list '
End
+ rsm.InsStrValue2 + ' ' + rsm.InsStrValue3 + ' ' + rsm.InsStrValue4 as 'Description',
cia.CollectionID,
cia.CollectionName
from v_Report_StatusMessageDetail rsm
```

```sql
left join v_CIAssignment cia on rsm.InsStrValue2 = cia.AssignmentID
where rsm.MessageID >= 30196 and rsm.MessageID <= 30198
or rsm.MessageID >= 30218  and rsm.MessageID <= 30221
order by 4 desc
```

162.   Software Update last month patch compliance report using Compliance Settings

```sql
DECLARE @StartDate DATETIME, @EndDate DATETIME
SET @StartDate = DATEADD(mm, DATEDIFF(m,0,GETDATE())-2,0)
SET @EndDate = DATEADD(s,-1,DATEADD(mm, DATEDIFF(m,0,GETDATE())-1,0))

SELECT
dbo.v_R_System.ResourceID,
dbo.v_R_System.Name0 AS [Name],
dbo.v_CICurrentComplianceStatus.ComplianceState,
v_LocalizedCIProperties_SiteLoc.DisplayName AS [BaselineName],
CAST(dbo.v_LocalizedCIProperties_SiteLoc.Description AS DATETIME) AS [BaselineDate],
SUMMARY.LastActiveTime,
OP.LastBootUpTime0 AS [LastBootUpTime],
DATEDIFF(Day, OP.LastBootUpTime0, GETDATE()) as [DaysSinceReboot]
FROM
dbo.v_BaselineTargetedComputers
INNER JOIN dbo.v_R_System ON v_R_System.ResourceID = dbo.v_BaselineTargetedComputers.ResourceID
INNER JOIN dbo.v_ConfigurationItems ON dbo.v_ConfigurationItems.CI_ID =
v_BaselineTargetedComputers.CI_ID
INNER JOIN dbo.v_CICurrentComplianceStatus ON dbo.v_CICurrentComplianceStatus.CI_ID =
dbo.v_ConfigurationItems.CI_ID AND
dbo.v_CICurrentComplianceStatus.ResourceID = dbo.v_BaselineTargetedComputers.ResourceID
INNER JOIN dbo.v_LocalizedCIProperties_SiteLoc ON dbo.v_LocalizedCIProperties_SiteLoc.CI_ID =
dbo.v_ConfigurationItems.CI_ID
INNER JOIN dbo.v_CH_ClientSummary SUMMARY ON dbo.v_R_System.ResourceID = SUMMARY.ResourceID
LEFT JOIN dbo.v_GS_OPERATING_SYSTEM OP ON dbo.v_R_System.ResourceID = OP.ResourceID
WHERE
dbo.v_LocalizedCIProperties_SiteLoc.DisplayName LIKE 'Windows Updates %'
and dbo.v_LocalizedCIProperties_SiteLoc.DisplayName LIKE 'Windows Updates April%'
AND CAST(dbo.v_LocalizedCIProperties_SiteLoc.Description AS DATETIME) BETWEEN @StartDate AND @EndDate
ORDER BY dbo.v_R_System.Name0
```

163.   Client Health showing inactive or failed clients status

```sql
SELECT ClientstateDescription AS [ClientState],
COUNT(ResourceID) AS [NumberofClients],
CONVERT(varchar, 100 * count(*) / tot,1) as 'Percent'
FROM v_CH_ClientSummary,
(SELECT COUNT(*) as tot FROM v_CH_ClientSummary) x
GROUP BY ClientstateDescription, tot ORDER BY [NumberofClients] DESC
```

164.   Check if allow clients to use fallback source location for content

```sql
WITH XMLNAMESPACES ( DEFAULT
'http://schemas.microsoft.com/SystemsCenterConfigurationManager/2009/06/14/Rules','http://schemas.micro
soft.com/SystemCenterConfigurationManager/2009/AppMgmtDigest' as p1)
SELECT
A.[App Name] [Application],max(A.[DT Name])[DT Name],A.Type,A.ExecutionContext,A.RequiresLogOn
,A.UserInteractionMode,case when A.FallbackToUnprotectedDP='true' then 'Yes' else 'No' end as
[FallbackToUnprotectedDP]
,A.OnSlowNetwork
from (
SELECT LPC.DisplayName [App Name]
,(LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Title)[1]', 'nvarchar(max)')) AS
[DT Name]
```

```
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/@Technology)[1]',
'nvarchar(max)') AS [Type]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:Contents/p1:Content/p
1:Location)[1]', 'nvarchar(max)') AS [ContentLocation]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:UninstallAction/p1:Ar
gs/p1:Arg)[1]', 'nvarchar(max)') AS [UninstallCommandLine]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:InstallAction/p1:Args
/p1:Arg)[3]', 'nvarchar(max)') AS [ExecutionContext]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:InstallAction/p1:Args
/p1:Arg)[4]', 'nvarchar(max)') AS [RequiresLogOn]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:InstallAction/p1:Args
/p1:Arg)[8]', 'nvarchar(max)') AS [UserInteractionMode]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:Contents/p1:Content/p
1:FallbackToUnprotectedDP)[1]', 'nvarchar(max)') AS [FallbackToUnprotectedDP]
,LDT.SDMPackageDigest.value('(/p1:AppMgmtDigest/p1:DeploymentType/p1:Installer/p1:Contents/p1:Content/p
1:OnSlowNetwork)[1]', 'nvarchar(max)') AS [OnSlowNetwork]
FROM
dbo.fn_ListApplicationCIs(1033) LPC
RIGHT Join fn_ListDeploymentTypeCIs(1033) LDT ON LDT.AppModelName = LPC.ModelName
where LDT.CIType_ID = 21 AND LDT.IsLatest = 1
) A
GROUP BY A.[App Name],A.Type,A.ExecutionContext,A.RequiresLogOn,A.UserInteractionMode,
A.OnSlowNetwork,A.FallbackToUnprotectedDP
```

165.   All Workstations Computers Name with Last Logon User and Serial No Detailed Report

```
Declare @CollectionID Varchar(8)
Set @CollectionID = 'SMS00001'

Select
Distinct (VRS.Netbios_Name0) as 'Name',
PC_BIOS_DATA.SerialNumber00 as 'SerialNumber',
VRS.User_Domain0+'\'+ VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID'
from V_R_System VRS
Left Outer join PC_BIOS_DATA on PC_BIOS_DATA.MachineID = VRS.ResourceId
Left Outer join Operating_System_DATA on Operating_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
left outer join v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
Left Outer join v_R_User on VRS.User_Name0 = v_R_User.User_Name0
where VRS.Operating_System_Name_and0 like '%Workstation%'
and (VRS.Obsolete0 = 0 or VRS.Obsolete0 is null)
and VRS.Client0 = 1
and v_FullCollectionMembership.CollectionID = @CollectionID
GROUP BY
VRS.Netbios_Name0,VRS.Client0,PC_BIOS_DATA.SerialNumber00,Vrs.User_Domain0,Vrs.User_Name0,v_R_User.Mail
0
order by VRS.Netbios_Name0
```

166. Check Client Versions with Percentage

```sql
Declare @CollectionID Varchar(8)
Set @CollectionID = 'SMS00001'

Select sys.Client_version0 as 'Client Version',
CASE
WHEN sys.client_version0 = '5.00.8325.1000' THEN 'ConfigMgr 1511'
WHEN sys.client_version0 = '5.00.8355.1000' THEN 'ConfigMgr 1602'
WHEN sys.client_version0 = '5.00.8355.1306' THEN 'ConfigMgr 1602 - Update Rollup 1'
WHEN sys.client_version0 = '5.00.8412.1007' THEN 'ConfigMgr 1606'
WHEN sys.client_version0 = '5.00.8355.1306' THEN 'ConfigMgr 1602 - Update Rollup 1'
WHEN sys.client_version0 = '5.00.8412.1307' THEN 'ConfigMgr 1606 - Update Rollup 1'
WHEN sys.client_version0 = '5.00.8458.1007' THEN 'ConfigMgr 1610'
WHEN sys.client_version0 = '5.00.8458.1520' THEN 'SCCM 1610 - Update rollup (KB4010155)'
WHEN sys.client_version0 = '5.00.8498.1008' THEN 'ConfigMgr 1702'
WHEN sys.client_version0 = '5.00.8498.1711' THEN 'ConfigMgr 1702 - Update rollup 1 (KB4019926)'
ELSE
sys.client_version0
END as 'ConfigMgr Release',
Count(DISTINCT sys.ResourceID) as 'Client Count',
(STR((COUNT(sys.ResourceID)*100.0/(
Select COUNT(SYS.ResourceID)
From v_FullCollectionMembership FCM INNER JOIN V_R_System sys on FCM.ResourceID = SYS.ResourceID
Where
FCM.CollectionID = @CollectionID
and
Sys.Client0= '1')),5,2)) + ' %' AS 'Percent %'
From v_FullCollectionMembership FCM INNER JOIN V_R_System sys on FCM.ResourceID = SYS.ResourceID
Where SYS.Client0 = '1' and FCM.CollectionID = @CollectionID
Group By sys.Client_version0
Order by sys.Client_version0 DESC
```

167. Microsoft Installed application counts for a specific Collection

```sql
Declare @CollectionID Varchar(8)
Set @CollectionID = 'SMS00001'

SELECT distinct DisplayName0,
Count(arp.ResourceID) AS 'Count',
Publisher0,
@CollectionID as CollectionID
FROM v_Add_Remove_Programs arp
JOIN v_FullCollectionMembership fcm on arp.ResourceID = fcm.ResourceID
WHERE fcm.CollectionID = @CollectionID AND (Publisher0 LIKE 'Microsoft%')
AND DisplayName0 NOT LIKE '%Hotfix%' AND DisplayName0 NOT LIKE '%Security Update%'
AND DisplayName0 NOT LIKE '%Update for%' AND DisplayName0 NOT LIKE '%.NET%'
AND DisplayName0 NOT LIKE '%Viewer%' AND DisplayName0 NOT LIKE '%Language Pack%'
AND DisplayName0 NOT LIKE '%Internet Explorer%' AND DisplayName0 NOT LIKE '%MSXML%'
AND DisplayName0 NOT LIKE '%SDK%' AND DisplayName0 NOT LIKE '%C++%'
AND DisplayName0 NOT LIKE '%Redistributable%' AND DisplayName0 NOT LIKE '%Search%'
AND DisplayName0 NOT LIKE '%SMS%' AND DisplayName0 NOT LIKE '%Silverlight%'
AND DisplayName0 NOT LIKE '%Live Meeting%' AND DisplayName0 NOT LIKE '%(KB%'
AND DisplayName0 NOT LIKE '%Office Web%' AND DisplayName0 NOT LIKE '%Office %Proof%'
AND DisplayName0 NOT LIKE '%Server %Proof%' AND DisplayName0 NOT LIKE '%Office %Shared%'
AND DisplayName0 NOT LIKE '%Baseline Security Analyzer%' AND DisplayName0 NOT LIKE '%Compatibility Pack%'
AND DisplayName0 NOT LIKE '%User State Migration Tools%'
GROUP BY DisplayName0, Publisher0
ORDER BY Publisher0
```

168.   Client Health Dashboard

```sql
Declare @CollectionID Varchar(8)
Set @CollectionID = 'SMS00001'
select
Vrs.Name0 as 'Computer Name',
Vrs.User_Name0 as 'User Name',
case when Vrs.client0 = 1 then 'Installed'
else 'NotInstalled' end as 'ClientAgentStatus',
summ.ClientStateDescription,
case when summ.ClientActiveStatus = 0 then 'Inactive'
when summ.ClientActiveStatus = 1 then 'Active'
end as 'ClientActiveStatus',
summ.LastActiveTime,
case when summ.IsActiveDDR = 0 then 'Inactive'
when summ.IsActiveDDR = 1 then 'Active'
end as 'IsActiveDDR',
case when summ.IsActiveHW = 0 then 'Inactive'
when summ.IsActiveHW = 1 then 'Active'
end as 'IsActiveHW',
case when summ.IsActiveSW = 0 then 'Inactive'
when summ.IsActiveSW = 1 then 'Active'
end as 'IsActiveSW',
case when summ.ISActivePolicyRequest = 0 then 'Inactive'
when summ.ISActivePolicyRequest = 1 then 'Active'
end as 'ISActivePolicyRequest',
case when summ.IsActiveStatusMessages = 0 then 'Inactive'
when summ.IsActiveStatusMessages = 1 then 'Active'
end as 'IsActiveStatusMessages',
summ.LastOnline,
summ.LastDDR,
summ.LastHW,
summ.LastSW,
summ.LastPolicyRequest,
summ.LastStatusMessage,
summ.LastHealthEvaluation,
case when LastHealthEvaluationResult = 1 then 'Not Yet Evaluated'
when LastHealthEvaluationResult = 2 then 'Not Applicable'
when LastHealthEvaluationResult = 3 then 'Evaluation Failed'
when LastHealthEvaluationResult = 4 then 'Evaluated Remediated Failed'
when LastHealthEvaluationResult = 5 then 'Not Evaluated Dependency Failed'
when LastHealthEvaluationResult = 6 then 'Evaluated Remediated Succeeded'
when LastHealthEvaluationResult = 7 then 'Evaluation Succeeded'
end as 'Last Health Evaluation Result',
case when LastEvaluationHealthy = 1 then 'Pass'
when LastEvaluationHealthy = 2 then 'Fail'
when LastEvaluationHealthy = 3 then 'Unknown'
end as 'Last Evaluation Healthy',
case when summ.ClientRemediationSuccess = 1 then 'Pass'
when summ.ClientRemediationSuccess = 2 then 'Fail'
else ''
end as 'ClientRemediationSuccess',
summ.ExpectedNextPolicyRequest
from V_R_System Vrs
Left join v_CH_ClientSummary summ on Vrs.ResourceID = summ.ResourceID
Where Vrs.ResourceID in (select ResourceID from v_FullCollectionMembership fcm where fcm.CollectionID =
@CollectionID)
order by Vrs.Name0
```

169. Client Version with Percentage

```sql
Declare @CollectionID Varchar(8)
Set @CollectionID = 'SMS00001'

Select
sys.Client_version0 as 'Client Version',
CASE
WHEN sys.client_version0 = '5.00.8325.1000' THEN 'ConfigMgr 1511'
WHEN sys.client_version0 = '5.00.8355.1000' THEN 'ConfigMgr 1602'
WHEN sys.client_version0 = '5.00.8355.1306' THEN 'ConfigMgr 1602 - Update Rollup 1'
WHEN sys.client_version0 = '5.00.8412.1007' THEN 'ConfigMgr 1606'
WHEN sys.client_version0 = '5.00.8355.1306' THEN 'ConfigMgr 1602 - Update Rollup 1'
WHEN sys.client_version0 = '5.00.8412.1307' THEN 'ConfigMgr 1606 - Update Rollup 1'
WHEN sys.client_version0 = '5.00.8458.1007' THEN 'ConfigMgr 1610'
WHEN sys.client_version0 = '5.00.8458.1520' THEN 'SCCM 1610 - Update rollup (KB4010155)'
WHEN sys.client_version0 = '5.00.8498.1008' THEN 'ConfigMgr 1702'
WHEN sys.client_version0 = '5.00.8498.1711' THEN 'ConfigMgr 1702 - Update rollup 1 (KB4019926)'
ELSE
sys.client_version0
END as 'ConfigMgr Release',
Count(DISTINCT sys.ResourceID) as 'Client Count',
(STR((COUNT(sys.ResourceID)*100.0/(
Select COUNT(SYS.ResourceID)
From v_FullCollectionMembership FCM INNER JOIN V_R_System sys on FCM.ResourceID = SYS.ResourceID
Where
FCM.CollectionID = @CollectionID
and
Sys.Client0= '1')),5,2)) + ' %' AS 'Percent %'
From v_FullCollectionMembership FCM INNER JOIN V_R_System sys on FCM.ResourceID = SYS.ResourceID
Where SYS.Client0 = '1' and FCM.CollectionID = @CollectionID
Group By sys.Client_version0
Order by sys.Client_version0 DESC
```

170. Compare packages with Another DP

```sql
select
Pkg.PackageID,
Pkg.Name,
Case Pkg.PackageType
When 0 Then 'Package'
When 3 Then 'Driver'
When 4 Then 'TaskSequence'
When 5 Then 'softwareUpdate'
When 7 Then 'Virtual'
When 8 Then 'Application'
When 257 Then 'Image'
When 258 Then 'BootImage'
When 259 Then 'OS'
Else ' '
END AS 'Type'
from v_Package Pkg
where Pkg.PackageID in
(
select PackageID from v_DistributionPoint where ServerNALPath like  '%Master DP%'
and PackageID not in
(select PackageID from v_DistributionPoint where ServerNALPath like  '%Compare DP%')
)
order by 3
```

171. OneDrivesetup and Groove application report based on Software Inventory

```sql
Declare @CollectionID Varchar(8)
Set @CollectionID = 'SMS00001'

SELECT distinct
Vrs.Netbios_Name0,
Vrs.User_Name0,
Vrs.AD_Site_Name0,
Sf.FileVersion,
Sf.FileName,
Sf.FileVersion
FROM v_R_System Vrs
Inner Join v_GS_SoftwareFile Sf On Vrs.ResourceID = Sf.ResourceID
Inner join v_FullCollectionMembership Fcm on Vrs.ResourceID = Fcm.ResourceId
WHERE Sf.FileName like 'OneDriveSetup.exe' or Sf.FileName like 'groove.exe'
and Fcm.CollectionID = @CollectionID
GROUP BY
Vrs.Netbios_Name0,
Vrs.User_Name0,
Vrs.AD_Site_Name0,
Sf.FileName,
Sf.FileVersion
ORDER BY Vrs.Netbios_Name0
```

172. All SCCM Servers Inventory status

```sql
Select
Distinct (VRS.Netbios_Name0) as 'Server Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 1 Then 'VMWare'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('3','4')Then 'Desktop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('8','9','10','11','12','14') Then 'Laptop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 6 Then 'Mini Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 7 Then 'Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 13 Then 'All in One'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 15 Then 'Space-Saving'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 17 Then 'Main System Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 21 Then 'Peripheral Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 22 Then 'Storage Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 23 Then 'Rack Mount Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 24 Then 'Sealed-Case PC'
Else 'Others' End 'CaseType',
LEFT(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0),
ISNULL(NULLIF(CHARINDEX(',',MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0)) - 1, -
1),LEN(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0))))as 'IPAddress',
MAX (v_GS_NETWORK_ADAPTER_CONFIGUR.MACAddress0) as 'MACAddress',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
VRS.Client_Version0 as 'ClientVersion',
VRS.Creation_Date0 as 'ClientCreationDate',
DateDiff(D, VRS.Creation_Date0, GetDate()) 'ClientCreationDateAge',
VRS.AD_Site_Name0 as 'ADSiteName',
dbo.v_GS_OPERATING_SYSTEM.InstallDate0 AS 'OSInstallDate',
DateDiff(D, v_GS_OPERATING_SYSTEM.InstallDate0, GetDate()) 'OSInstallDateAge',
Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100) as 'LastBootDate',
DateDiff(D, Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100), GetDate()) as 'LastBootDateAge',
PC_BIOS_DATA.SerialNumber00 as 'SerialNumber',
v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0 as 'AssetTag',
PC_BIOS_DATA.ReleaseDate00 as 'ReleaseDate',
PC_BIOS_DATA.Name00 as 'BiosName',
PC_BIOS_DATA.SMBIOSBIOSVersion00 as 'BiosVersion',
```

```sql
v_GS_PROCESSOR.Name0 as 'ProcessorName',
case when Computer_System_DATA.Manufacturer00 like 'VMware%' Then 'VMWare'
when Computer_System_DATA.Manufacturer00 like 'Gigabyte%' Then 'Gigabyte'
when Computer_System_DATA.Manufacturer00 like 'VIA Technologies%' Then 'VIA Technologies'
when Computer_System_DATA.Manufacturer00 like 'MICRO-STAR%' Then 'MICRO-STAR'
Else Computer_System_DATA.Manufacturer00 End 'Manufacturer',
Computer_System_DATA.Model00 as 'Model',
Computer_System_DATA.SystemType00 as 'OSType',
v_GS_COMPUTER_SYSTEM.Domain0 as 'DomainName',
VRS.User_Domain0+'\'+ VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
Case when v_GS_COMPUTER_SYSTEM.domainrole0 = 0 then 'Standalone Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 1 Then 'Member Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 2 Then 'Standalone Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 3 Then 'Member Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 4 Then 'Backup Domain Controller'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 5 Then 'Primary Domain Controller'
End 'Role',
case when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Enterprise Edition' Then 'Microsoft(R)
Windows(R) Server 2003 Enterprise Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Standard Edition' Then 'Microsoft(R) Windows(R)
Server 2003 Standard Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Web Edition' Then 'Microsoft(R) Windows(R)
Server 2003 Web Edition'
Else Operating_System_DATA.Caption00 End 'OSName',
Operating_System_DATA.CSDVersion00 as 'ServicePack',
Operating_System_DATA.Version00 as 'Version',
((v_GS_X86_PC_MEMORY.TotalPhysicalMemory0/1024)/1000) as 'TotalRAMSize(GB)',
max(v_GS_LOGICAL_DISK.Size0 / 1024) AS 'TotalHDDSize(GB)',
v_GS_WORKSTATION_STATUS.LastHWScan as 'LastHWScan',
DateDiff(D, v_GS_WORKSTATION_STATUS.LastHwScan, GetDate()) as 'LastHWScanAge'
from V_R_System VRS
Left Outer join PC_BIOS_DATA on PC_BIOS_DATA.MachineID = VRS.ResourceId
Left Outer join Operating_System_DATA on Operating_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_WORKSTATION_STATUS on v_GS_WORKSTATION_STATUS.ResourceID = VRS.ResourceId
Left Outer join Computer_System_DATA on Computer_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_X86_PC_MEMORY on v_GS_X86_PC_MEMORY.ResourceID = VRS.ResourceId
Left Outer join v_GS_PROCESSOR on v_GS_PROCESSOR.ResourceID = VRS.ResourceId
Left Outer join v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = VRS.ResourceId
Left Outer join v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
Left Outer join v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID = VRS.ResourceId
left outer join v_GS_COMPUTER_SYSTEM on v_GS_COMPUTER_SYSTEM.ResourceID = VRS.ResourceId
left outer join v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
left outer join v_GS_NETWORK_ADAPTER_CONFIGUR on v_GS_NETWORK_ADAPTER_CONFIGUR.ResourceID =
VRS.ResourceId
left outer join v_GS_LOGICAL_DISK on v_GS_LOGICAL_DISK.ResourceID = Vrs.ResourceId AND
v_GS_LOGICAL_DISK.DriveType0 = 3
Left Outer join v_R_User on VRS.User_Name0 = v_R_User.User_Name0
where (VRS.Obsolete0 = 0 or VRS.Obsolete0 is null)
and VRS.Netbios_Name0 in
(
Select LEFT(ServerName, CHARINDEX('.', ServerName) - 1) from v_SystemResourceList where RoleName = 'SMS Site System'
)
GROUP BY VRS.Netbios_Name0,VRS.Client0,VRS.Active0,v_GS_SYSTEM_ENCLOSURE.ChassisTypes0,
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0,VRS.Client_Version0,Vrs.Creation_Date0,
Vrs.AD_Site_Name0,v_Gs_Operating_System.InstallDate0,v_Gs_Operating_System.LastBootUpTime0,
PC_BIOS_DATA.SerialNumber00,v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0,PC_BIOS_DATA.ReleaseDate00,
PC_BIOS_DATA.Name00,PC_BIOS_DATA.SMBIOSBIOSVersion00,v_GS_PROCESSOR.Name0,Computer_System_DATA.Manufact
urer00,Computer_System_DATA.Model00,Computer_System_DATA.SystemType00,v_GS_COMPUTER_SYSTEM.Domain0,
Vrs.User_Domain0,Vrs.User_Name0,v_R_User.Mail0,v_GS_COMPUTER_SYSTEM.DomainRole0,Operating_System_DATA.Caption0
0,Operating_System_DATA.CSDVersion00,Operating_System_DATA.Version00,v_GS_X86_PC_MEMORY.TotalPhysicalMemory0,v_
GS_WORKSTATION_STATUS.LastHWScan
order by VRS.Netbios_Name0
```

173.   All Workstations Assets Inventory details status

```sql
Declare @CollectionID as varchar(8)
Declare @ProjectName as varchar(25)

Set @CollectionID = 'SMS00001'
Set @ProjectName = 'LAB'

Select
Distinct (VRS.Netbios_Name0) as 'Name',
Case when VRS.Client0 = 1 Then 'Yes' Else 'No' End 'Client',
Case when VRS.Active0 = 1 Then 'Yes' Else 'No' End 'Active',
Case when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 1 Then 'VMWare'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('3','4')Then 'Desktop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 IN('8','9','10','11','12','14') Then 'Laptop'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 6 Then 'Mini Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 7 Then 'Tower'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 13 Then 'All in One'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 15 Then 'Space-Saving'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 17 Then 'Main System Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 21 Then 'Peripheral Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 22 Then 'Storage Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 23 Then 'Rack Mount Chassis'
when v_GS_SYSTEM_ENCLOSURE.ChassisTypes0 = 24 Then 'Sealed-Case PC'
Else 'Others' End 'CaseType',
LEFT(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0),
ISNULL(NULLIF(CHARINDEX(',',MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0)) - 1, -
1),LEN(MAX(v_GS_NETWORK_ADAPTER_CONFIGUR.IPAddress0))))as 'IPAddress',
MAX (v_GS_NETWORK_ADAPTER_CONFIGUR.MACAddress0) as 'MACAddress',
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0 as 'AssignedSite',
VRS.Client_Version0 as 'ClientVersion',
VRS.Creation_Date0 as 'ClientCreationDate',
VRS.AD_Site_Name0 as 'ADSiteName',
dbo.v_GS_OPERATING_SYSTEM.InstallDate0 AS 'OSInstallDate',
DateDiff(D, dbo.v_GS_OPERATING_SYSTEM.InstallDate0, GetDate()) 'OSInstallDateAge',
Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100) as 'LastBootDate',
DateDiff(D, Convert(VarChar, v_Gs_Operating_System.LastBootUpTime0,100), GetDate()) as
'LastBootDateAge',
PC_BIOS_DATA.SerialNumber00 as 'SerialNumber',
v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0 as 'AssetTag',
PC_BIOS_DATA.ReleaseDate00 as 'ReleaseDate',
PC_BIOS_DATA.Name00 as 'BiosName',
PC_BIOS_DATA.SMBIOSBIOSVersion00 as 'BiosVersion',
v_GS_PROCESSOR.Name0 as 'ProcessorName',
case when Computer_System_DATA.Manufacturer00 like 'VMware%' Then 'VMWare'
when Computer_System_DATA.Manufacturer00 like 'Gigabyte%' Then 'Gigabyte'
when Computer_System_DATA.Manufacturer00 like 'VIA Technologies%' Then 'VIA Technologies'
when Computer_System_DATA.Manufacturer00 like 'MICRO-STAR%' Then 'MICRO-STAR'
Else Computer_System_DATA.Manufacturer00 End 'Manufacturer',
Computer_System_DATA.Model00 as 'Model',
Computer_System_DATA.SystemType00 as 'OSType',
v_GS_COMPUTER_SYSTEM.Domain0 as 'DomainName',
VRS.User_Domain0+'\'+ VRS.User_Name0 as 'UserName',
v_R_User.Mail0 as 'EMailID',
Case when v_GS_COMPUTER_SYSTEM.domainrole0 = 0 then 'Standalone Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 1 Then 'Member Workstation'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 2 Then 'Standalone Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 3 Then 'Member Server'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 4 Then 'Backup Domain Controller'
when v_GS_COMPUTER_SYSTEM.domainrole0 = 5 Then 'Primary Domain Controller'
End 'Role',
```

```sql
case when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Enterprise Edition'
Then 'Microsoft(R) Windows(R) Server 2003 Enterprise Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Standard Edition' Then
'Microsoft(R) Windows(R) Server 2003 Standard Edition'
when Operating_System_DATA.Caption00 = 'Microsoft(R) Windows(R) Server 2003, Web Edition' Then
'Microsoft(R) Windows(R) Server 2003 Web Edition'
Else Operating_System_DATA.Caption00 End 'OSName',
Operating_System_DATA.CSDVersion00 as 'ServicePack',
Operating_System_DATA.Version00 as 'Version',
((v_GS_X86_PC_MEMORY.TotalPhysicalMemory0/1024)/1000) as 'TotalRAMSize(GB)',
max(v_GS_LOGICAL_DISK.Size0 / 1024) AS 'TotalHDDSize(GB)',
v_GS_WORKSTATION_STATUS.LastHWScan as 'LastHWScan',
DateDiff(D, v_GS_WORKSTATION_STATUS.LastHwScan, GetDate()) as 'LastHWScanAge',
@ProjectName as 'AccountName'
from V_R_System VRS
Left Outer join PC_BIOS_DATA on PC_BIOS_DATA.MachineID = VRS.ResourceId
Left Outer join Operating_System_DATA on Operating_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_WORKSTATION_STATUS on v_GS_WORKSTATION_STATUS.ResourceID = VRS.ResourceId
Left Outer join Computer_System_DATA on Computer_System_DATA.MachineID = VRS.ResourceId
Left Outer join v_GS_X86_PC_MEMORY on v_GS_X86_PC_MEMORY.ResourceID = VRS.ResourceId
Left Outer join v_GS_PROCESSOR on v_GS_PROCESSOR.ResourceID = VRS.ResourceId
Left Outer join v_GS_SYSTEM_ENCLOSURE on v_GS_SYSTEM_ENCLOSURE.ResourceID = VRS.ResourceId
Left Outer join v_Gs_Operating_System on v_Gs_Operating_System .ResourceID = VRS.ResourceId
Left Outer join v_RA_System_SMSAssignedSites on v_RA_System_SMSAssignedSites.ResourceID =
VRS.ResourceId
left outer join v_GS_COMPUTER_SYSTEM on v_GS_COMPUTER_SYSTEM.ResourceID = VRS.ResourceId
left outer join v_FullCollectionMembership on v_FullCollectionMembership.ResourceID = VRS.ResourceId
left outer join v_GS_NETWORK_ADAPTER_CONFIGUR on v_GS_NETWORK_ADAPTER_CONFIGUR.ResourceID =
VRS.ResourceId
left outer join v_GS_LOGICAL_DISK on v_GS_LOGICAL_DISK.ResourceID = Vrs.ResourceId AND
v_GS_LOGICAL_DISK.DriveType0 = 3
Left Outer join v_R_User on VRS.User_Name0 = v_R_User.User_Name0
where VRS.Operating_System_Name_and0 like '%Workstation%'
and (VRS.Obsolete0 = 0 or VRS.Obsolete0 is null)
and v_FullCollectionMembership.CollectionID = @CollectionID
GROUP BY VRS.Netbios_Name0,VRS.Client0,VRS.Active0,v_GS_SYSTEM_ENCLOSURE.ChassisTypes0,
v_RA_System_SMSAssignedSites.SMS_Assigned_Sites0,VRS.Client_Version0,Vrs.Creation_Date0,
Vrs.AD_Site_Name0,v_Gs_Operating_System.InstallDate0,v_Gs_Operating_System.LastBootUpTime0,
PC_BIOS_DATA.SerialNumber00,v_GS_SYSTEM_ENCLOSURE.SMBIOSAssetTag0,PC_BIOS_DATA.ReleaseDate00,
PC_BIOS_DATA.Name00,PC_BIOS_DATA.SMBIOSBIOSVersion00,v_GS_PROCESSOR.Name0,Computer_System_DATA.Manufact
urer00,
Computer_System_DATA.Model00,Computer_System_DATA.SystemType00,v_GS_COMPUTER_SYSTEM.Domain0,
Vrs.User_Domain0,Vrs.User_Name0,v_R_User.Mail0,v_GS_COMPUTER_SYSTEM.DomainRole0,Operating_System_DATA.C
aption00,
Operating_System_DATA.CSDVersion00,Operating_System_DATA.Version00,v_GS_X86_PC_MEMORY.TotalPhysicalMemo
ry0,v_GS_WORKSTATION_STATUS.LastHWScan
order by VRS.Netbios_Name0
```

174.   All ConfigMgr roles status

```sql
Select distinct
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Site System') as 'SiteSys',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Component Server') as 'CompSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Site Server') as 'SiteSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Management Point') as 'MP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Distribution Point') as 'DP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS SQL Server') as 'SQL',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Software Update Point') as 'SUP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS SRS Reporting Point') as 'SSRS',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Reporting Point') as 'RPT',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Fallback Status Point') as 'FSP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Server Locator Point') as 'SLP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS PXE Service Point') as 'PXE',
```

```sql
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS System Health Validator') as 'SysVal',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS State Migration Point') as 'SMP',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Notification Server') as 'NotiSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Provider') as 'SMSPro',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Application Web Service') as 'WebSer',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Portal Web Site') as 'WebSite',
(select COUNT(*) from v_SystemResourceList where RoleName = 'SMS Branch distribution point') as 'BranDP'
from v_SystemResourceList
```

175. All ConfigMgr Roles Detailed status

```sql
SELECT
srl.ServerName,
srl.SiteCode,
vs.SiteName,
vrs.AD_Site_Name0 as ADSite,
vs.ReportingSiteCode as Parent,
vs.Installdir,
MAX(CASE srl.rolename WHEN 'SMS Site System' THEN 'Yes' Else ' ' END) as SiteSys,
MAX(CASE srl.rolename WHEN 'SMS Component Server' THEN 'Yes' Else ' ' END) as CompSer,
MAX(CASE srl.rolename WHEN 'SMS Site Server' THEN 'Yes' Else ' ' END) as SiteSer,
MAX(CASE srl.rolename WHEN 'SMS Management Point' THEN 'Yes' Else ' ' END) as MP,
MAX(CASE srl.rolename WHEN 'SMS Distribution Point' THEN 'Yes' Else ' ' END) as DP,
MAX(CASE srl.rolename WHEN 'SMS SQL Server' THEN 'Yes' Else ' ' END) as 'SQL',
MAX(CASE srl.rolename WHEN 'SMS Software Update Point' THEN 'Yes' Else ' ' END) as SUP,
MAX(CASE srl.rolename WHEN 'SMS SRS Reporting Point' THEN 'Yes' Else ' ' END) as SSRS,
MAX(CASE srl.RoleName WHEN 'SMS Reporting Point' THEN 'Yes' Else ' ' END) as RPT,
MAX(CASE srl.rolename WHEN 'SMS Fallback Status Point' THEN 'Yes' Else ' ' END) as FSP,
MAX(CASE srl.rolename WHEN 'SMS ServerName Locator Point' THEN 'Yes' Else ' ' END) as SLP,
MAX(CASE srl.rolename WHEN 'SMS PXE Service Point' THEN 'Yes' Else ' ' END) as PXE,
MAX(CASE srl.rolename WHEN 'AI Update Service Point' THEN 'Yes' Else ' ' END) as AssI,
MAX(CASE srl.rolename WHEN 'SMS State Migration Point' THEN 'Yes' Else ' ' END) as SMP,
MAX(CASE srl.rolename WHEN 'SMS System Health Validator' THEN 'Yes' Else ' ' END) as SysVal,
MAX(CASE srl.rolename WHEN 'SMS Notification Server' THEN 'Yes' Else ' ' END) as NotiSer,
MAX(CASE srl.rolename WHEN 'SMS Provider' THEN 'Yes'  Else ' ' END) as SMSPro,
MAX(CASE srl.rolename WHEN 'SMS Application Web Service' THEN 'Yes' Else ' ' END) as WebSer,
MAX(CASE srl.rolename WHEN 'SMS Portal Web Site' THEN 'Yes' Else ' ' END) as WebSite,
MAX(CASE srl.rolename WHEN 'SMS Branch distribution point' THEN 'Yes'  Else ' ' END) as BranDP
FROM
v_SystemResourceList as srl
LEFT JOIN v_site vs on srl.ServerName = vs.ServerName
LEFT JOIN v_R_System_Valid vrs on LEFT(srl.ServerName, CHARINDEX('.', srl.ServerName) - 1)  =
vrs.Netbios_Name0
GROUP BY
srl.ServerName,
srl.SiteCode,
vs.SiteName,
vs.ReportingSiteCode,
vrs.AD_Site_Name0,
vs.InstallDir
ORDER BY srl.sitecode,srl.ServerName
```

176. All Software applications deployments status

```sql
Select
Vaa.AssignmentName as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Vaa.ApplicationName as 'ApplicationName',
CASE when Vaa.DesiredConfigType = 1 Then 'Install' when vaa.DesiredConfigType = 2 Then 'Uninstall' Else
'Others' End as 'DepType',
Ds.CollectionName as 'CollectionName',
```

```sql
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' when
Ds.DeploymentIntent = 3 Then 'Simulate' End as 'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_ApplicationAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 1
order by Ds.DeploymentTime desc
```

177.   All Software packages deployments status

```sql
Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Left(Ds.SoftwareName, CharIndex('(',(Ds.SoftwareName))-1)as 'ApplicationName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 2
order by Ds.DeploymentTime desc
```

178.   All Software updates deployments status

```sql
Select
Vaa.AssignmentName as 'DeploymentName',
Ds.CollectionName as 'CollectionName',
```

```sql
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'Others',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberSuccess = 0) or (Ds.NumberSuccess is null) Then '0' Else (round(Ds.NumberSuccess/
convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.CreationTime, GetDate()) as 'CreatedDays',
Vaa.CreationTime as 'CreationTime',
Vaa.LastModificationTime as 'LastModifiedTime',
Vaa.LastModifiedBy as 'LastModifiedBy'
from v_DeploymentSummary Ds
left join  v_CIAssignment Vaa on Ds.AssignmentID = Vaa.AssignmentID
Where Ds.FeatureType = 5
order by Ds.DeploymentTime desc
```

179.   All Operating systems deployments status

```sql
Select
Vaa.AdvertisementName  as 'DeploymentName',
Right(Ds.CollectionName,3) as 'Stage',
Ds.SoftwareName as 'TaskSequenceName',
Ds.ProgramName 'DepType',
Ds.CollectionName as 'CollectionName',
CASE when Ds.DeploymentIntent = 1 Then 'Required' when Ds.DeploymentIntent = 2 Then 'Available' End as
'Purpose',
Ds.DeploymentTime as 'AvailableTime',
Ds.EnforcementDeadline as 'RequiredTime',
Ds.NumberTotal as 'Target',
Ds.NumberSuccess as 'Success',
Ds.NumberInProgress as 'Progress',
Ds.NumberErrors as 'Errors',
Ds.NumberOther as 'ReqNotMet',
Ds.NumberUnknown as 'Unknown',
case when (Ds.NumberTotal = 0) or (Ds.NumberTotal is null) Then '100' Else (round( (Ds.NumberSuccess +
Ds.NumberOther) / convert (float,Ds.NumberTotal)*100,2)) End as 'Success%',
DateDiff(D,Ds.DeploymentTime, GetDate()) as 'AvailableDays',
DateDiff(D,Ds.EnforcementDeadline, GetDate()) as 'RequiredDays',
DateDiff(D,Ds.ModificationTime, GetDate()) as 'CreatedDays',
Ds.CreationTime as 'CreationTime',
Ds.ModificationTime as 'LastModifiedTime',
'Administrator' as 'LastModifiedBy'
from v_DeploymentSummary Ds
join v_Advertisement Vaa on Ds.OfferID  = Vaa.AdvertisementID
Where Ds.FeatureType = 7
order by Ds.DeploymentTime desc
```