**Property Analysis BI Developer - On-boarding Task Report**

1. **Display a list of all property names and their property id’s for Owner Id: 1426.**

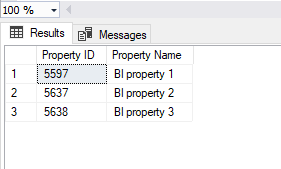
declare @OwnerId int = 1426

SELECT p.Id AS 'Property ID',p.[Name] AS 'Property Name'

FROM dbo.OwnerProperty op INNER JOIN dbo.Property p ON op.PropertyId =p.Id

WHERE op.OwnerId=@OwnerId AND op.OwnershipStatusId =1 /\*Owner\*/

AND p.IsActive =1



**Assumptions & Consideration:**

Considered active properties and also checked ownership status (Owner, Co-Owner, Property Manager) to be an Owner

1. **Display the current home value for each property in question a).**

DECLARE @OwnerId int = 1426

SELECT p.Id as [Property Id], p.[Name] as [Property Name], pv.[Value] as [Property Value]

FROM dbo.OwnerProperty op INNER JOIN dbo.Property p ON op.propertyid =p.id

INNER JOIN (

SELECT v.\*

FROM dbo.PropertyHomeValue v

inner join (

SELECT PropertyId, MAX([date]) as LatestDate

FROM PropertyHomeValue

WHERE IsActive=1

GROUP BY PropertyId

) lp ON v.PropertyId = lp.PropertyId and v.date = lp.LatestDate

WHERE v.HomeValueTypeId = 1 /\*Current Home Value\*/ and v.IsActive=1

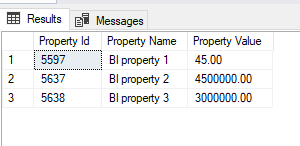
) pv

ON pv.PropertyId=p.id

INNER JOIN dbo.PropertyHomeValueType pvt ON pvt.Id = pv.HomeValueTypeId

WHERE op.OwnerId = @OwnerId AND op.OwnershipStatusId =1 /\*Owner\*/ AND p.IsActive=1

ORDER BY p.Id



**Assumptions & Consideration:**

To get the current home value, consider the latest date for the property ID, checked active status of the property and Ownership status

1. **For each property in question a), return the following:** 
   1. **Using rental payment amount, rental payment frequency, tenant start date and tenant end date to write a query that returns the sum of all payments from start date to end date.**

-- With CTE using sum of PropertyRentalPayment.Amount

WITH ActualPayment(PropertyId, TenantId, FrequencyCode,[StartDate],[EndDate], [Sum of all payments])

AS

(

select tp.PropertyId, tp.TenantId, pf.Code as FrequencyCode, tp.[StartDate],tp.[EndDate], sum(rp.Amount) [Sum of all payments]

FROM [dbo].[TenantProperty] tp

INNER JOIN dbo.[TenantPaymentFrequencies] pf on pf.Id = tp.PaymentFrequencyId

INNER JOIN dbo.[PropertyRentalPayment] rp on rp.PropertyId = tp.PropertyId

GROUP BY tp.PropertyId, tp.TenantId, pf.Code, tp.[StartDate],tp.[EndDate]

),

ExpectedPayment(PropertyId, TenantId, [StartDate],[EndDate], [Expected Payments])

AS

(

SELECT tp.PropertyId, tp.TenantId, tp.StartDate,tp.EndDate,

CASE

WHEN (tp.PaymentFrequencyId = 1 /\*Weekly\*/) THEN DATEDIFF(WEEK,tp.StartDate,tp.EndDate) \* tp.PaymentAmount

WHEN (tp.PaymentFrequencyId = 2 /\*Fortnightly\*/) THEN (DATEDIFF(WEEK,tp.StartDate,tp.EndDate)/2) \* tp.PaymentAmount

ELSE (DATEDIFF(MONTH,tp.StartDate,tp.EndDate) +1) \* tp.PaymentAmount

END AS [Expected Payment]

FROM dbo.TenantProperty tp INNER JOIN dbo.property p ON (tp.PropertyId =p.Id)

INNER JOIN dbo.TenantPaymentFrequencies tpf ON (tpf.Id =tp.PaymentFrequencyId)

INNER JOIN dbo.OwnerProperty op ON op.PropertyId=p.Id

)

SELECT p.Id AS 'Property ID', p.[Name] AS 'Property Name', tp.StartDate, tp.EndDate, ap.[Sum of all payments], ep.[Expected Payments]

FROM dbo.OwnerProperty op INNER JOIN dbo.Property p ON op.propertyid =p.id

INNER JOIN [dbo].[TenantProperty] tp ON tp.PropertyId = p.Id

INNER JOIN ActualPayment ap ON (ap.TenantId = tp.TenantId AND ap.PropertyId = tp.PropertyId)

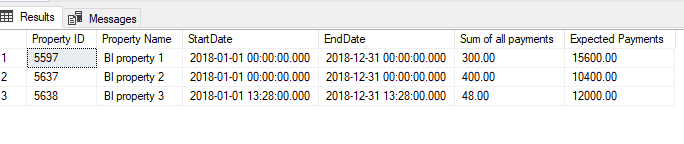
INNER JOIN ExpectedPayment ep ON (ep.TenantId = tp.TenantId AND ep.PropertyId = tp.PropertyId)

WHERE op.OwnerId=1426 AND op.OwnershipStatusId =1 /\*Owner\*/

AND p.IsActive =1

ORDER BY p.Id, p.[Name], tp.[TenantId]

GO



**Assumptions & Consideration:**

* Sum of all payments is taken from Property Rental Property.
* Amount field, Expected payment is calculated using TenantProperty.PaymentAmount field.
* Checked active status and ownership status

Without CTE (using TenantProperty.PaymentAmount )

SELECT tp.PropertyId,p.[Name],tpf.[Name],tp.StartDate,tp.EndDate,tp.PaymentAmount,

CASE

WHEN (tp.PaymentFrequencyId = 1 /\*Weekly\*/) THEN DATEDIFF(WEEK,tp.StartDate,tp.EndDate) \* tp.PaymentAmount

WHEN (tp.PaymentFrequencyId = 2 /\*Fortnightly\*/) THEN (DATEDIFF(WEEK,tp.StartDate,tp.EndDate)/2) \* tp.PaymentAmount

ELSE (DATEDIFF(MONTH,tp.StartDate,tp.EndDate) +1) \* tp.PaymentAmount

END AS 'Expected Payment'

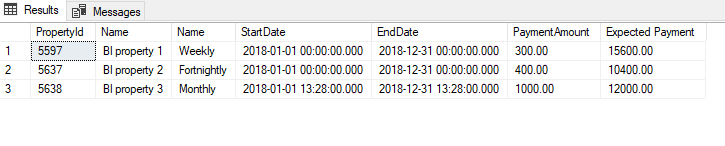
FROM dbo.TenantProperty tp INNER JOIN dbo.property p ON (tp.PropertyId =p.Id)

INNER JOIN dbo.TenantPaymentFrequencies tpf ON (tpf.Id =tp.PaymentFrequencyId)

INNER JOIN dbo.OwnerProperty op ON op.PropertyId=p.Id

WHERE op.OwnerId = 1426 and op.OwnershipStatusId =1 /\*Owner\*/ AND p.IsActive =1

GO



**Assumptions & Consideration:**

* Sum of payment is calculated using from TenantProperty. PaymentAmount field

With CTE Above (using TenantProperty.PaymentAmount )

WITH ActualPayment(PropertyId, TenantId, [StartDate],[EndDate], [PaymentFrequencyId], [PaymentAmount], [Expected Payment])

AS

(

SELECT tp.PropertyId,tp.TenantId,tp.StartDate,tp.EndDate, tp.PaymentFrequencyId, tp.PaymentAmount,

CASE

WHEN (tp.PaymentFrequencyId = 1 /\*Weekly\*/) THEN DATEDIFF(WEEK,tp.StartDate,tp.EndDate) \* tp.PaymentAmount

WHEN (tp.PaymentFrequencyId = 2 /\*Fortnightly\*/) THEN (DATEDIFF(WEEK,tp.StartDate,tp.EndDate)/2) \* tp.PaymentAmount

ELSE (DATEDIFF(MONTH,tp.StartDate,tp.EndDate) +1) \* tp.PaymentAmount

END AS 'Expected Payment'

FROM dbo.TenantProperty tp

)

SELECT tp.PropertyId,p.[Name],tpf.[Name],tp.StartDate,tp.EndDate, tp.[PaymentAmount], tp.[Expected Payment]

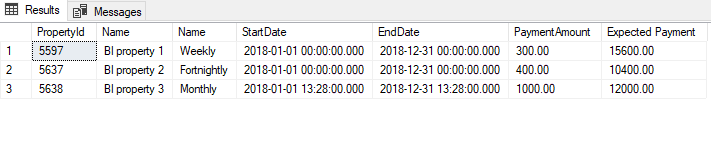
FROM ActualPayment tp INNER JOIN dbo.property p ON (tp.PropertyId =p.Id)

INNER JOIN dbo.TenantPaymentFrequencies tpf ON (tpf.Id =tp.PaymentFrequencyId)

INNER JOIN dbo.OwnerProperty op ON op.PropertyId=p.Id

WHERE op.OwnerId = 1426 and op.OwnershipStatusId =1 /\*Owner\*/ AND p.IsActive =1

* 1. **Display the yield.**

Display Yield

SELECT tp.PropertyId,p.[Name],tpf.[Name],tp.StartDate,tp.EndDate,tp.PaymentAmount,

CASE

WHEN (tp.PaymentFrequencyId = 1 /\*Weekly\*/) THEN DATEDIFF(WEEK,tp.StartDate,tp.EndDate) \* tp.PaymentAmount

WHEN (tp.PaymentFrequencyId = 2 /\*Fortnightly\*/) THEN (DATEDIFF(WEEK,tp.StartDate,tp.EndDate)/2) \* tp.PaymentAmount

ELSE (DATEDIFF(MONTH,tp.StartDate,tp.EndDate) +1) \* tp.PaymentAmount

END AS 'Expected Payment' ,pf.Yield

FROM dbo.TenantProperty tp INNER JOIN dbo.property p ON (tp.PropertyId =p.Id)

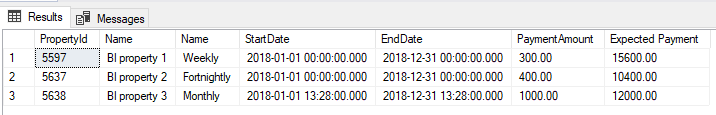
INNER JOIN dbo.TenantPaymentFrequencies tpf ON (tpf.Id =tp.PaymentFrequencyId)

INNER JOIN dbo.OwnerProperty op ON op.PropertyId=p.Id

INNER JOIN dbo.PropertyFinance pf ON pf.propertyid=p.Id

WHERE op.OwnerId = 1426 and op.OwnershipStatusId =1 /\*Owner\*/ AND p.IsActive =1

GO



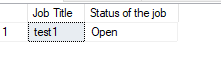
Assumption: If yield is expected from PropertyFinance- yield field. Above query for it.

1. **Display all the jobs available**

SELECT j.JobDescription AS 'Job Title',js.[Status] AS 'Status of the job'

FROM dbo.job j INNER JOIN dbo.JobStatus js ON j.Id=js.Id

WHERE js.id=1 /\* Open\*/



1. **Display all property names, current tenants first and last names and rental payments per week/ fortnight/month for the properties in question a).**

DECLARE @OwnerId int = 1426

SELECT pty.[Id] , pty.[Name] PropertyName,

p.FirstName TenantFirstName,p.LastName AS TenantLastName,

tpf.[Name] AS 'Rental Payment Frequency',prp.Amount AS 'Rental Payment'

FROM dbo.tenant t INNER JOIN dbo.Person p ON t.id=p.Id

INNER JOIN dbo.TenantProperty tp ON tp.TenantId=t.Id

INNER JOIN dbo.Property pty ON pty.Id=tp.PropertyId

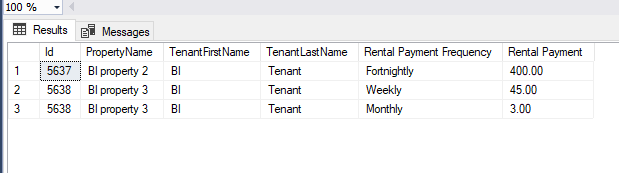
INNER JOIN dbo.PropertyRentalPayment prp ON prp.PropertyId =pty.id

INNER JOIN dbo.TenantPaymentFrequencies tpf ON tpf.id=prp.FrequencyType

INNER JOIN dbo.OwnerProperty op ON op.PropertyId=pty.Id

WHERE t.IsActive =1 and tp.IsActive = 1 AND (tp.EndDate IS NOT NULL OR tp.EndDate >= GETDATE())

AND op.OwnerId=@OwnerId



Assumption:

* Current active tenants are taken by checking the TenantPropertyEndDate.

If null 🡪 Open Tenancy without end date

If has value 🡪 It should not be a past date

**SSRS Report Query**

SELECT OwnerProperty.OwnerId, Person.FirstName + ' ' + Person.LastName AS Name, Property.Id AS PropertyID, Property.Bedroom, Property.Bathroom, TenantProperty.PaymentAmount, CASE

WHEN (TenantProperty.PaymentFrequencyId = 1 /\*Weekly\*/) THEN 'per week'

WHEN (TenantProperty.PaymentFrequencyId = 2 /\*Fortnightly\*/) THEN 'per fortnight'

ELSE 'per month'

END AS 'PaymentDuration',

PropertyExpense.Description AS PropertyExpenseDescription, PropertyExpense.Amount AS PropertyExpenseAmount, PropertyExpense.Date, Property.Name AS PropertyName,

Person.FirstName + ' ' + Person.LastName AS OwnerName,

Address.Number + ' ' + Address.Street + ' ' + Address.Suburb + ' ' + Address.City + ' ' + Address.Region + ' ' + Address.PostCode AS Address, TenantPaymentFrequencies.Name AS [PaymentFrequecy]

FROM OwnerProperty INNER JOIN

Property ON OwnerProperty.PropertyId = Property.Id INNER JOIN

PropertyExpense ON Property.Id = PropertyExpense.PropertyId INNER JOIN

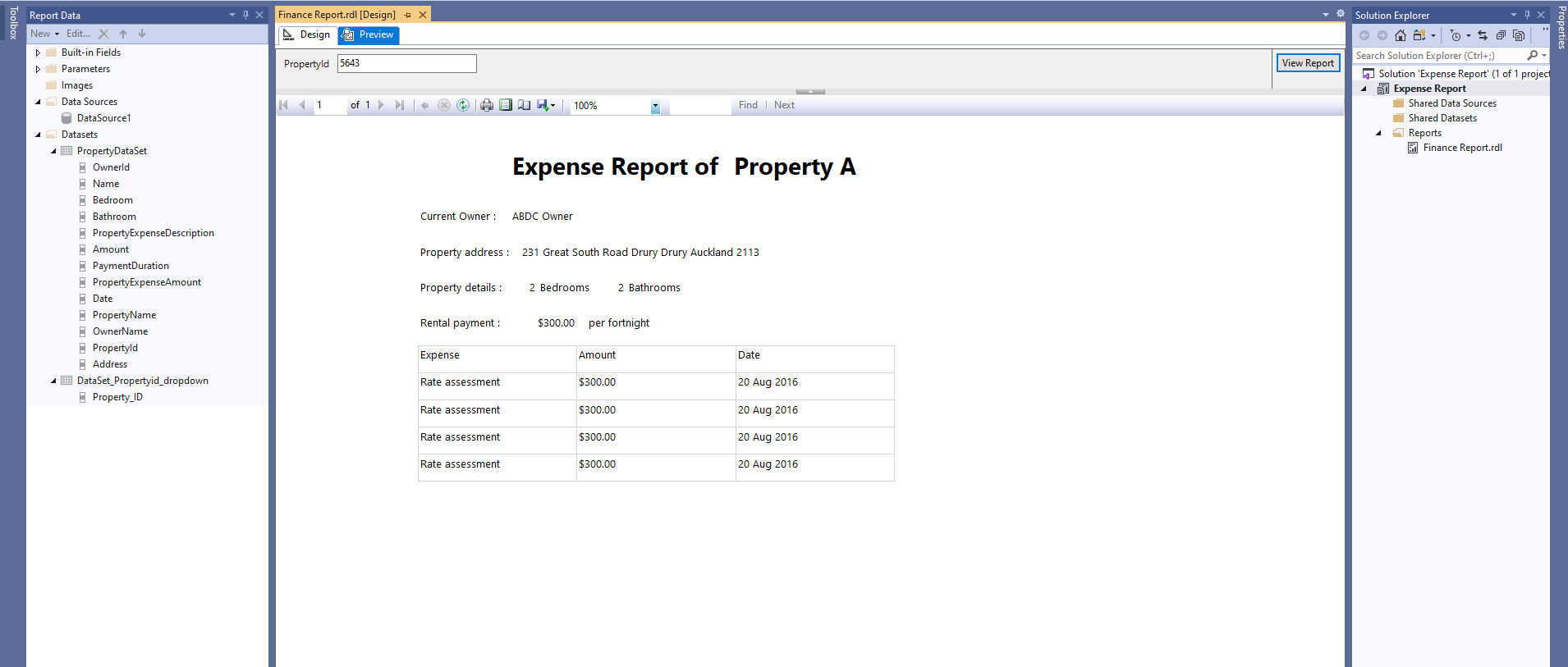
Person ON OwnerProperty.OwnerId = Person.Id INNER JOIN

Address ON Property.AddressId = Address.AddressId INNER JOIN

TenantProperty ON Property.Id = TenantProperty.PropertyId INNER JOIN

TenantPaymentFrequencies ON TenantProperty.PaymentFrequencyId = TenantPaymentFrequencies.Id

WHERE (Property.Id IN (@Para\_Property)) AND Property.IsActive =1



Assumption:

* Check active property using PropertyIsactive.