A picture containing text, clipart

Description automatically generated

Sri Lanka Institute of Information Technology

**Data Warehousing and Business Intelligence**

Assignment 2

2021

**Name: Gavindya N.A.C**

**Registration No: IT20409982**

Contents

[**1.** Data Warehouse Introduction 2](#_Toc103709405)

[**2.** Cube Implementation 3](#_Toc103709406)

[2.1 Procedure for cube Implementation 3](#_Toc103709407)

[2.1.1 Step 1: Creating Data Source 3](#_Toc103709408)

[2.1.2 Step 2: Creating Data View 4](#_Toc103709409)

[2.1.3 Step 3: Creating the Cube 4](#_Toc103709410)

[**3.** Demonstration of OLAP Operations 5](#_Toc103709411)

[3.1 Connection to the SSAS Cube 5](#_Toc103709412)

[3.2 Excel Report for OLAP Operations Demonstrations 6](#_Toc103709413)

[3.2.1 Drill down and rollup demonstration 6](#_Toc103709414)

[3.2.2 Slice Demonstration 9](#_Toc103709415)

[3.2.3 Dice Demonstration 10](#_Toc103709416)

[3.2.4 Pivot Chart 11](#_Toc103709417)

[**4.** SQL Server Reporting Service (SSRS) Reports 13](#_Toc103709418)

[4.1 Basic steps of Report builder 13](#_Toc103709419)

[4.1.1 Step 1: Creating Data source 13](#_Toc103709420)

[4.1.2 Step 2: Creating a dataset 14](#_Toc103709421)

[4.1.3 Step 3: Creating a Matrix or a Table 15](#_Toc103709422)

[4.1.4 Step 4: Report Demonstration 15](#_Toc103709423)

# Data Warehouse Introduction

The data source used here is the ‘**Brooklyn Home Sales’** which is the database which was created as the data warehouse earlier. The data warehouse was created using the Brooklyn Home Sales Dataset which consists of home sales prices of the Brooklyn from 2003 to 2017. This dataset has over 20k of data which through sales fact table and through dimension tables which are Customer, Building, Owner, and Date.

This data source is used for analysis for reporting services and to implement dashboards.

Graphical user interface, application

Description automatically generated

**DataSource**: Brooklyn Home Sales database

Snowflake schema used.

Graphical user interface, application

Description automatically generated

# Cube Implementation

A multidimension data cube is a structure, that contains information for analytical purposes. The main constituents of a cube are dimensions and measures. Mainly this contains two facts,

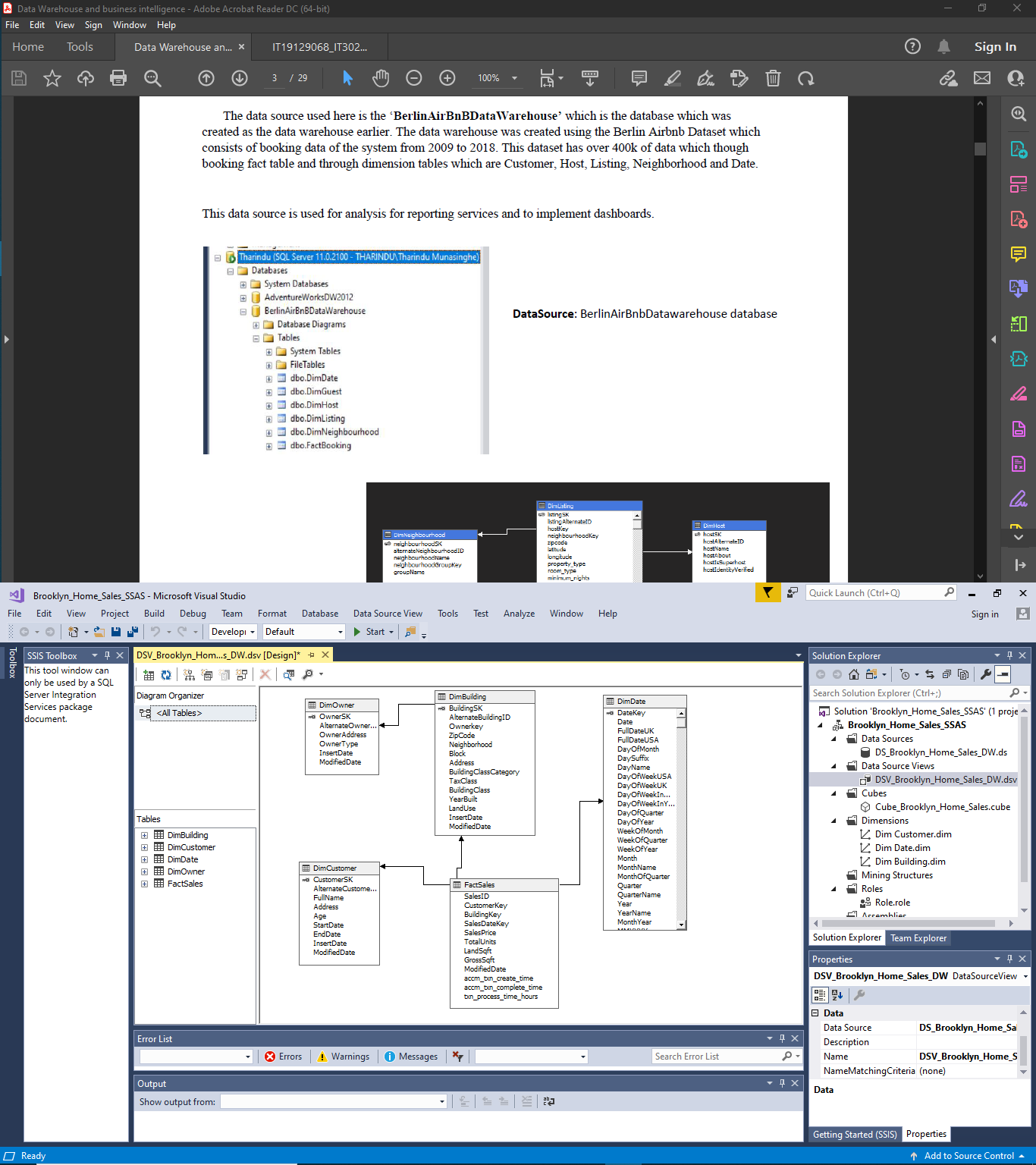
* Dimensions: Define the structure of the cube that you use to slice and dice over.
* Measures: Provide aggregated numerical values of interest to the end user.

For this process SQL Server Data Tools was used.

## 2.1 Procedure for cube Implementation

### 2.1.1 Step 1: Creating Data Source

The Data source was created by connecting to the ‘**Brooklyn Home Sales DW’** database through SQL server management studio. Specific windows username and password was used to connect to the SQL Server management studio



Data Source and Impersonation information

### 2.1.2 Step 2: Creating Data View

This step includes getting the relations and views of our dataset since the analysis service only has access to the relations that are present in this data source view. Therefore, using the data source that was created in the step 1 the data source view is created.

Graphical user interface, application

Description automatically generated

### 2.1.3 Step 3: Creating the Cube

This step includes creation of the cube using our data source view. We will mention the fact sales relation as the fact table as it contains all our measures required for the analysis process.

Graphical user interface, application, table

Description automatically generated

# Demonstration of OLAP Operations

**OLAP** stands for ***Online Analytical Processing*** Server. It is a software technology that allows users to analyze information from multiple database systems at the same time. It is based on multidimensional data model and allows the user to query on multi-dimensional data. An effective OLAP solution solves problems for both business users and IT departments. This is an important part of Business Intelligence, providing powerful capabilities for data mining and trend analysis. OLAP helps to analyze big data amounts from different perspectives rapidly.

For the Demonstration, to connect the excel workbooks and to get the data to the semantic layer we can use MDX query. We can build up the MDX query through SSAS project by browsing data.

In my scenario I have not used MDX query instead I have used the data tab in the ribbon to get data from my SSAS Cube.

## 3.1 Connection to the SSAS Cube

Graphical user interface, application, table, Excel

Description automatically generatedGraphical user interface, application, table, Excel

Description automatically generated

Graphical user interface, application, table, Excel

Description automatically generated

## 3.2 Excel Report for OLAP Operations Demonstrations

### 3.2.1 Drill down and rollup demonstration

* The **Role Up** operation performs, aggregation on a multidimensional data cube either by climbing up the hierarchy or by dimension reduction. Roll-up is like zooming out on the data cubes
* The **Drill Down** operation is carried out either by descending a concept hierarchy for a dimension or by adding a new dimension. This lets a user deploy highly detailed data from a data cube. The drill-down operation is the reverse operation of roll-up. Drill-down is like zooming-in on the data cube

**Roll up - Total Sales Price by Neighborhood Group**

Graphical user interface, application, table, Excel

Description automatically generated

Graphical user interface, application, table, Excel

Description automatically generated**Drill down**

### 3.2.2 Slice Demonstration

A **slice** is a subset of the cubes corresponding to a single value for one or more members of the dimension.

**Slice - Total Sales by Year**

**Graphical user interface, application, table, Excel

Description automatically generated**

### 3.2.3 Dice Demonstration

**Dice** operation is similar to a slice. The difference in dice is you select 2 or more dimensions that result in the creation of a sub-cube. Dicing on the other hand, is more of a zoom feature that selects a subset over all the dimensions, but for specific values of the dimension.

**Graphical user interface, application, table, Excel

Description automatically generated**

### 3.2.4 Pivot Chart

**Pivot Chart – Total Units Sold in Neighborhood by Age**

Table

Description automatically generated

# SQL Server Reporting Service (SSRS) Reports

**SQL Server Reporting Service** (SSRS) is a reporting software that allows you to produce formatted reports with tables in the form of data, images, graphs, and charts. These reports are hosted on a server that can be executed any time using parameters defined by the users.

The web portal of a Reporting Services report server is a web-based experience. In the portal, you can view reports, mobile reports, KPIs, and navigate through the elements in your report server instance. You can also use the web portal to administer a single report server instance.

Graphical user interface, text

Description automatically generated

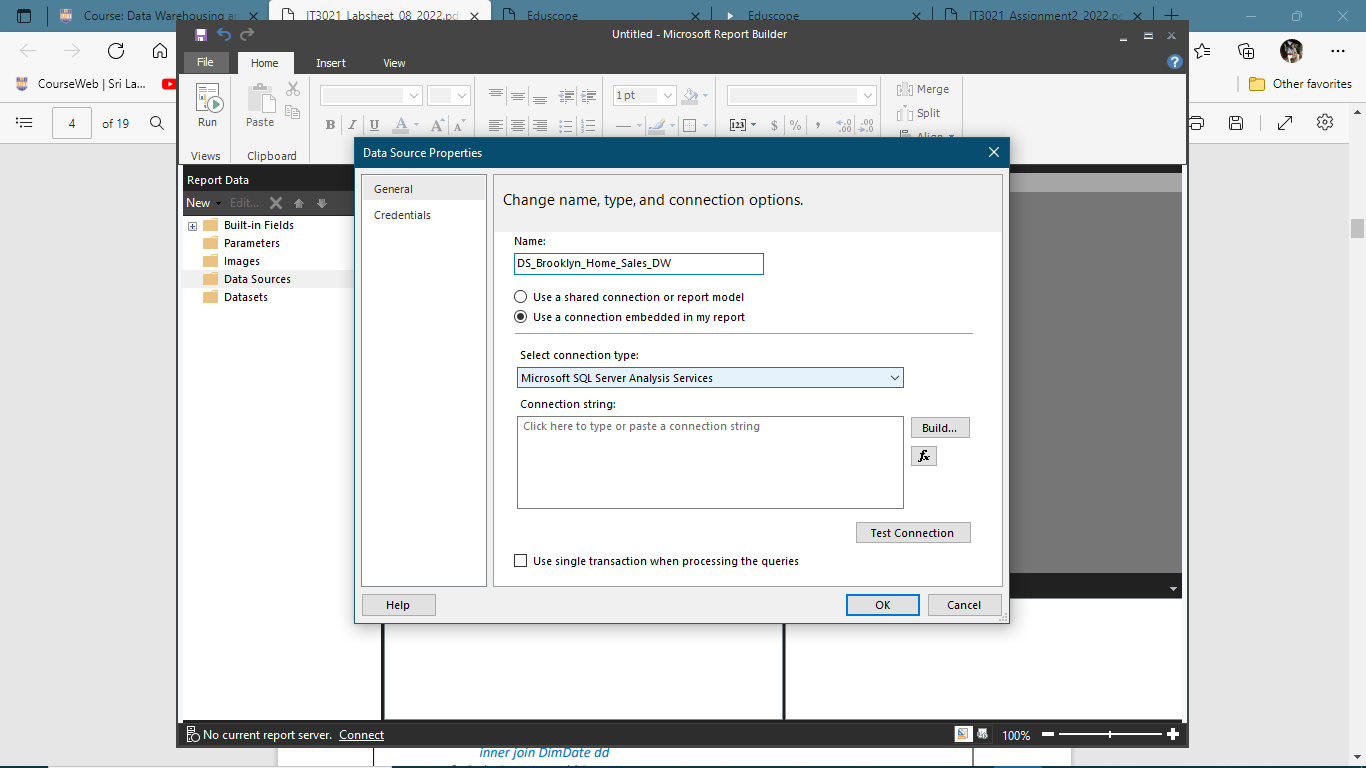
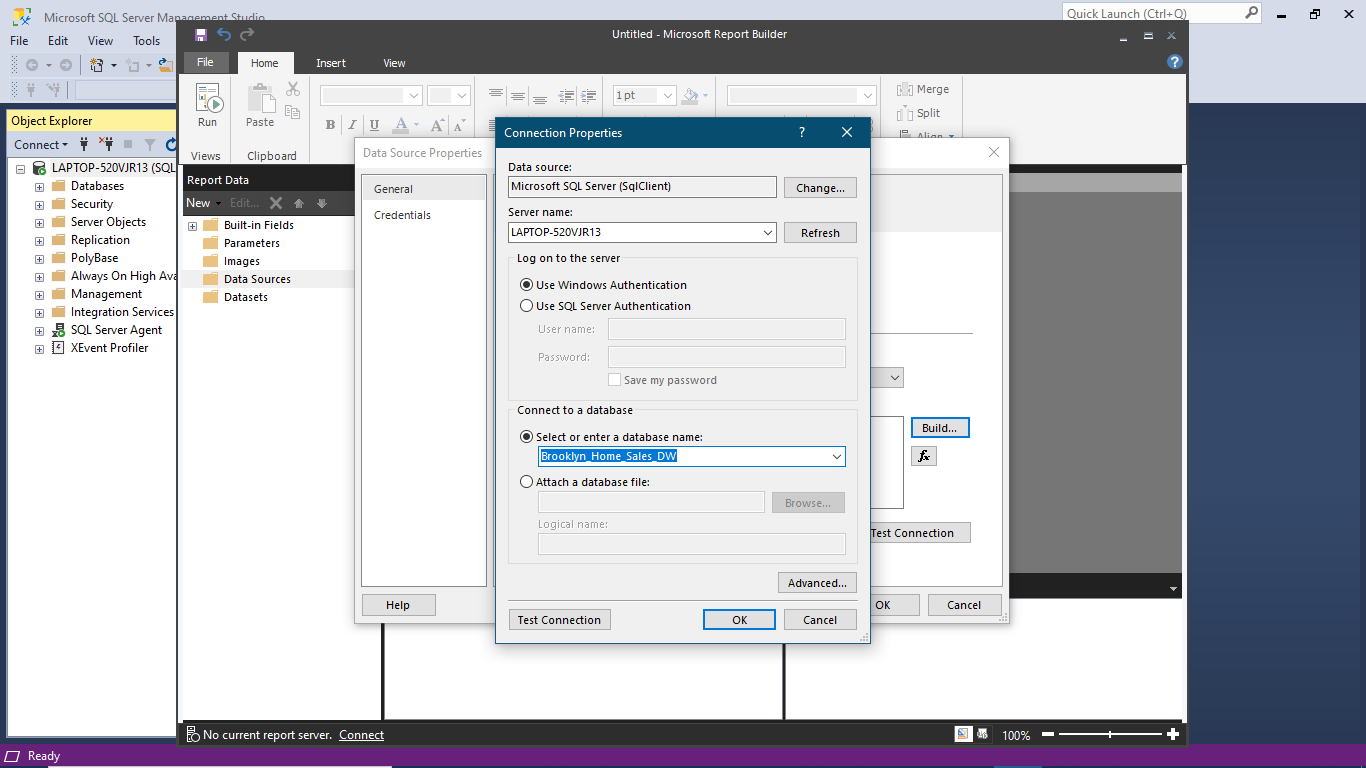
To generate this **“Report Builder”** application was used.

**Report Builder** is a stand-alone app, installed on your computer by you or an administrator. You can install it from the Microsoft Download Center, from a SQL Server 2016 **Reporting** Services or later (SSRS) **report** server, or from a SharePoint site integrated with **Reporting** Services.

## 4.1 Basic steps of Report builder

### 4.1.1 Step 1: Creating Data source

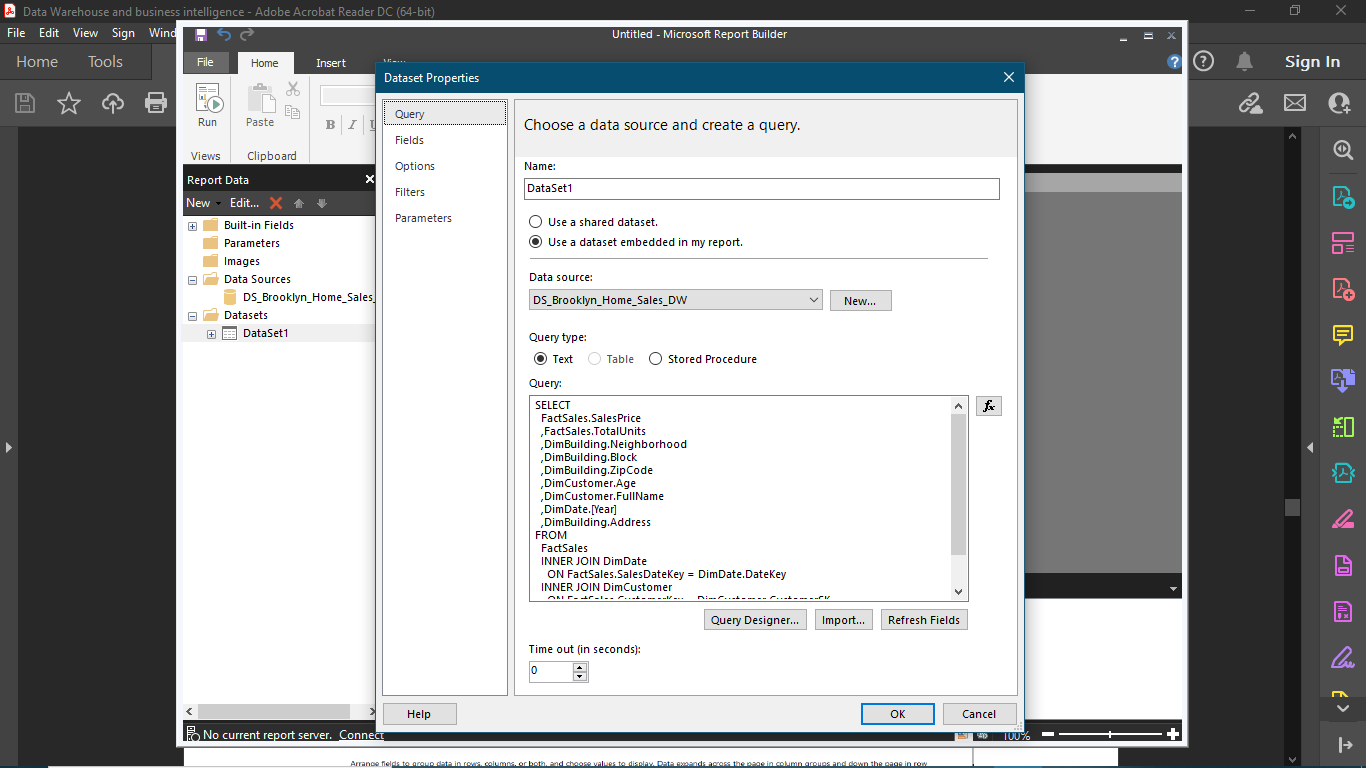
In this step we will be connecting to our data source ‘Brooklyn Home Sales’ which is the data warehouse created earlier.



**4.1.2**

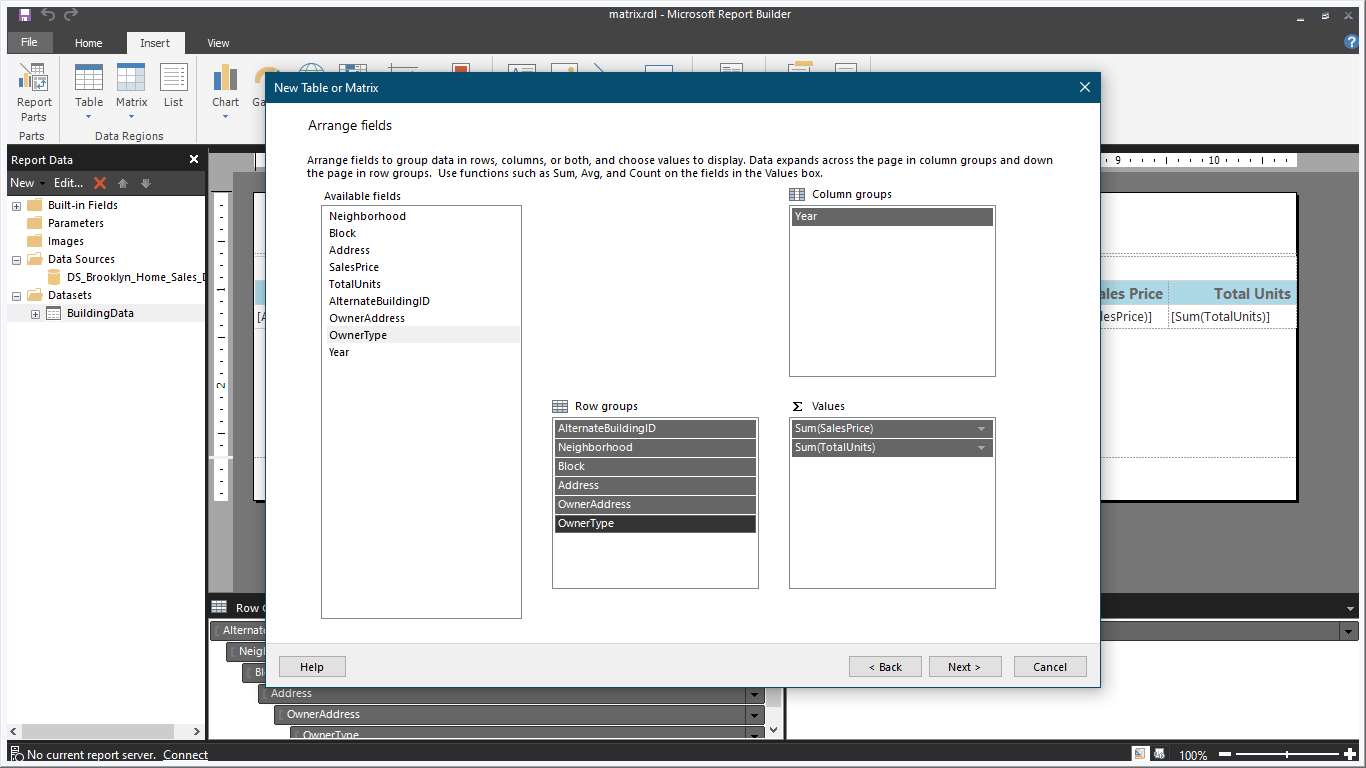
### 4.1.2 Step 2: Creating a dataset

In this step a necessary dataset will be imported to our report builder using a query which will assigned manually according to the requirements



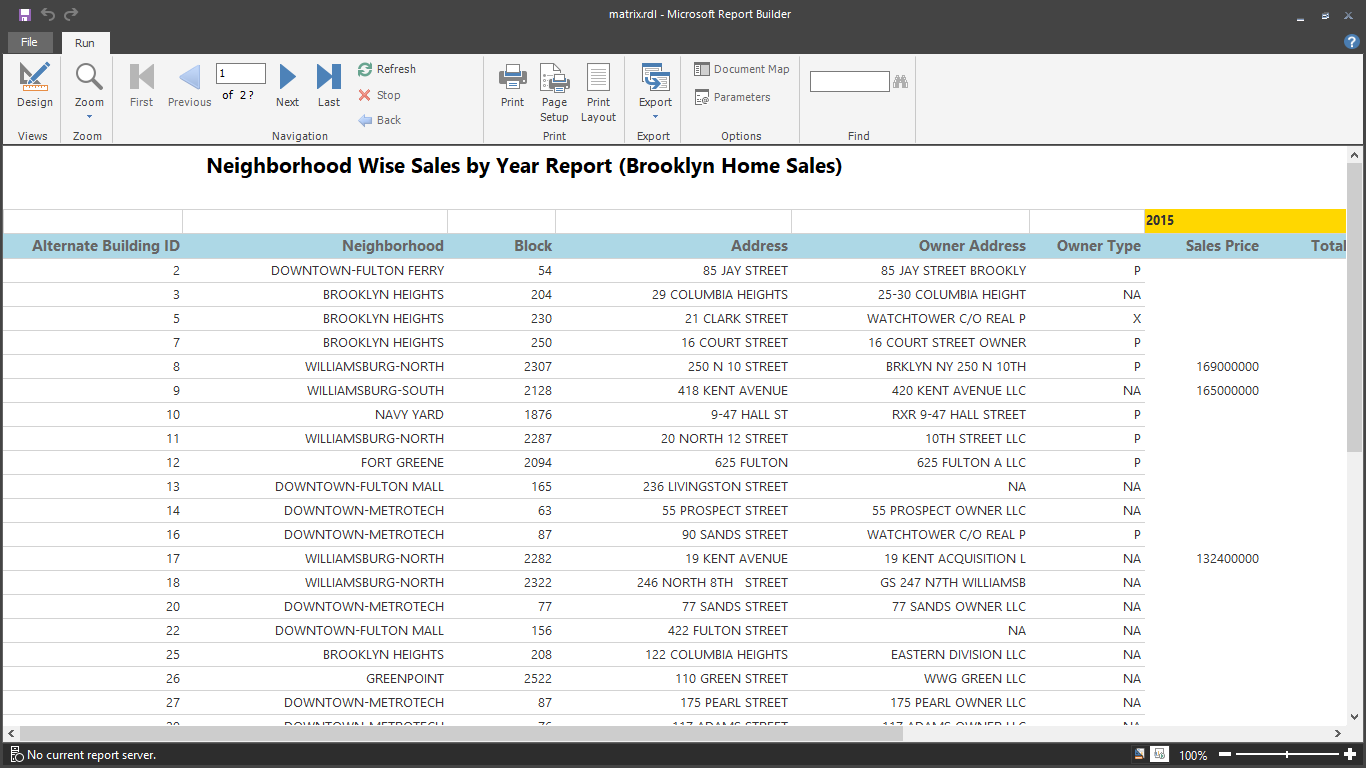
### 4.1.3 Step 3: Creating a Matrix or a Table

Using the imported dataset, we will create a matrix or a table according to our requirements



### 4.1.4 Step 4: Report Demonstration

#### 4.1.4.1 Matrix – Total Sales by Neighborhood Report

 In SSRS Matrix is very similar to a table, but it is configured to show data grouped by columns and rows, with aggregate data at the intersection. This is like using a pivot table in a spreadsheet.

#### 4.1.4.2 Multi-Parameters – Total Sales by neighborhood and Block

In SSRS using Multiparameter-values allows us to pass either one or more than the input value to the report. Also, it offers a “Select All” option that helps to select all parameter values.

Graphical user interface, text

Description automatically generatedGraphical user interface, text, application

Description automatically generated

*Selecting Blocks according to the selected Neighborhood*

*Selecting Neighborhoods*

* + Graphical user interface, text

    Description automatically generatedHere we have created two parameter and three data sets as following.
    - * DataSet1: This data set contains the Sales Price data which will be displayed in the report through the matrix.

Query: SELECT

DimDate.[Year]

,DimBuilding.Neighborhood

,DimBuilding.Block

,DimBuilding.Address

,FactSales.SalesPrice

,FactSales.TotalUnits

,DimCustomer.Age

FROM

FactSales

INNER JOIN DimDate

ON FactSales.SalesDateKey = DimDate.DateKey

INNER JOIN DimCustomer

ON FactSales.CustomerKey = DimCustomer.CustomerSK

INNER JOIN DimBuilding

ON FactSales.BuildingKey = DimBuilding.BuildingSK

where DimBuilding.Block IN (@Block)

* + - * Neighborhood list: This data set contains the required Neighborhood names which is needed to be shown for

selection as the 1st parameter.

**Query**: select distinct Neighborhood

from DimBuilding

* + Block\_list: This contains the Blocks which are in the selected Neighborhood.

When the Neighborhood names are selected, they will be passed as a parameter to this query and the relevant data will be retrieved from the database.

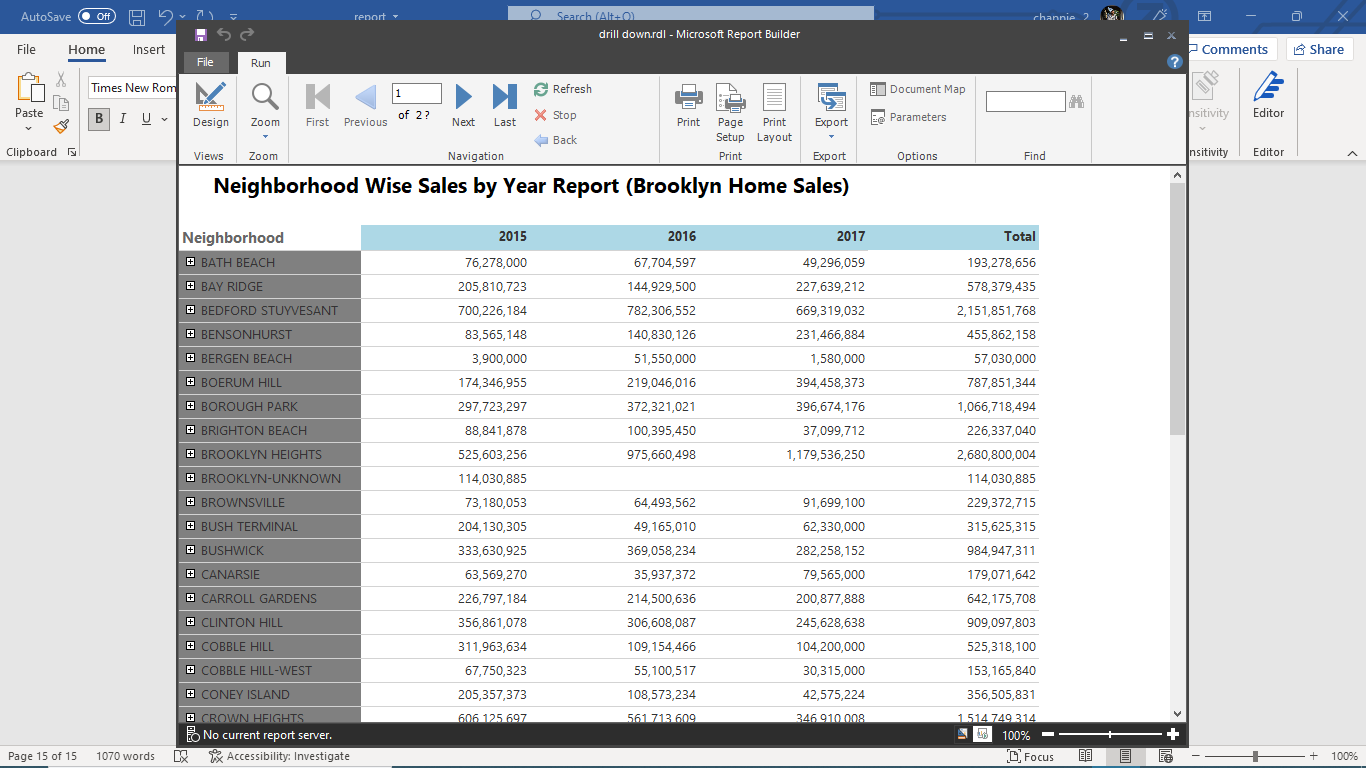
**Query**: select Neighborhood, Block

from DimBuilding

where Neighborhood IN (@Neighborhood)

#### 4.1.4.3 Drill down – Total Sales by Neighborhood Group

In SSRS using Drill Down means allowing users to show or hide the column data by providing plus and minus symbols on a textbox (In short, providing interactivity to the user.).



*When you click on the plus mark in front of the Neighborhood names, it will display the details of Blocks in that Neighborhood*

Graphical user interface, application, table

Description automatically generated*.*

#### Graphical user interface, application Description automatically generated4.1.4.4 Drill through – Total Earnings by Neighborhood Group

Graphical user interface, application

Description automatically generated