Data Preparation with Power BI

Reza Rad
About Reza Rad

DW/BI Consultant, Mentor, Trainer
SQL Server MVP
Author of SQL Server and BI books
Author of Power BI from Rookie to Rock Star
Microsoft Certified Trainer
Microsoft Certified Professional
Co-Leader of NZ BI User Group

@Rad_Reza
rezarad@radacad.com
www.radacad.com/blog
www.linkedin.com/in/rezarad
Agenda

Why Data Preparation?
Introduction to Power Query
Power Query Formula Language: M
Top Five M Functionalities
Date Dimension with Power Query
Two use case scenarios;
  Fitbit Data Integration
  Online Course Date/Time Scheduler
Steps To Data Insight

Data Preparation

Data Modeling

Data Visualization
Building Block; Keystone

Data Preparation is the very first step in Data Insight

Good Data Preparation => Useful Insight
Bad Data Preparation => Lots of Re-Work
Data Preparation

Getting Data from Multiple Sources
Data Cleansing
Data Transformation

Preparing Data for Modeling
Data Can Be Like This
Simpler; Better
DO NOTs

DO NOT add tables/files as is
DO NOT bring 100s of tables unchanged
DO NOT leave naming as is
DO NOT leave data types as is
DO NOT load the whole data set if you don’t require it
Why?

Tables can be joined together to create more flatten and simpler structure => Better Modeling
Name your tables and columns for the end user => Better End User Experience, Better Q&A
Proper Data Types makes modelling easier => Better Modeling
Filtering part of the data before loading it into memory is cost effective => Better Performance
DO

Create flatten structure for tables (specially dimensions)
Name tables and columns as you design table in restaurant
Set Proper Data Types based on the data in each field
Remove Outliers and null/blanks/data quality issues that your model might suffer from
Filter Part of the data that is not required.
Data Preparation Tool

For Data Analyst: Power Query

Power Query for Power BI
Power Query for Excel
Introduction to Power Query
Comes with Different Shapes and Sizes

Power BI

Excel 2016

Excel 2013, 2010
How to Get it?

Power BI Desktop
Built-in for Excel 2016
Free Add-in download for Excel 2013, and 2010
Wide Range of Data Sources Supported

- From Web
- From SharePoint List
- From OData Feed
- From Hadoop File (HDFS)
- From Active Directory
- From Dynamics CRM Online
- From Microsoft Exchange
- From Facebook
- From Salesforce Objects
- From Salesforce Reports
- From ODBC
- Blank Query
- From Microsoft Azure SQL Database
- From Microsoft Azure Marketplace
- From Microsoft Azure HDInsight
- From Microsoft Azure Blob Storage
- From Microsoft Azure Table Storage
- From SQL Server Database
- From Microsoft Access Database
- From SQL Server Analysis Services Database
- From Oracle Database
- From IBM DB2 Database
- From MySQL Database
- From PostgreSQL Database
- From Sybase Database
- From Teradata Database
Data Transformations on GUI
Load to Data Model

Load To

Select how you want to view this data in your workbook.
- [ ] Table
- [ ] Only Create Connection

Select where the data should be loaded.
- [ ] New worksheet
- [ ] Existing worksheet: $A$1

Add this data to the Data Model

[Load] [Cancel]
Movies Data Mash Up
Power Query Formula
Language: M
What is M?

Everything that Happens on GUI works with a Code Behind
Code Behind is a Functional Language: M

```
let TableA = #table({"CustomerId", "TranDate","TranCount"},
{,
 {1,DateTime.FromText("2014-01-01 01:00:00.000"),10},
 {1,DateTime.FromText("2014-01-01 02:00:00.000"),5},
 {1,DateTime.FromText("2014-01-03 01:00:00.000"),5},
 {1,DateTime.FromText("2014-01-04 02:00:00.000"),80}
 },

 TableB = #table({"CustomerId", "TranDate","TranCount"},
 {,
 {1,DateTime.FromText("2014-01-01 02:00:00.000"),20},
 {1,DateTime.FromText("2014-01-01 03:00:00.000"),5},
 {1,DateTime.FromText("2014-01-02 01:00:00.000"),20},
 {1,DateTime.FromText("2014-01-02 03:00:00.000"),15},
 {1,DateTime.FromText("2014-01-03 01:00:00.000"),5},
 {1,DateTime.FromText("2014-01-03 02:00:00.000"),80}
 },

 TableATransformed=Table.Sort(
 Table.AddColumn(TableA,"Date",each Date.From([TranDate]))
 ,{"CustomerId","TranDate"}
 )

 in
 Table.Group(TableATransformed,{"CustomerId","Date"},{"Total",each List.Last([TranCount])})
```
More about Formula Language

M is much more powerful than Power Query GUI;
Not all functionalities of M implemented through GUI.
If you want to be Professional In Power Query; You should be Expert in M
M is a functional language
Structure of M

```
let Source = "Hello World!"
in
Source
```
let
  x = 1
in
  x
M is Case Sensitive
Three Base Structures in M

Table

<table>
<thead>
<tr>
<th>CustomerId</th>
<th>TranDate</th>
<th>TranCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/01/2014 1:00:00 a.m.</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>1/01/2014 2:00:00 a.m.</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>3/01/2014 1:00:00 a.m.</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>4/01/2014 2:00:00 a.m.</td>
<td>80</td>
</tr>
</tbody>
</table>

List

<table>
<thead>
<tr>
<th>List</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
</tbody>
</table>

Record

<table>
<thead>
<tr>
<th>CustomerId</th>
<th>TranDate</th>
<th>TranCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/01/2014 2:00:00 a.m.</td>
<td></td>
</tr>
</tbody>
</table>
Top 5 M Functionalities
Top Five M Functionalities

1. Robust List of Built-in Functions
2. Custom Functions
3. Error Handling
4. EACH; Singleton Function
5. Generator Functions
Robust Built-in Functions

Table Functions
Date Functions

#shared keyword

<table>
<thead>
<tr>
<th>Table Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table.AddColumn</td>
</tr>
<tr>
<td>Table.RemoveColumns</td>
</tr>
<tr>
<td>Table.ReorderColumns</td>
</tr>
<tr>
<td>Table.SelectColumns</td>
</tr>
<tr>
<td>Table.ReplaceValue</td>
</tr>
<tr>
<td>Table.Sort</td>
</tr>
<tr>
<td>Table.AddIndexColumn</td>
</tr>
<tr>
<td>Table.FillDown</td>
</tr>
</tbody>
</table>
Custom Functions

Re-use your code
Reduce redundancy
Increase consistency
Lambda Expression

\[(x) \implies x + 1\]

Input Parameters \implies Function Body

\[(x,y) \implies x + y\]
Parameters

Parameters (optional, required)

```
let Func = (optional x) => if x is null then "nothing" else x
in Func(5)
```
Multi-Line Functions

Casecade let/in clause

```javascript
let
Func = (x) => {
    let
    <body>
    in
    <return_value>
in
Func(5)
```
Generators

List can be generated

List.Dates(<start date>,<end date>)

Generates List of Dates from <start date> to <end date>

It can be used as Loop structure
EACH; Singleton Function

Single Valued parameter
Can be applied on transforms to the items in List or Table

```csharp
DaysInMonthList = List.Transform(MonthList, each
Date.DaysInMonth(DateTime.FromText(Text.From(Year) & "-
& Text.From(_)) & "-01")))
```
Day Number of Year
Day Number of Year

2016-05-03
YYYY-MM-DD

Fetch All Months Before this month

01  31
02  29
03  31
04  30

121 + 3 = 124
Error Handling

Control of Execution when error happens
Sending proper error messages
Preventing failure with messages for troubleshooting
Error Handling
Date Dimension with Power Query
## Date Dimension

<table>
<thead>
<tr>
<th>#</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
<th>FullDateString</th>
<th>DateKey</th>
<th>FiscalYear</th>
<th>FiscalQuarter</th>
<th>CalendarQuarter</th>
<th>IsoWeekDay</th>
<th>DayOfWeek</th>
<th>MonthName</th>
<th>DayOfWeekName</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2013</td>
<td>6</td>
<td>15</td>
<td>6/15/2013</td>
<td>20130615</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Saturday</td>
</tr>
<tr>
<td>2</td>
<td>2013</td>
<td>6</td>
<td>16</td>
<td>6/16/2013</td>
<td>20130616</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Sunday</td>
</tr>
<tr>
<td>3</td>
<td>2013</td>
<td>6</td>
<td>17</td>
<td>6/17/2013</td>
<td>20130617</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Monday</td>
</tr>
<tr>
<td>4</td>
<td>2013</td>
<td>6</td>
<td>18</td>
<td>6/18/2013</td>
<td>20130618</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Tuesday</td>
</tr>
<tr>
<td>5</td>
<td>2013</td>
<td>6</td>
<td>19</td>
<td>6/19/2013</td>
<td>20130619</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Wednesday</td>
</tr>
<tr>
<td>6</td>
<td>2013</td>
<td>6</td>
<td>20</td>
<td>6/20/2013</td>
<td>20130620</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Thursday</td>
</tr>
<tr>
<td>7</td>
<td>2013</td>
<td>6</td>
<td>21</td>
<td>6/21/2013</td>
<td>20130621</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Friday</td>
</tr>
<tr>
<td>8</td>
<td>2013</td>
<td>6</td>
<td>22</td>
<td>6/22/2013</td>
<td>20130622</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Saturday</td>
</tr>
<tr>
<td>9</td>
<td>2013</td>
<td>6</td>
<td>23</td>
<td>6/23/2013</td>
<td>20130623</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Sunday</td>
</tr>
<tr>
<td>10</td>
<td>2013</td>
<td>6</td>
<td>24</td>
<td>6/24/2013</td>
<td>20130624</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Monday</td>
</tr>
<tr>
<td>11</td>
<td>2013</td>
<td>6</td>
<td>25</td>
<td>6/25/2013</td>
<td>20130625</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Tuesday</td>
</tr>
<tr>
<td>12</td>
<td>2013</td>
<td>6</td>
<td>26</td>
<td>6/26/2013</td>
<td>20130626</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Wednesday</td>
</tr>
<tr>
<td>13</td>
<td>2013</td>
<td>6</td>
<td>27</td>
<td>6/27/2013</td>
<td>20130627</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Thursday</td>
</tr>
<tr>
<td>14</td>
<td>2013</td>
<td>6</td>
<td>28</td>
<td>6/28/2013</td>
<td>20130628</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>6 June</td>
<td>6 June</td>
<td>Friday</td>
</tr>
<tr>
<td>15</td>
<td>2013</td>
<td>6</td>
<td>29</td>
<td>6/29/2013</td>
<td>20130629</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Saturday</td>
</tr>
<tr>
<td>16</td>
<td>2013</td>
<td>6</td>
<td>30</td>
<td>6/30/2013</td>
<td>20130630</td>
<td>2013</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6 June</td>
<td>6 June</td>
<td>Sunday</td>
</tr>
<tr>
<td>17</td>
<td>2013</td>
<td>7</td>
<td>1</td>
<td>7/1/2013</td>
<td>20130701</td>
<td>2013</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3 July</td>
<td>3 July</td>
<td>Monday</td>
</tr>
<tr>
<td>18</td>
<td>2013</td>
<td>7</td>
<td>2</td>
<td>7/2/2013</td>
<td>20130702</td>
<td>2013</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3 July</td>
<td>3 July</td>
<td>Tuesday</td>
</tr>
<tr>
<td>19</td>
<td>2013</td>
<td>7</td>
<td>3</td>
<td>7/3/2013</td>
<td>20130703</td>
<td>2013</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3 July</td>
<td>3 July</td>
<td>Wednesday</td>
</tr>
<tr>
<td>20</td>
<td>2013</td>
<td>7</td>
<td>4</td>
<td>7/4/2013</td>
<td>20130704</td>
<td>2013</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>4 July</td>
<td>4 July</td>
<td>Thursday</td>
</tr>
<tr>
<td>21</td>
<td>2013</td>
<td>7</td>
<td>5</td>
<td>7/5/2013</td>
<td>20130705</td>
<td>2013</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5 July</td>
<td>5 July</td>
<td>Friday</td>
</tr>
<tr>
<td>22</td>
<td>2013</td>
<td>7</td>
<td>6</td>
<td>7/6/2013</td>
<td>20130706</td>
<td>2013</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>6 July</td>
<td>6 July</td>
<td>Saturday</td>
</tr>
</tbody>
</table>
### Features

- Configurable via table in Excel Sheet
- Fiscal Columns
- Public Holidays Fetched Live
How it works

Generators; to build the base structure
EACH; to apply transformations on each item in list
Date/Text Functions
OData
Table Functions
Public Holidays


<table>
<thead>
<tr>
<th>Date</th>
<th>Holiday</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wednesday, January 1</td>
<td>New Year’s Day</td>
</tr>
<tr>
<td>Monday, January 20</td>
<td>Birthday of Martin Luther King, Jr.</td>
</tr>
<tr>
<td>Monday, February 17*</td>
<td>Washington’s Birthday</td>
</tr>
<tr>
<td>Monday, May 25</td>
<td>Memorial Day</td>
</tr>
<tr>
<td>Friday, July 3**</td>
<td>Independence Day</td>
</tr>
<tr>
<td>Monday, September 7</td>
<td>Labor Day</td>
</tr>
<tr>
<td>Monday, October 12</td>
<td>Columbus Day</td>
</tr>
<tr>
<td>Wednesday, November 11</td>
<td>Veterans Day</td>
</tr>
<tr>
<td>Thursday, November 26</td>
<td>Thanksgiving Day</td>
</tr>
<tr>
<td>Friday, December 25</td>
<td>Christmas Day</td>
</tr>
</tbody>
</table>

*This holiday is designated as “Washington’s Birthday” in section 6103(a) of title 5 of the United States Code, which is the law that specifies holidays for Federal employees. Although other institutions such as state and local governments and private businesses may use other names, it is our policy to always refer to holidays by the names designated in the law.

**July 4, 2020 (the legal public holiday for Independence Day) falls on a Saturday. For most Federal employees, Friday, July 3, will be treated as a holiday for pay and leave purposes. (See 5 U.S.C. 6103(b).)
Date Dimension
Fitbit Data Integration
Fitbit

The Fitbit dashboard shows various health and fitness statistics, including steps, calories burned, and active minutes. The image also displays a range of Fitbit devices, from the lightweight Zip to the more advanced Blaze and Surge smartwatches.
Problem

Data Manipulation for report generation

Integrating Multiple Files
Problem

Announce Online Courses in Different Time Zones
Dynamic Time Zone Table Generation for each Event/Course
Summary

- Introduction to Power Query
- Power Query Formula Language: M
- Custom Function, Generators, Built-in Functions, Error Handling
- Date Dimension, Fitbit Integration, TimeZone Scheduler
References to Study More

The Demos From THIS webinar:

Power Query and Power BI posts on my blog:

Power Query Formula Categories online Help:

PDFs to Learn M on MSDN:

Power Query for Power BI and Excel (Book, Author; Chris Webb):
http://www.apress.com/9781430266914

M for Data Monkey (Book, Author; Ken Puls):
Q & A

@Rad_Reza
reza@radacad.com
www.radacad.com/blog
www.linkedin.com/in/rezarah
Thank You