

Reference vs Duplicate

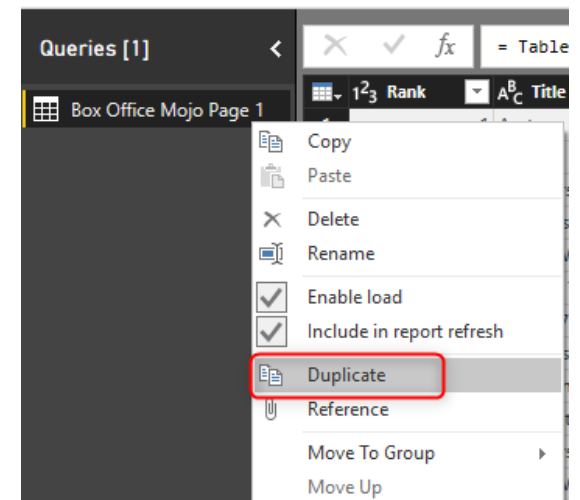
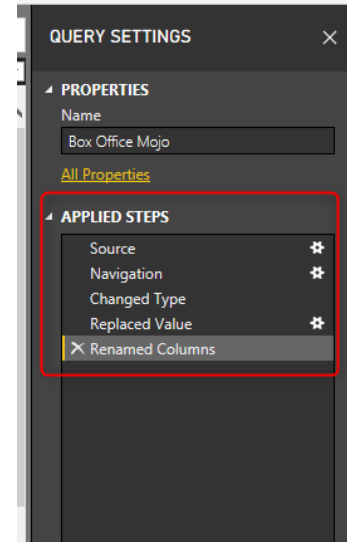
Quando si utilizzano tabelle e query in Power Query in Power BI, è possibile copiarle tramite queste azioni:

- Duplica (Duplicate)
- Riferimento (Reference)

Mi è sempre stata fatta la domanda su quale sia la vera differenza tra queste due azioni. La spiegazione è semplice ma molto importante da capire. Perché quando conosci la differenza, la userai correttamente.

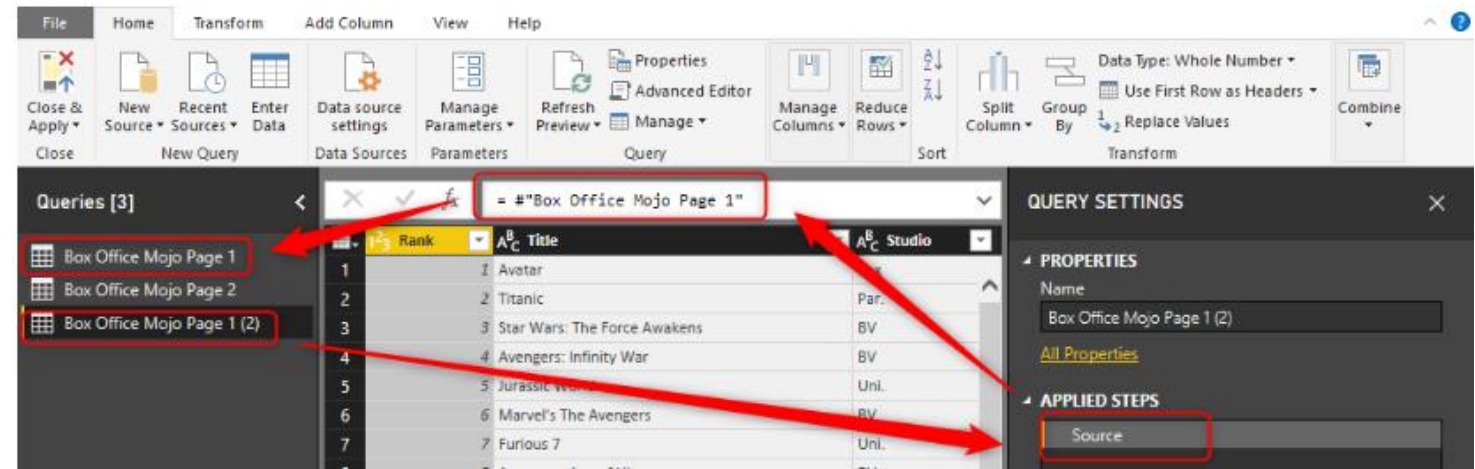
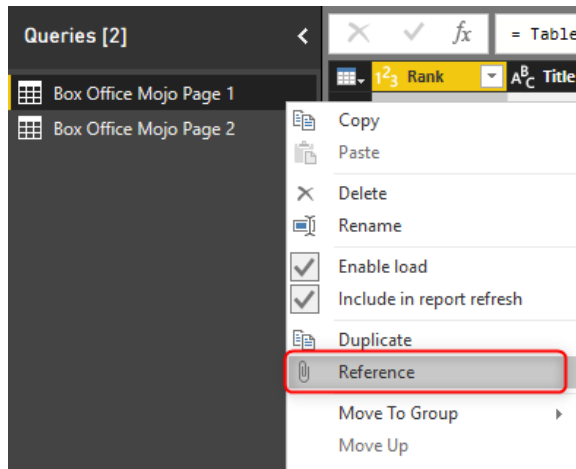
Reference vs Duplicate

- Se stai cercando di copiare un'intera query con tutti i suoi passaggi, Duplicate è il tuo amico.
- Ad esempio, supponiamo che abbiamo ottenuto i dati da una pagina web che ci mostra le informazioni sui film del best seller. Il link di riferimento è <https://www.boxofficemojo.com/alltime/world/?pagenum=1&p=.htm>
- Dopo aver effettuato delle trasformazioni ti rendi conto che i dati del link indicato sono solo per i primi cento film perché quella pagina web non contiene i film rimanenti. Per ottenere gli altri, è necessario passare alla pagina 2, che ha un URL diverso (**`pagenum=2`**), ma la stessa struttura di dati.
- Bene, cosa devi fare? Dobbiamo rifare tutti i passaggi di trasformazione anche per la pagina 2? **Non Serve** possiamo sfruttare Duplicate.



Reference vs Duplicate

- Il riferimento è un altro modo di copiare una query, tuttavia, la grande differenza è che quando si fa riferimento a una query, la nuova query avrà solo un passaggio: un puntamento alla query originale.
- Una query referenziata **non avrà i passaggi applicati della query originale da utilizzare quando** NON vogliamo cambiare nessuna delle query esistenti, perché vogliamo usarle come fonte per altre operazioni.
- L'unico passaggio è il link alla sorgente dati dalla query originale. Cosa significa? Significa che se si apportano modifiche alla query originale, questa nuova query avrà un impatto su tutte le query collegate.



Recap Reference vs Duplicate

Isolamento dall'originale o dipendenza dall'originale

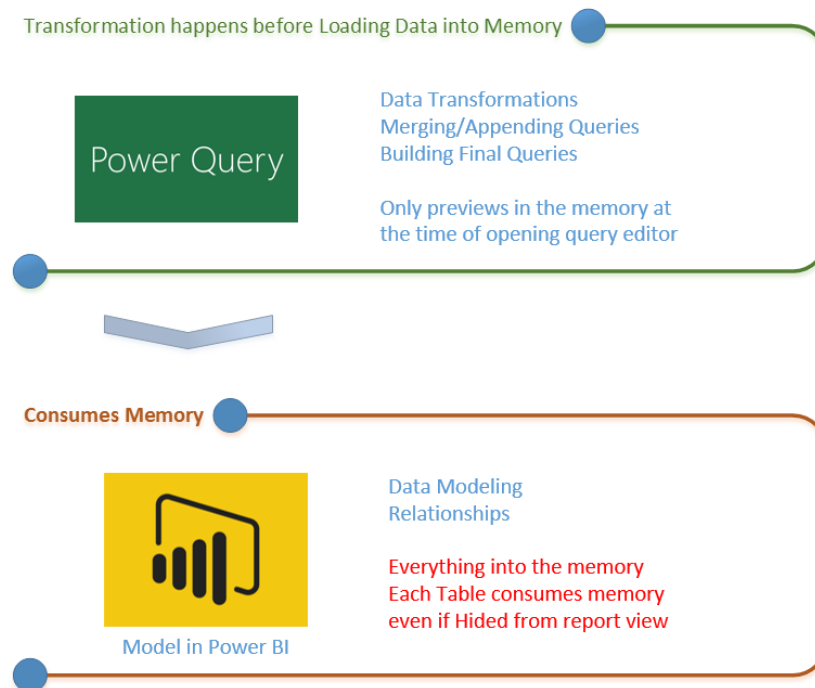
Duplica crea una nuova copia con tutti i passaggi esistenti. La nuova copia verrà isolata dalla query originale. È possibile apportare modifiche alla query originale o nuova e NON si influenzeranno a vicenda. Il riferimento, d'altra parte, è una nuova copia con un solo passaggio: ottenere dati dalla query originale. Se si apporta una modifica alla query originale, la nuova query avrà un impatto. Per esempio; Se si rimuove una colonna dalla query originale, la nuova query non la avrà se ha utilizzato il metodo di riferimento per la copia.

Limitazione del riferimento

Non è possibile utilizzare le query referenziate in tutte le situazioni. Immaginiamo che si abbia una query 1 e che venga creato un riferimento a quella in una query 2. Non è possibile utilizzare il risultato della query 2 nella query 1! **Creerà un riferimento circolare. Stai combinando una query con un riferimento alla query stessa, è impossibile!**

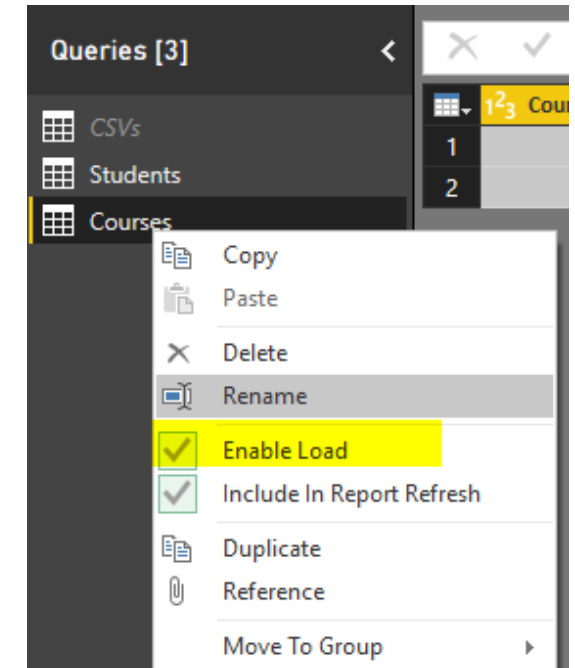
Funzionalità di Enabled Load

- Per impostazione predefinita, quando si sceglie di chiudere e applicare le trasformazioni dall'editor di query, tutte le query verranno caricate nel modello, indipendentemente dal fatto che si desideri utilizzarle nel modello finale o meno.
- E' possibile disabilitare il caricamento di alcune query, in particolare le query utilizzate come trasformazione intermedia per produrre la query finale per il modello. Questo è un'operazione da fare quando il tuo modello diventa grande.



Funzionalità di Enabled Load

- L'approccio migliore è disabilitare il caricamento prima di chiudere l'editor delle query cliccando con il tasto destro sulla query e scegliendo «Enable Load» come si vede nell'immagine laterale
- Disabilitare il caricamento non significa che la query non verrà aggiornata, significa solo che la query non verrà caricata in memoria.
- Quando si fa clic su Aggiorna modello in Power BI o quando si verifica un aggiornamento pianificato, anche le query contrassegnate come Disabilitate verranno aggiornate, ma i loro dati verranno utilizzati come origine intermedia per altre query invece di caricarsi direttamente nel modello.



Utilizzo impostazioni internazionali in Power BI

Le Impostazioni internazionali / Impostazioni locali in Excel e Power BI Desktop Power Query definiscono come

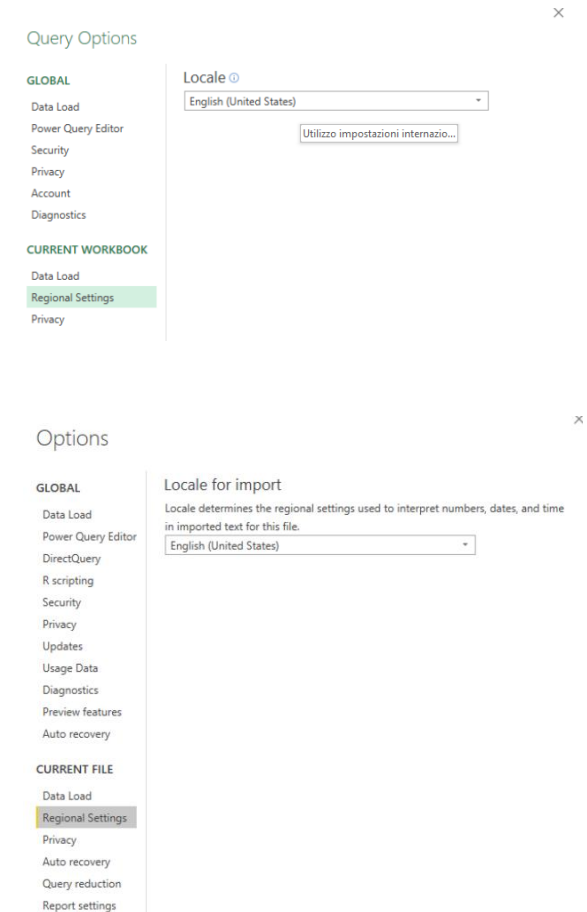
Date, orari e numeri sono interpretati nei file di testo.

- Nel sistema operativo Windows le impostazioni internazionali e internazionali nel Pannello di controllo determinano i numeri, Valuta, orari, date vengono interpretati sul tuo computer
- Power Query utilizza le impostazioni internazionali e locali del sistema operativo Windows per interpretare i dati.
- Il problema sorge se si ottengono file di testo da un paese diverso che utilizza un formato diverso per Date,
- Tempi e numeri

Utilizzo impostazioni internazionali in Power BI

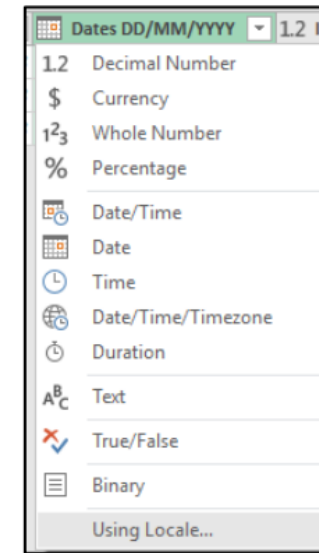
Due modi per risolvere il problema:

1. Impostazioni Regional (valido per tutti i file di testo importati)
 - Power Query utilizza le impostazioni internazionali e locali del sistema operativo Windows per interpretare i dati
 - È possibile modificare le Impostazioni principali in Power Query Editor in modo che tutti i file di testo importati vengano tutti interpretati per l'impostazione regionale specificata.
 - Opzioni query, quindi nella finestra di dialogo Opzioni query fai clic su Impostazioni principali, come mostrato qui a lato. (versione Excel e Power BI)



Utilizzo impostazioni internazionali in Power BI

2. Impostazione "Data Type using locale ..." per singole colonne (stesso processo in Excel e Power BI Desktop)
 - È possibile modificare le impostazioni internazionali per una colonna specifica utilizzando l'icona Tipo di dati nella parte superiore di colonna e quindi facendo clic sull'opzione "Using Locale..." e quindi selezionando le impostazioni internazionali dalla finestra di dialogo, come mostrato qui a lato



Change Type with Locale

Change the data type and select the locale of origin.

Data Type
Date

Locale
French (France)

i Sample input values:

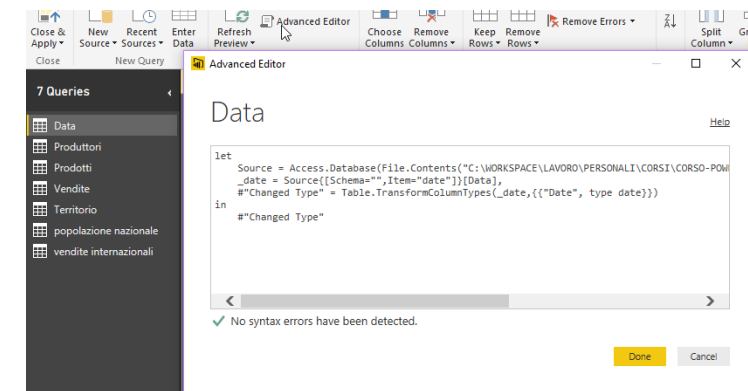
29/03/2016
mardi 29 mars 2016
29 mars
mars 2016

OK

Cancel

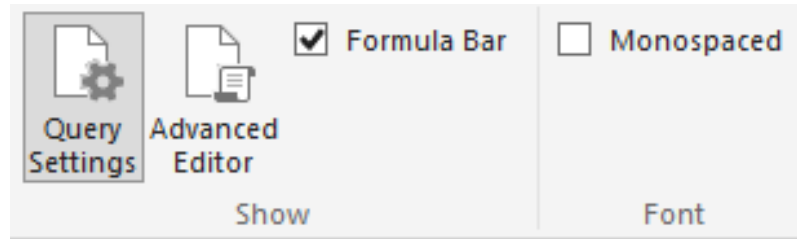
Linguaggio M: Tutte le query sono in M

- Quello che abbiamo visto prima realizzato da UI viene scritto in una query "M"
- Possiamo vedere le query cliccando su "Advanced Editor" da Home, o dalla tab View
- La query include le connessioni e il tipo di connessione
- Tutte le trasformazioni applicate nel Query Editor sono aggiunte nel query code
- La lista degli step si riflette sulla query con lo stesso ordine
- Si può modificare la query e controllarla



Linguaggio M: Che cosa è M?

- Linguaggio funzionale con F#
 - Non ci sono costrutti come loop ma oggetti come le lambda functions
 - E' un linguaggio funzionale non come il linguaggio di Excel o il VBA
 - Capire M serve per scrivere query con calcoli e trasformazioni complesse
- Flessibile
 - Può essere invocato dinamicamente con Evaluate
 - I target sono gli information workers e i data analysts (intermedi e avanzati)
 - Familiare e facile da ricordare
 - Facile da leggere e da scrivere; ha sintassi limitata e usa simboli non standard
 - E' un set più grande di quello fornito da Excel formula language
 - Potenti capacità per utenti avanzati



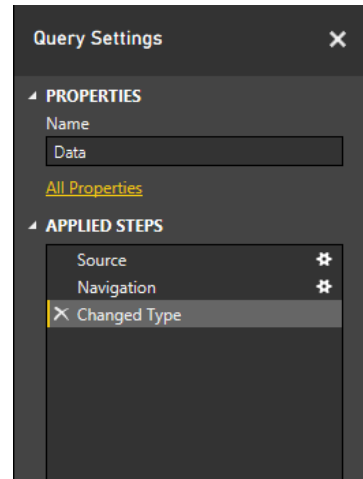
Linguaggio M: Che cosa è M?

Guida di riferimento per le sorgenti di codice M:

- Specifiche di Microsoft Formula Query per Excel Formula (cerca su Google e scarica)
- In una query vuota digitare: `= #shared` per ottenere un elenco di informazioni sul codice M di Power Query.
 - ** Se si convertono le informazioni ottenute con `= #shared` in una tabella, è possibile filtrare per trovare l'argomento desiderato.
- Cerca su Google e vai al sito Microsoft, come: <https://msdn.microsoft.com/en-us/query-bi/m/power-query-m-reference>
- Per informazioni specifiche su una funzione, digita un segno di uguale e poi il nome della funzione nella formula bar e otterrai assistenza su quella particolare funzione

Linguaggio M: Flusso del linguaggio: Let/in

- Valutazione del flusso passo dopo passo
 - Il risultato di uno step è tipicamente utilizzato nello step successivo
 - Uno step può essere definito e riutilizzato
 - Uno step non viene valutato se non è utilizzato
- L'espressione **LET** contiene una serie di espressioni nominate che sono referenziate da un'altra espressione nominata
 - Queste espressioni diventano gli step della query
 - Ogni query creata dall'interfaccia grafica consiste in una singola espressione
- L'espressione **LET** termina con **in**, dove è calcolato l'ultimo valore dell'espressione
 - Questo è l'output della query
 - Può referenziare qualsiasi espressione



```
let
    Source = Sql.Database(".", "AdventureWorksDW2012"),
    Sales =
        Table.FirstN(Source[[Schema="dbo",Item="Sales"]][Data],10),
    Rank = List.Sort(Sales[SalesAmount],Order.Ascending),
in
    Rank
```

Linguaggio M: Tipi di funzioni

- Ha una sintassi come una "lambda expression"
- Una funzione è un valore che, quando è invocata con dei parametri, produce un nuovo valore
- Le funzioni sono scritte con una lista di parametri all'interno di due () seguito dal simbolo =>, che precede una espressione che definisce una funzione

```
(x) => (x*x) //Multiply the value x to x
(x,y) => (x*y) //Multiply the value x to y

[
  Mol = (x,y) => x*y
]
```

Demo 3: vendite per cap

Utilizzo delle funzioni

Gli script già pronti di M


- Esempio di calendario: 3-Script_DateDimension.m
- Esempio di calendaio fiscale: 3-Script_DateDimensionNOFiscal.m

Error handling

- Errore a livello di riga

= Table.RemoveRowsWithErrors("#Changed Type", {"Età"})

Nome	Cognome	Età
Marco	Pozzan	39
Mario	Rossi	NA




Nome	Cognome	Età
Marco	Pozzan	39

- Errore a livello di riga con dettaglio di riga

= Table.ExpandRecordColumn("#Inserted Eta_Try", "Eta_Try", {"HasError", "Value", "Error"}, {"Eta_Try.HasError", "Eta_Try.Value", "Eta_Try.Error"}),

	Nome	Cognome	Età
	Marco	Pozzan	39
	Mario	Rossi	NA



	Nome	Cognome	Età	Eta_Try.HasError	Eta_Try.Value	Eta_Try.Error.Reason	Eta_Try.Error.Message	Eta_Try.Error.Detail
1	Marco	Pozzan	39	FALSE	39	null	null	null
2	Mario	Rossi	Error	TRUE	null	DataFormat.Error	We couldn't convert to Number.	NA

Error handling

- Gestione diretta con Try expression (think: "catch")

- Prova la valutazione
- Codifica i risultati e gli errori con un record
- Opzionalmente la clausola otherwise



```
try error "Bad"  
// [ HasError = true,  
//   Error = [  
//     Reason = "Expression.Error",  
//     Message = "Bad",  
//     Detail = null  
//   ]  
// ]
```

- =Table.AddColumn(Source, "Eta_Bis", each try Number.From([Età]) otherwise 0)



```
try error "Bad" otherwise 42 // 42
```

Nome	Cognome	Età
Marco	Pozzan	39
Mario	Rossi	NA



Nome	Cognome	Età	Età_bis
Marco	Pozzan	39	39
Mario	Rossi	NA	0

Demo 4: gestione degli errori

Gestione degli errori

Conclusioni

- Il linguaggio "M" offre molto di più della UI
- Power Query è il tool di ETL per Power BI
- Power Query può essere gestito dall'IT
- Certificare le query
- Consumare le viste
- E' un prodotto destinato a crescere di importanza

Laboratorio che consolida le basi di Power Query e aggiunge trasformazioni che ancora non abbiamo visto **(da svolgere in autonomia)**