

Preparing Data for Cost Estimating

Data consolidation techniques in Excel

Michael Finnegan
April 9, 2025

WARNING

- The information provided in this presentation is strictly a viewpoint of Mike Finnegan and is not meant to reflect the views of ICEAA or any outside entity.
- Mike Finnegan has spent 11 years in Cost Estimating after completing his Master's in Mathematics. He has created this presentation based purely on anecdotal data to share his various lessons learned and **open a discussion on the use of simple tools to cultivate depth of knowledge for both new and experienced cost estimators while performing day-to-day functions in the cost estimating field.**
- The goal is not to say one way or another is the more correct way of doing things.
- This is not a sales pitch.

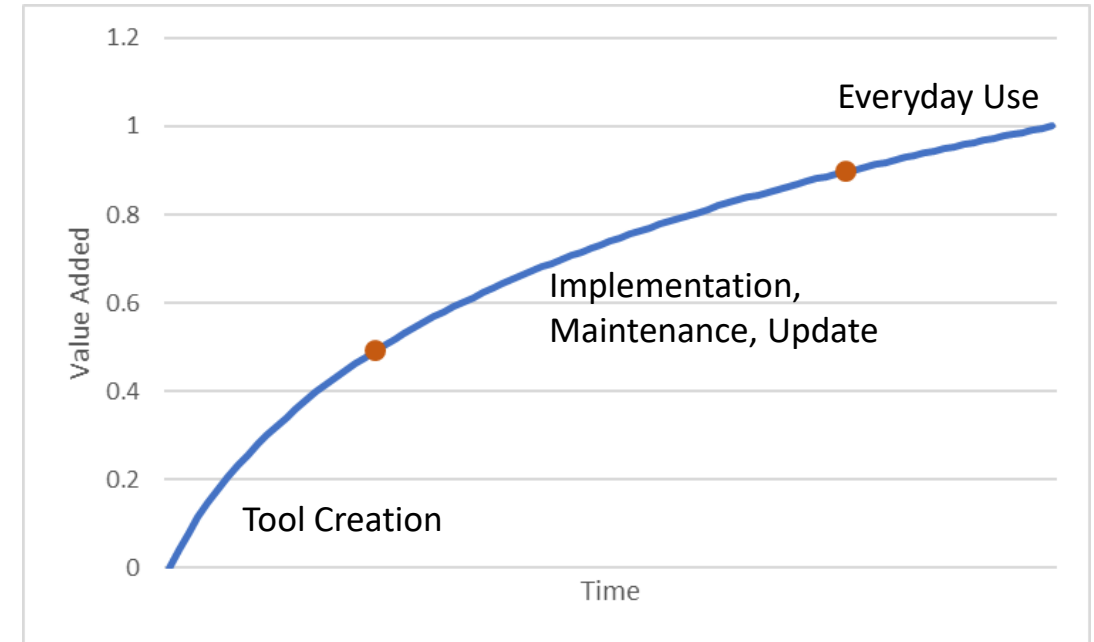
What I use Excel for

- Data Mapping
 - Add skill mix, CAM, WBS, any other category that is not included in the canned report
 - Separate data to various buckets for CERs
- Database / Normalization
 - Technical, Schedule, Cost data, Charge line to WBS maps, Escalation, etc.
- BOE and Pricing tools
 - BOE template that automates spread of input hours / cost
 - Power Query to build pricing with Wrap Rates or Detail Cost Breakdown (example to follow)
- (Parametric) Cost Models
 - Multivariate regression not included ☹️
- Cost Comparison / Test of Reasonableness Charts
- CDRL Automation
- @Risk / Basic Stats



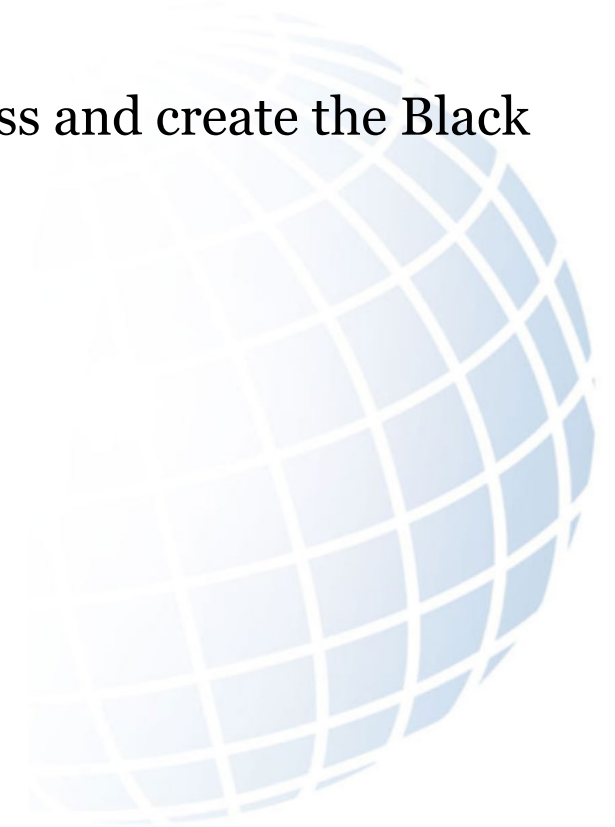
Benefits of Tools/Analysis in Excel (In my opinion)

- Encourages “womb-to-tomb” knowledge
 - “You can only estimate what you can explain”
 - Step-by-step processes instead of Input/Output
 - **WHY is the answer correct**
- Allows for Transparency/Flexibility in tools
 - Troubleshooting
 - Some tools are meant to be broken
- Templates can easily be replicated for other uses
 - “Grab, Snag, Claim, Rename”
- Create cross-checks for approved black box tools
- Excel includes the Basic Stats / ANOVA tables with simple functions
- Macro “Coding” is easier to learn than most languages – Record feature
- Countless online resources for tips/tricks/walk throughs



Dangers of Excel Tools

- The creator is the sole source of the ins and outs
 - “If I get hit by a bus...” tab
- It’s not pretty
- Macros, if used improperly, and other processes can bog down a process and create the Black Box that we are trying to avoid (see next slide)
- Compliance / Consistency across teams
- Wants vs. Needs
- Tool Creators not being involved in daily processes



Is Excel really too slow?

- Links to External Sources
 - See Macro on Slide 7 / 8
- Vlookups vs. Index Match (or XLOOKUP)
 - Index/Match in place of PowerPivots, Queries
- Remove unnecessary Formats, Conditional Formattings, “blank” cells
- Reduce columns in generated reports
 - For example – I run weekly actuals reports. Assume 100,000 rows x 10 columns. If I only use 7 columns, that’s 300,000 cells I don’t need. I could add 42,857 more rows of data before I reach the same size
- Simplifying Formulas and Macros
 - Reduce the use of “Volatile” functions – TODAY, RAND, INDIRECT, SUMIF ☹
 - After calculating, keep one row of formulas, Paste Special Values the rest of the data (macro in backup)

Macro for External Data Sources

```
Sub BringInActuals()  
' BringInActuals Macro  
  
Dim Wb As Workbook  
  
'Clear old data  
Windows("WORKBOOK.xlsx").Activate  
Sheets("DESTINATION TAB").Select  
Range("Range").Select  
Range(Selection, Selection.End(xlDown)).Select  
Selection.ClearContents  
Selection.End(xlUp).Select  
Selection.End(xlToLeft).Select  
  
'Open Source Data; Copy Data  
Set Wb = Workbooks.Open("\FILE PATH\SOURCE DATA.xlsx", False, True)  
Wb.Activate  
"Top Left of data table  
Range("A1").Select  
Range(Selection, Selection.End(xlToRight)).Select  
Range(Selection, Selection.End(xlDown)).Select  
Selection.Copy  
  
'Paste Data  
Windows(" WORKBOOK.xlsx").Activate  
Sheets(" DESTINATION TAB ").Select  
"Top Left cell of Destination Table  
Range("A1").Select  
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone,  
SkipBlanks:=False, Transpose:=False  
  
'Close Source Data  
Windows("Actuals Report Program A.xlsx").Activate  
Application.CutCopyMode = False  
ActiveWindow.Close  
  
End Sub
```

Uses:

- Stack actuals from multiple programs to avoid Queries, Pivots
- Bring in data tables from external sources to keep links within workbook
- Escalation Indices, Chargeline to WBS, CLIN, Roll-ups

Pros:

- Removes need to lookup from external workbook in formulas
- Reduces human error
- Simple to update if Source or Destination workbooks change (use your macro notes!)
- Able to stack data from multiple sources with additional code
- Only need partial file names (see next slide)

Macro for More External Data Sources

- Macro Code for files with changing names:

```
Sub VaryingFileName()  
'BringInActuals Macro  
Dim Wb As String  
'Open Source Data; Copy Data  
Wb = Dir("\\Sw\els\data\ces_library\USERS\Finnegan, Mike\zzClosetUndertheStairs\ICEAA Stuff\Actuals Report" & "*.xlsx")  
If Wb <> "" Then  
    Workbooks.Open Filename:="\\Sw\els\data\ces_library\USERS\Finnegan, Mike\zzClosetUndertheStairs\ICEAA Stuff\" & Wb  
End If  
End Sub
```

- Macro for multiple files within a workbook:

```
'Paste Data in Destination Tab  
Windows("ICEAA SoCal Excel Tips.xlsm").Activate  
Sheets("Actuals Report").Select  
'Find Top left cell below previous table  
Range("A1").Select  
Selection.End(xlDown).Select  
ActiveCell.Offset(1).Select  
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks:=False, Transpose:=False
```



Macro for Copy/Paste Formulas

```
Sub CopyPasteFormulas()  
'Dim LastRow As Long  
'Find total Range needed  
Range("A1").Select  
Range(Selection, Selection.End(xlDown)).Select  
LastRow = Selection.Rows.Count  
'Paste "LastRow" value in N1 to spot-check  
Range("N1").Value = (LastRow)  
'Copy Formulas  
Range("G2:L2").Select  
Selection.Copy  
  
'Select the range to paste the formulas, Calculate to refresh formulas  
Range("G3:L" & LastRow).Select  
ActiveSheet.Paste  
Application.CutCopyMode = False  
Calculate  
  
'Copy formulas, paste values  
Selection.Copy  
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _  
:=False, Transpose:=False  
End Sub
```

- Reduce workbook run-time with minimizing the amount of active formulas



The Basics

Functions to learn

- Index/Match
- SUMIFS
- “Copy/Paste” macros
- Offsets
- Power Query



Pricing Model in Excel

See Example in Workbook



Conclusion / Discussion

- Incorporate the use of “black box” tools when necessary
- Use “transparent” tools for teaching, training, expanding skillsets
- Allow for the opportunity to pass off tools to other team members



Back Up



BASIC FUNCTIONS



Index / Match

- =INDEX(*Array*, *Row Number*, *Column Number*)

- Example:

| | A | B | C |
|---|--------|----------|----------|
| 1 | Colors | Column 2 | Column 3 |
| 2 | Row 2 | Red | Blue |
| 3 | Row 3 | Green | Yellow |

- =INDEX($\$A\$1:\$C\$3,2,3$) will return “Blue”
- Using \$ before the Column Letter or Row Number “anchors” the array if you drag/drop the formula
- Anything inside of an Excel formula can also be a formula!
- =MATCH(*Lookup Value*, *Match Array*, *Match Type*)

- Example

| | D |
|---|------|
| 4 | Year |
| 5 | 2021 |
| 6 | 2022 |
| 7 | 2023 |
| 8 | 2024 |

- =MATCH($2023,\$D\$4:\$D\$8,0$) will return 4, since 2023 is the fourth row down in the Array
 - I’ve only ever used 0 for the Match Type
 - Watch for value types: General, Text, Number, Etc. – Text needs Quotes around Lookup Value

Index / Match

- Embedding MATCH formulas within INDEX:

| | B | C |
|---|-----------|----------|
| 3 | Wrap Rate | 2015 |
| 4 | Cat_1 | \$122.33 |
| 5 | Cat_2 | \$131.65 |
| 6 | Cat_3 | \$137.04 |
| 7 | Cat_4 | \$143.91 |
| 8 | Cat_5 | \$146.73 |

- =INDEX(\$C\$3:\$C\$8,MATCH("Cat_4",\$B\$3:\$B\$8,0),1)
 - \$C\$3:\$C\$8 is the Array (return values, output) (this time using just one column instead of a table)
 - MATCH("Cat_4",\$B\$3:\$B\$8,0) is the Row Number of the INDEX formula
 - 1 is the Column Number of the INDEX formula; very typical to just have 1 here, depending on strategy
- MATCH formula returns 5, so the INDEX Formula is =INDEX(\$C\$3:\$C\$8,5,1)
 - 1st column of the C array is just column C, 5th row of the C array is \$143.91

Index / Match

- Embedding MATCH formulas within INDEX:

| | B | C | D | E | F | G | H |
|---|-----------|----------|----------|----------|----------|----------|----------|
| 3 | Wrap Rate | 2015 | 2016 | 2017 | 2018 | 2019 | 2020 |
| 4 | Cat_1 | \$122.33 | \$125.20 | \$128.05 | \$130.57 | \$135.33 | \$140.05 |
| 5 | Cat_2 | \$131.65 | \$135.56 | \$137.38 | \$141.24 | \$142.67 | \$146.29 |
| 6 | Cat_3 | \$137.04 | \$143.03 | \$148.37 | \$151.44 | \$153.65 | \$159.48 |
| 7 | Cat_4 | \$143.91 | \$148.22 | \$150.50 | \$157.10 | \$162.47 | \$169.75 |
| 8 | Cat_5 | \$146.73 | \$149.39 | \$154.61 | \$158.58 | \$165.37 | \$171.35 |

- =INDEX(\$C\$3:\$H\$8,MATCH("Cat_2",\$B\$3:\$B\$8,0),MATCH(2018,\$B\$3:\$H\$3,0))
 - \$C\$3:\$H\$8 is the Array (return values, output) (this time using a table instead of just one column)
 - MATCH("Cat_2",\$B\$3:\$B\$8,0) is the Row Number of the INDEX formula – Returns 3
 - MATCH(2018,\$B\$3:\$H\$3,0) is the Column Number of the INDEX formula– Returns 5
 - =INDEX(\$C\$3:\$H\$8,3,5) = \$141.24

Index / Match – Escalation Example pt 1

Problem Statement – Just completed an estimate for a new Kart. I need to show cost comparison to prior Karts as a test of reasonableness.

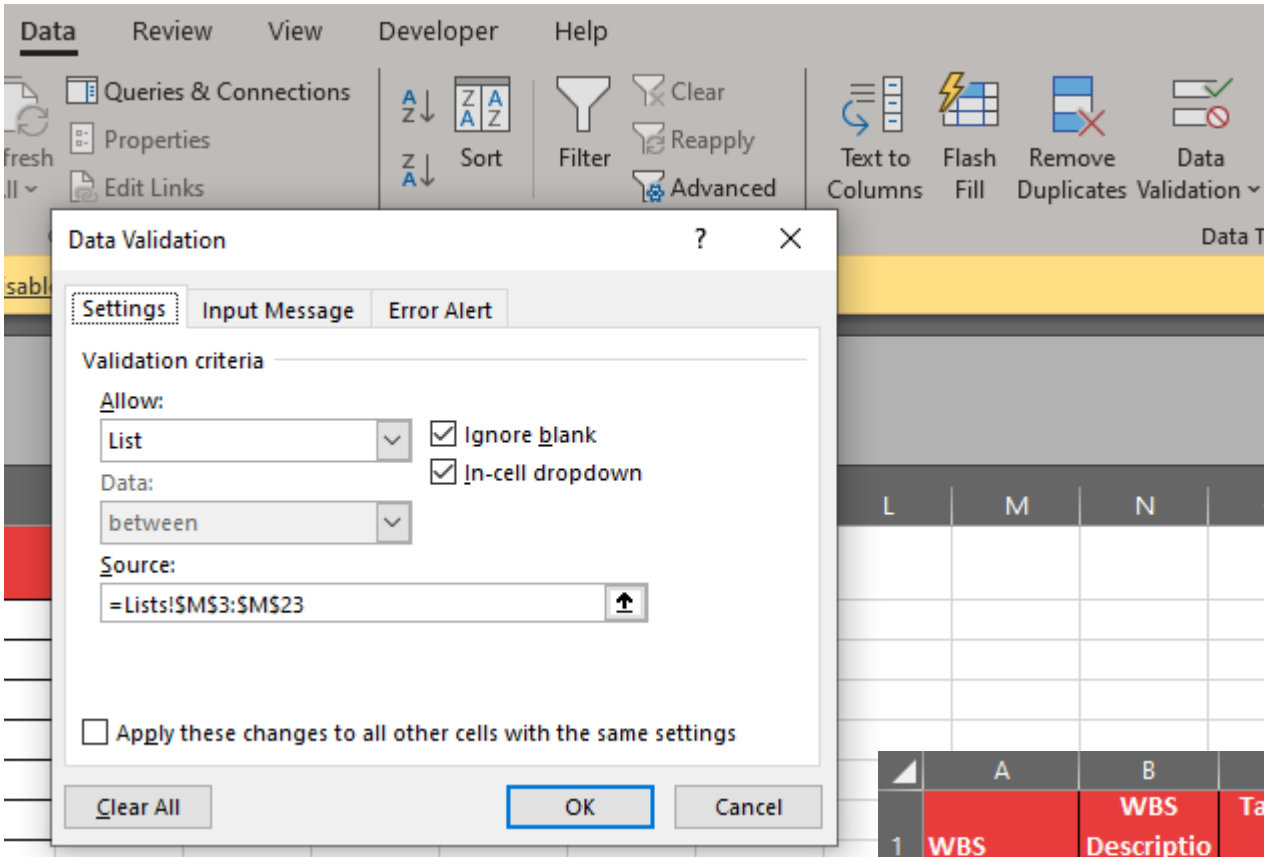
- Pull in Cost History to Column D through...?
- Create a dropdown list in Cell D1
- Create “Unique ID” Column in Program Cost History to pull in data. Each Program Name and WBS ID will create a unique pairing.

| | C | D | E | F | G |
|---|----------|-----|-----------------|--------|-----------|
| 1 | Program | WBS | WBS Description | Cost | Unique ID |
| 2 | Red Fire | A | Vehicle | 206.12 | =C2&D2 |

- Use an Index/Match function to pull in data from Program Cost History tab to Cost Comparison tab
- =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A2,'Program Cost History'!\$G:\$G,0),1)
- The MATCH function is combining the Program Title from D1 and the WBS number from Column A, and look up the corresponding ROW from the new column we just created on the Program Cost History tab
- Copy formula from Cell D2 to D3 through D7. Note the \$'s in the formula to lock in Row 1 and Column A, note how the formula changes when we drag it down.

| | A | B | C |
|---|-------|-------------|--------------|
| | | WBS | Tanooki Kart |
| 1 | WBS | Description | Estimate |
| 2 | A | Vehicle | \$173.39 |
| 3 | A.1 | Tires | \$39.20 |
| 4 | A.2 | Kart Body | \$122.25 |
| 5 | A.2.a | insides | \$66.85 |
| 6 | A.2.b | outsides | \$55.40 |
| 7 | A.3 | Glider | \$11.94 |

Photos of Steps 😊



| | A | B | C | D |
|---|-------|------------|---------|--|
| | | WBS | Tanooki | DK Jumbo |
| 1 | WBS | Descriptio | Kart | |
| 2 | A | Vehicle | 171.22 | =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A2,'Program Cost History'!\$G:\$G,0),1) |
| 3 | A.1 | Tires | 38.59 | =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A3,'Program Cost History'!\$G:\$G,0),1) |
| 4 | A.2 | Kart Body | 103.81 | =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A4,'Program Cost History'!\$G:\$G,0),1) |
| 5 | A.2.a | insides | 57.87 | =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A5,'Program Cost History'!\$G:\$G,0),1) |
| 6 | A.2.b | outsides | 45.94 | =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A6,'Program Cost History'!\$G:\$G,0),1) |
| 7 | A.3 | Glider | 28.82 | =INDEX('Program Cost History'!\$F:\$F,MATCH(D\$1&\$A7,'Program Cost History'!\$G:\$G,0),1) |

Index / Match – Escalation Example pt 2

Problem Statement – Pulled in Cost History – but is it normalized to our estimate year?

- Luckily, we have schedule data ready to go in our Program History Database
- Pull in the DK Jumbo Escalation From Year value of 2011 and the corresponding Escalation Rate of 131%
 - Option 1
 - Add an Escalation column to the Schedule Data
 =INDEX(Escalation!\$G\$5:\$G\$41,MATCH(\$G6,Escalation!\$B\$5:\$B\$41,0),1)
 - INDEX/MATCH on the Cost Comparison tab OR the Program Cost History tab to multiple the program costs by the Escalation %
 - Option 2
 - Imbed a INDEX/MATCH within an INDEX/MATCH
 =INDEX(Escalation!\$G\$5:\$G\$41,MATCH(INDEX(Schedule!\$G\$5:\$G\$25,1),MATCH(\$C2,Schedule!\$B\$5:\$B\$25,0),1),Escalation!\$B\$5:\$B\$41,0),1)

| | B | C | D | F | G |
|----|---------------|------------|------------|------------|---------------|
| 5 | Program | Start Date | End Date | Midpoint | Esc From Year |
| 6 | Red Fire | 8/27/2005 | 7/2/2008 | 1/29/2007 | 2007 |
| 7 | Heart Coach | 10/7/2005 | 2/17/2006 | 12/12/2005 | 2005 |
| 8 | Goo-Goo Buggy | 1/12/2008 | 12/11/2013 | 12/27/2010 | 2010 |
| 9 | Koopa Dasher | 1/27/2008 | 1/10/2011 | 7/19/2009 | 2009 |
| 10 | DK Jumbo | 6/23/2008 | 1/8/2014 | 4/1/2011 | 2011 |

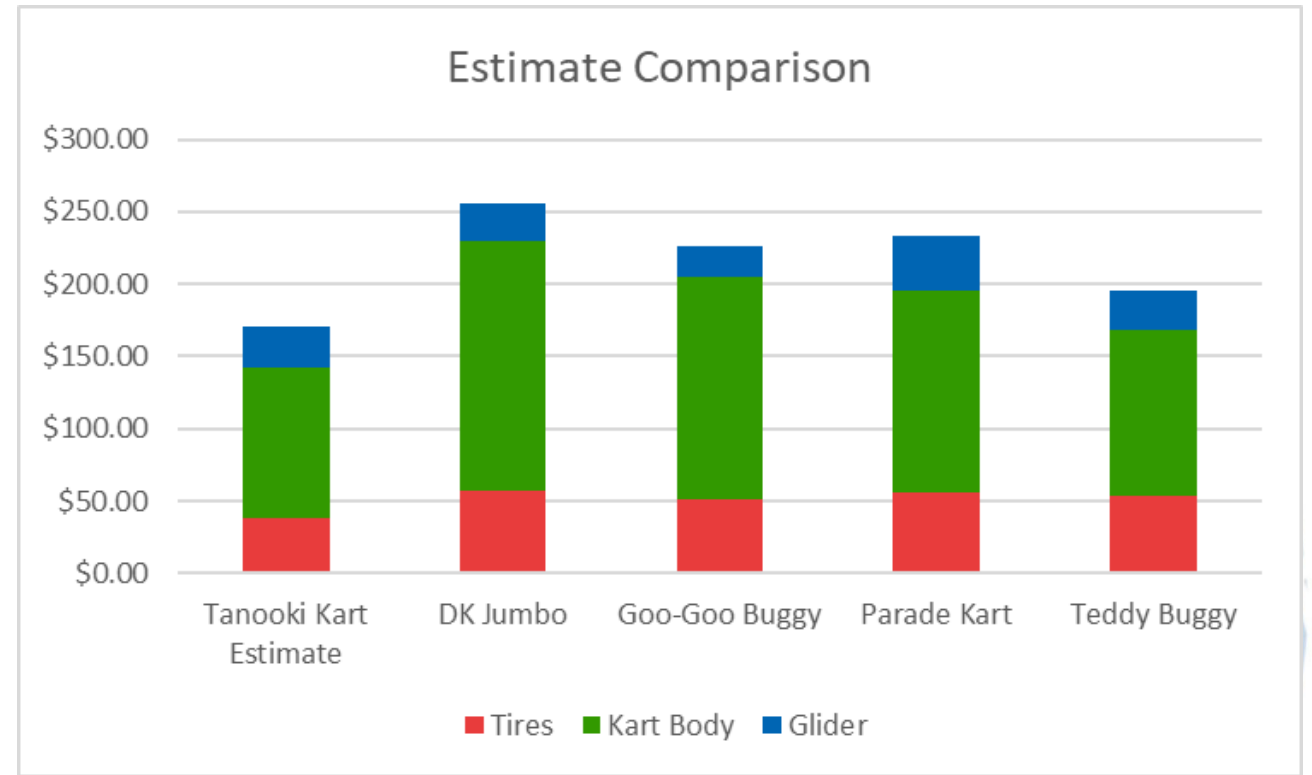
| | B | C | D | F | G | H |
|----|---------------|------------|------------|------------|---------------|-------|
| 5 | Program | Start Date | End Date | Midpoint | Esc From Year | Esc % |
| 6 | Red Fire | 8/27/2005 | 7/2/2008 | 1/29/2007 | 2007 | 145% |
| 7 | Heart Coach | 10/7/2005 | 2/17/2006 | 12/12/2005 | 2005 | 150% |
| 8 | Goo-Goo Buggy | 1/12/2008 | 12/11/2013 | 12/27/2010 | 2010 | 135% |
| 9 | Koopa Dasher | 1/27/2008 | 1/10/2011 | 7/19/2009 | 2009 | 141% |
| 10 | DK Jumbo | 6/23/2008 | 1/8/2014 | 4/1/2011 | 2011 | 131% |

| | C | D | E | F | G | H |
|----|----------|-------|-----------------|--------|---------------|---------------------|
| 1 | Program | WBS | WBS Description | Cost | Unique ID | 2025 Escalated Cost |
| 26 | DK Jumbo | A | Vehicle | 195.12 | DK JumboA | 255.81 |
| 27 | DK Jumbo | A.1 | Tires | 43.82 | DK JumboA.1 | 57.45 |
| 28 | DK Jumbo | A.2 | Kart Body | 131.53 | DK JumboA.2 | 172.44 |
| 29 | DK Jumbo | A.2.a | insides | 72.70 | DK JumboA.2.a | 95.31 |
| 30 | DK Jumbo | A.2.b | outsides | 58.83 | DK JumboA.2.b | 77.13 |
| 31 | DK Jumbo | A.3 | Glider | 19.77 | DK JumboA.3 | 25.92 |

Index / Match – Escalation Example Bonus

Creating a Chart

- Highlight all data
- On the Insert tab, select Recommended Charts
- On the All Chart tab, select the Stacked Columns
- Remove Vehicle from the selection



XLOOKUP

- =XLOOKUP(*Lookup_Value*, *Lookup_Array*, *Return_Array*, [*If_Not_Found*])

• Example:

| | A | B | C |
|---|---------------------------------------|----------|----------|
| 1 | Colors | Column 2 | Column 3 |
| 2 | Row 2 | Red | Blue |
| 3 | Row 3 | Green | Yellow |
| 4 | =XLOOKUP("RED", \$B1:\$B3, \$C1:\$C3) | | |

- =XLOOKUP("Red", \$B\$1:\$B\$3, \$C\$1:\$C\$3) will return "Blue"
- It is not required to use [*If_Not_Found*] but it can be useful to avoid the ugly #N/A errors
- =XLOOKUP("Purple", \$B\$1:\$B\$3, \$C\$1:\$C\$3) will return "#N/A" since Purple is not in Column B
- =XLOOKUP("Purple", \$B\$1:\$B\$3, \$C\$1:\$C\$3, "Not on List") will return "Not on List"
- The same Cost Comparison Table can be created with the XLOOKUP:

| | A | B | C | D | E | F | G | H | I | J | K |
|---|-----|-------------|----------|---|---------|--------|-------|---|---|---|---|
| | | WBS | Tanooki | | Goo-Goo | Parade | Teddy | | | | |
| 1 | WBS | Description | Kart | DK Jumbo | Buggy | Kart | Buggy | | | | |
| 2 | A | Vehicle | \$171.22 | =XLOOKUP(D\$1&\$A2, 'Program Cost History'!\$G:\$G, 'Program Cost History'!\$H:\$H) | | | | | | | |

- XLOOKUPS can also look up across rows, and be stacked to be more efficient

SUMIFS

- 2 types of SUMIFs:
 - =SUMIF(*Range*, *Criteria*, [*SUM RANGE*])
 - =SUMIFS([*SUM RANGE*], *Criteria Range 1*, *Criteria 1*, *Criteria Range 2*, *Criteria 2*, ...)
- Even when I only have 1 criteria, I default to a SUMIFS. That was I can add more criteria later without changing the format of the function
- Example – Sum the Number of Copies of Mario World sold
 - =SUMIFS(\$C\$2:\$C\$10,\$B\$2:\$B\$10,"Mario World") = 95
 - In the example to the right, we can also let the Criteria 1 be linked to a cell, in this case \$E2, to get the same result.
 - The formula can then be dragged/dropped to F3 and F4.

| | A | B | C |
|----|--------|-------------|-------------|
| 1 | Month | Game | Copies Sold |
| 2 | Oct-93 | Mario World | 38 |
| 3 | Oct-93 | Mario Kart | 37 |
| 4 | Oct-93 | Mario Paint | 21 |
| 5 | Nov-93 | Mario World | 22 |
| 6 | Nov-93 | Mario Kart | 21 |
| 7 | Nov-93 | Mario Paint | 33 |
| 8 | Dec-93 | Mario World | 35 |
| 9 | Dec-93 | Mario Kart | 38 |
| 10 | Dec-93 | Mario Paint | 25 |

| | A | B | C | D | E | F | G | H |
|----|--------|-------------|-------------|---|-------------|---|---|---|
| 1 | Month | Game | Copies Sold | | Game | Copies Sold | | |
| 2 | Oct-93 | Mario World | 38 | | Mario World | =SUMIFS(\$C\$2:\$C\$10,\$B\$2:\$B\$10,\$E2) | | |
| 3 | Oct-93 | Mario Kart | 37 | | Mario Kart | | | |
| 4 | Oct-93 | Mario Paint | 21 | | Mario Paint | | | |
| 5 | Nov-93 | Mario World | 22 | | | | | |
| 6 | Nov-93 | Mario Kart | 21 | | | | | |
| 7 | Nov-93 | Mario Paint | 33 | | | | | |
| 8 | Dec-93 | Mario World | 35 | | | | | |
| 9 | Dec-93 | Mario Kart | 38 | | | | | |
| 10 | Dec-93 | Mario Paint | 25 | | | | | |

SUMIFS

- Example – Sum the Number of Copies of Mario World sold across all stores, by month

- $=\text{SUMIFS}(\$K\$2:\$K\$28, \$J\$2:\$J\$10, \text{"Mario World"}, \$I\$2:\$I\$28, \text{"Oct-93"}) = 95$

- In the example to the right, we can also let the Criteria ties to the Row titles and Column Titles of new table.

| | M | N | O | P | Q | R |
|---|-------------|--|--------|--------|---|---|
| 1 | Game | Oct-93 | Nov-93 | Dec-93 | | |
| 2 | Mario World | $=\text{SUMIFS}(\$K\$2:\$K\$28, \$J\$2:\$J\$28, \$M2, \$I\$2:\$I\$28, N\$1)$ | | | | |
| 3 | Mario Kart | $\text{SUMIFS}(\text{sum_range}, \text{criteria_range1}, \text{criteria1}, [\text{criteria_range}]$ | | | | |
| 4 | Mario Paint | | | | | |

- $=\text{SUMIFS}(\$K\$2:\$K\$28, \$J\$2:\$J\$10, \$M2, \$I\$2:\$I\$28, N\$1) = 95$

- Note that we “tie down” column M in the first Criteria, and the Row 1 in the second Criteria.
- Quickly replicate table to summarize sales by store

| | H | I | J | K |
|----|--------------|--------|-------------|-------------|
| 1 | Location | Month | Game | Copies Sold |
| 2 | West County | Oct-93 | Mario World | 38 |
| 3 | West County | Oct-93 | Mario Kart | 37 |
| 4 | West County | Oct-93 | Mario Paint | 21 |
| 5 | West County | Nov-93 | Mario World | 22 |
| 6 | West County | Nov-93 | Mario Kart | 21 |
| 7 | West County | Nov-93 | Mario Paint | 33 |
| 8 | West County | Dec-93 | Mario World | 35 |
| 9 | West County | Dec-93 | Mario Kart | 38 |
| 10 | West County | Dec-93 | Mario Paint | 25 |
| 11 | Chesterfield | Oct-93 | Mario World | 21 |
| 12 | Chesterfield | Oct-93 | Mario Kart | 40 |
| 13 | Chesterfield | Oct-93 | Mario Paint | 39 |
| 14 | Chesterfield | Nov-93 | Mario World | 39 |
| 15 | Chesterfield | Nov-93 | Mario Kart | 22 |
| 16 | Chesterfield | Nov-93 | Mario Paint | 24 |
| 17 | Chesterfield | Dec-93 | Mario World | 23 |
| 18 | Chesterfield | Dec-93 | Mario Kart | 25 |
| 19 | Chesterfield | Dec-93 | Mario Paint | 33 |
| 20 | Galleria | Oct-93 | Mario World | 36 |
| 21 | Galleria | Oct-93 | Mario Kart | 27 |
| 22 | Galleria | Oct-93 | Mario Paint | 39 |
| 23 | Galleria | Nov-93 | Mario World | 34 |
| 24 | Galleria | Nov-93 | Mario Kart | 20 |
| 25 | Galleria | Nov-93 | Mario Paint | 35 |
| 26 | Galleria | Dec-93 | Mario World | 37 |
| 27 | Galleria | Dec-93 | Mario Kart | 36 |
| 28 | Galleria | Dec-93 | Mario Paint | 33 |

Pivot Tables

Problem Statement: Summarize Hours Incurred by Business Unit and CLIN

- For this example, we have an actuals report by Name and by Charge Code, but missing the CLIN and Bus Unit. We can assume each CLIN contains unique Charge Codes, and each Bus Unit contains unique names
- For this example, there are 4 years of actuals with weekly data, 20 names, 3 CLINs, 10 Charge Codes, and 10 Labor Categories (we may or may not need)
- This data set, 15,360 rows, is still small, but large enough that a Pivot table works better.
- Highlight all Rows and Columns, under “Insert” ribbon, select Pivot, New Worksheet

| | A | B | C | D | E | F |
|----|------|-----------|------|--------|-------------|-------|
| 1 | Year | Month End | Week | Name | Charge Code | Hours |
| 2 | 2021 | 1/28/2021 | 1 | Boo | SPRMROWLD | 7 |
| 3 | 2021 | 1/28/2021 | 1 | Boo | 6GLDNCNS | 9 |
| 4 | 2021 | 1/28/2021 | 1 | Boo | SPRMROKRT | 5 |
| 5 | 2021 | 1/28/2021 | 1 | Boo | 6GLDNCNS | 8 |
| 6 | 2021 | 1/28/2021 | 1 | Bowser | MRIOPRTY | 9 |
| 7 | 2021 | 1/28/2021 | 1 | Bowser | YSHICOOK | 10 |
| 8 | 2021 | 1/28/2021 | 1 | Bowser | MRIOPRTY | 9 |
| 9 | 2021 | 1/28/2021 | 1 | Bowser | MROTENNS | 6 |
| 10 | 2021 | 1/28/2021 | 1 | DK | SPRSMASH | 5 |
| 11 | 2021 | 1/28/2021 | 1 | DK | YSHISLND | 8 |
| 12 | 2021 | 1/28/2021 | 1 | DK | SPRMROWLD | 7 |
| 13 | 2021 | 1/28/2021 | 1 | DK | MRIOPRTY | 12 |
| 14 | 2021 | 1/28/2021 | 1 | Goomba | YSHISLND | 7 |
| 15 | 2021 | 1/28/2021 | 1 | Goomba | MRIOPRTY | 11 |
| 16 | 2021 | 1/28/2021 | 1 | Goomba | SPRMROWLD | 9 |
| 17 | 2021 | 1/28/2021 | 1 | Goomba | MRIOPRTY | 9 |

Pivot Tables

- Summarized by two categories not included in the initial data set
- Can add filters or Slicers to further dig into the data
- Can double-click any of the values to look at the source data lines that feed into the sum
- Can be used to validate data, find #N/A errors, etc.
- Can be refreshed if data source is updated

The screenshot shows an Excel PivotTable with the following data:

| Row Labels | GB | N64 | SNES | Grand Total |
|--------------------|---------------|---------------|---------------|----------------|
| Baddie | 5,399 | 16,779 | 17,211 | 39,389 |
| Hero | 9,843 | 8,286 | 7,895 | 26,024 |
| Pal | 1,779 | 5,333 | 6,001 | 13,113 |
| Villian | 20,878 | 20,396 | 10,763 | 52,037 |
| Grand Total | 37,899 | 50,794 | 41,870 | 130,563 |

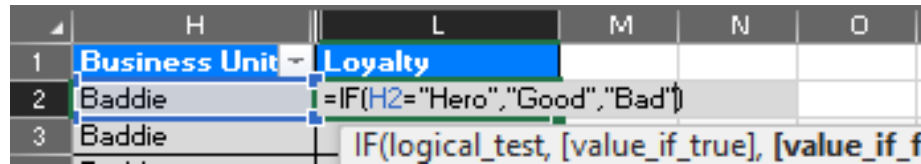
The PivotTable Fields task pane on the right shows the following configuration:

- Choose fields to add to report: Search
- Fields in the list: Year, Month End, Week, Name, Charge Code, Hours (checked), LaborRate, Business Unit (checked), CLIN (checked). More Tables...
- Drag fields between areas below:
- Filters: (empty)
- Columns: CLIN
- Rows: Business Unit
- Values: Sum of Hours

If Statements

- =IF(*Logical_Test*,*[Value_If_True]*,*[Value_If_False]*)
 - Will return different values or perform different operations given the result of the logical test
 - Multiple IF statements can be embedded within *[Value_If_True]* and *[Value_If_False]*
 - Multiple Criteria/Tests can be used within the *Logical_Test* by using AND() or OR() functions

- Example



| | H | L | M | N | O |
|---|---------------|---|---|---|---|
| 1 | Business Unit | Loyalty | | | |
| 2 | Baddie | =IF(H2="Hero","Good","Bad") | | | |
| 3 | Baddie | IF(logical_test, [value_if_true], [value_if_f | | | |

- =IF(H2="Hero", "Good", "Bad") will return "Bad" since clearly H2 = Baddie, not Hero
- We can build on this to include "Ally's" as good by using an OR() function for the logical test
- =IF(OR(H2="Hero",H2="Pal"), "Good", "Bad") will still return "Bad" since clearly H2 = Baddie, not Hero or Pal;

See next example for building in a function:

Side Quest – Going on Dates

Problem Statement: Asked to Sum by Quarters, asked to Sum by FY

- Formulas to Know

- =MONTH(*Serial_Number*) returns a value of 1-12
- =YEAR(*Serial_Number*) returns the Year number (might need to check data type on these)
- =YEARFRAC(*Start_Date*, *End_Date*) returns the number of years between two dates (can be rounded)
- =DAYS(*End_Date*, *Start_Date*) returns the number of days between two dates (also can just subtract)
- Use the & operation that we used in the Unique ID
- =RIGHT(*text*,*[num_chars]*) returns the X number of characters from the right.
 - =RIGHT(2021,2) will return 21

To get the Quarters in a format of QQYY we can use the operation:

=ROUNDUP(MONTH(\$B2)/3,)&"Q"&RIGHT(YEAR(\$B2),2)



The screenshot shows an Excel spreadsheet with columns A through O and rows 1 and 2. Row 1 has headers: A: Year, B: Month End, J: Quarter. Row 2 has data: A: 2021, B: 1/28/2021, J: =ROUNDUP(MONTH(\$B2)/3,)&"Q"&RIGHT(YEAR(\$B2),2). The formula bar shows the formula for cell J2.

| | A | B | J | K | L | M | N | O |
|---|------|-----------|--|---|---|---|---|---|
| 1 | Year | Month End | Quarter | | | | | |
| 2 | 2021 | 1/28/2021 | =ROUNDUP(MONTH(\$B2)/3,)&"Q"&RIGHT(YEAR(\$B2),2) | | | | | |

Similarly, for converting to FY, we can use:

=IF(MONTH(\$B2)>9,"FY"&RIGHT(YEAR(\$B2),2)+1,"FY"&RIGHT(YEAR(\$B2),2)+1)

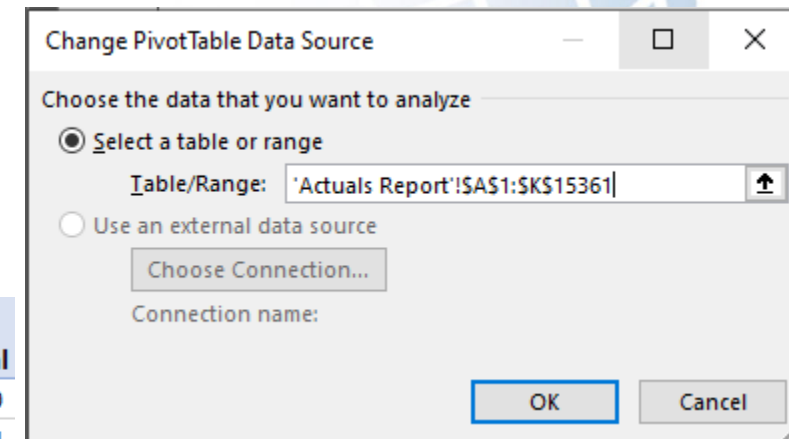
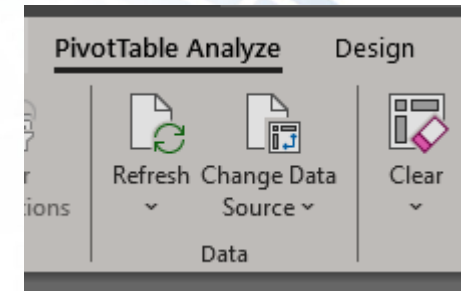
Side Quest – Going on Dates

| | A | B | C | D | E | F | G | H | I | J | K |
|---|------|-----------|-----|--------|------------|------|-----------|---------------|------|---------|------|
| 1 | Yes | Month End | Wee | Name | Charge Cod | Hour | LaborRate | Business Unit | CLIN | Quarter | FY |
| 2 | 2021 | 1/28/2021 | 1 | Boo | SPRMR0WLD | 7 | BD2 | Baddie | SNES | 1Q21 | FY21 |
| 3 | 2021 | 1/28/2021 | 1 | Boo | 6GLDNCNS | 9 | BD2 | Baddie | GB | 1Q21 | FY21 |
| 4 | 2021 | 1/28/2021 | 1 | Boo | SPRMR0KRT | 5 | BD2 | Baddie | SNES | 1Q21 | FY21 |
| 5 | 2021 | 1/28/2021 | 1 | Boo | 6GLDNCNS | 8 | BD2 | Baddie | GB | 1Q21 | FY21 |
| 6 | 2021 | 1/28/2021 | 1 | Bowser | MRIOPRTY | 9 | VL3 | Villian | N64 | 1Q21 | FY21 |
| 7 | 2021 | 1/28/2021 | 1 | Bowser | YSHICOOK | 10 | VL3 | Villian | GB | 1Q21 | FY21 |
| 8 | 2021 | 1/28/2021 | 1 | Bowser | MRIOPRTY | 9 | VL3 | Villian | N64 | 1Q21 | FY21 |
| 9 | 2021 | 1/28/2021 | 1 | Bowser | MROTENNS | 6 | VL3 | Villian | GB | 1Q21 | FY21 |

The Actuals Report now has as much data added to it as the generated report!

To bring the new columns into the Pivot Table, click anywhere in the pivot to bring up the PivotTable Analyze and Design tabs to the ribbon, then click *Change Data Source*

Select the entire range and hit OK. On the Field List, the new Columns should show up as options



| Sum of Hours | FY | | | | | Grand Total |
|--------------------|---------------|---------------|---------------|---------------|--------------|----------------|
| Business Unit | FY21 | FY22 | FY23 | FY24 | FY25 | |
| Baddie | 7,389 | 9,773 | 9,778 | 9,946 | 2,503 | 39,389 |
| Hero | 4,877 | 6,455 | 6,531 | 6,558 | 1,603 | 26,024 |
| Pal | 2,426 | 3,290 | 3,361 | 3,199 | 837 | 13,113 |
| Villian | 9,780 | 12,982 | 12,974 | 13,025 | 3,276 | 52,037 |
| Grand Total | 24,472 | 32,500 | 32,644 | 32,728 | 8,219 | 130,563 |

OFFSET

- =OFFSET(*Reference Cell*, *Rows*, *Columns*)
 - Reference Cell is the starting point
 - Rows moves the selected cell down with Positive Numbers, up with negative numbers
 - Columns moves the selected cell right with positive numbers, left with negative numbers

| | A | B | C | D | E |
|---|----|----|------------|----|----|
| 1 | B | I | N | G | O |
| 2 | 8 | 19 | 31 | 51 | 61 |
| 3 | 12 | 29 | 42 | 53 | 74 |
| 4 | 6 | 16 | Free Space | 46 | 72 |
| 5 | 9 | 21 | 37 | 48 | 63 |
| 6 | 11 | 18 | 39 | 57 | 65 |

- Bingo Card examples!
 - =OFFSET(\$C\$4,-1,1) = 53. We start from Free Space, then move up 1, and right 1.
 - =OFFSET(\$C\$4, 2, -2) =11. Start from Free Space, move down 2, and left 2.
- Uses – Time Phasing, Applying curves, creating dynamic tables with reference lists

Data Type Error Fix

- Issue with Cell Formats not Updating
 - In this issue, the dates were stored as text. After changing the Cell Type to Date, the values wont update without clicking in the cell and hitting “Enter”
 - A “quick fix” is multiplying all cells in the column by the value of 1 in the desired format, in this case 1 is 1/1/1900.
 - Type 1/1/1900 in a random blank cell, and copy the cell.
 - Highlight the entire column, right click and select Paste Special
 - In the menu, select “Multiply”

| | A | B |
|----|------|-----------|
| 1 | Year | Month End |
| 2 | 2021 | 1/28/2021 |
| 3 | 2021 | 1/28/2021 |
| 4 | 2021 | 1/28/2021 |
| 5 | 2021 | 1/28/2021 |
| 6 | 2021 | 1/28/2021 |
| 7 | 2021 | 1/28/2021 |
| 8 | 2021 | 1/28/2021 |
| 9 | 2021 | 1/28/2021 |
| 10 | 2021 | 1/28/2021 |

