

BRETT REDD – NOTABLE CONSIDERATIONS, SKILLS, & PORTFOLIO

NOTABLE CONSIDERATIONS

In an effort to ensure that my skill set is of utmost benefit to my next employer, I have spent an immense amount of time and effort completing additional business intelligence training since October 2020. Therefore, in addition to my 10 year background managing, analyzing, and visualizing data in SQL, Access, Excel, and Tableau, please consider the following:

- Through guided learning coursework, I am now able to leverage all facets of Power BI, including the data transformation editor, and relationship manager. I am also very comfortable computing DAX calculated columns and measures.
- I have taken an additional SQL course to refresh my knowledge, and ensure I leverage the appropriate syntax and methodology when computing aggregated functions, joins, conditional expressions, and creating databases and tables.
- Currently, I am in the process of learning python, and have surpassed all of the beginner level assignments and challenges with a very fluent understanding of the basics, such as assigning variables, strings/integers/floats, creating/calling lists and dictionaries, using for loops, while loops, if/elif/else, and finally defining classes and functions. A full list of skills I possess, as well as snapshots of dashboards I have created can be found below.

PROFESSIONAL ACHIEVEMENTS

TABLEAU

- Developed comprehensive dashboards to support marketing and annual reporting of CBREs project management service line in the Americas.
- Developed operational dashboards to support monitoring of KPIs, cash flow, staffing, closeout, and data management for CBRE client PMO. Required the use of Access/SQL to join data from multiple sources.
- Developed HR dashboards for insight into company demographics and to monitor time to fill requisitions within the project management service line.
- Developed client satisfaction / NPS (net promoter score) survey dashboards to draw insights into areas of success and opportunity.

EXCEL-BASED DATA ANALYSIS ACHIEVEMENTS

- Received "Bravo" Award for developing project management operational dashboards for CBREs largest financial client, using Access, excel, excel VBA macros, etc. This led to being recognized as a best practice to be leveraged on other accounts. Leveraged this methodology when upgrading dashboards to Tableau on other accounts.
- Created automated data management / data integrity reports leveraging Access SQL design manager to join data from multiple sources, and write conditional statements to identify missing data and data discrepancies.

SME – DEVELOPMENT OF KAHUA PROJECT MANAGEMENT SOFTWARE

- Selected to serve as a reporting SME, as well as participate in all application design reviews and UAT testing for project management software rolled out company-wide for CBRE. My primary objective was to ensure appropriate analysis methodologies, and data management capabilities, would be possible in the out-of-the-box offering.

SKILLS

SQL (PostgreSQL, SSMS)

Python3

HTML(5)

CSS

Tableau

PowerBI / PowerQuery / PowerPivot

DAX

Excel

MS Project

Visio

MS Office Suite

Various project management applications (Kahua / Proliance / SKIRE / Expesite, etc.)

Argus

Yardi

PORTFOLIO

Snapshots of dashboard views and visualizations I have created through guided learning coursework, and in my free time can be found below. In addition, interactive Tableau views I have created on Tableau public for “work out Wednesday” challenges can be found here: <https://public.tableau.com/profile/vizzyredds#!/?newProfile=&activeTab=0>

TABLEAU



Brett Redd | Realtor®
Mobile: (409) 450-9822
Email: Brett.Redd@exp Realty.com
Web: BrettRedd.exp Realty.com



Select City: McKinney | Elementary School Name: (All) | Construction Status: (All) | Month, Year of Close Date: (Multiple values) | SF Range: (Multiple values)

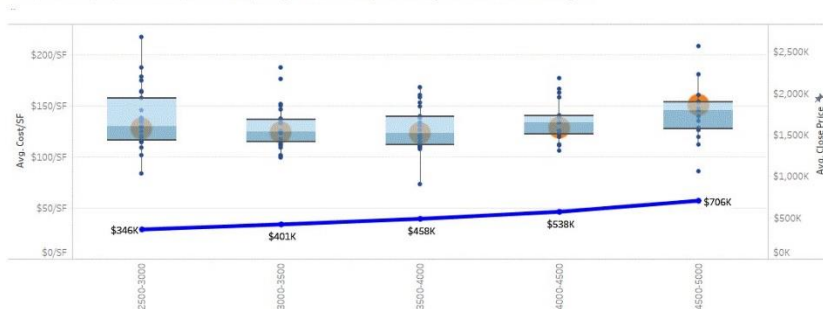
Collin County Average Sale Price by City - McKinney
SF Range(s): 2500-3000, 3000-3500, 3500-4000 and 2 more



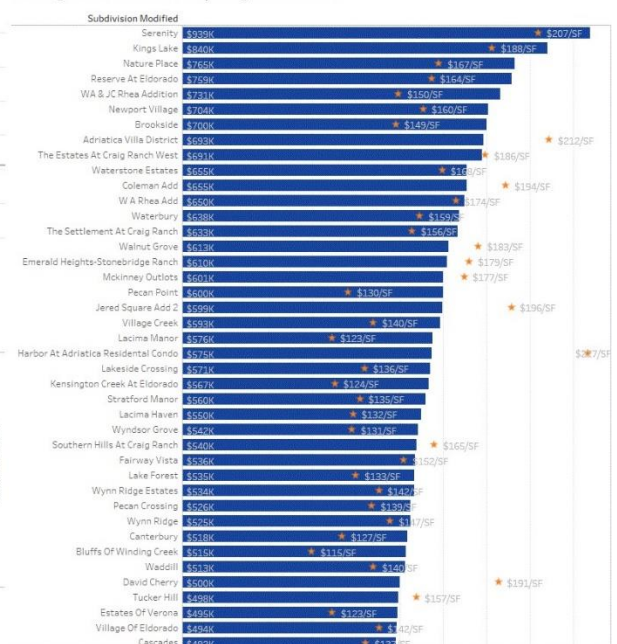
Sequential, vertical marks represent the range of average home prices, by city, within Collin County above. ^^^

The large, orange dot represents where the selected city falls within each range.

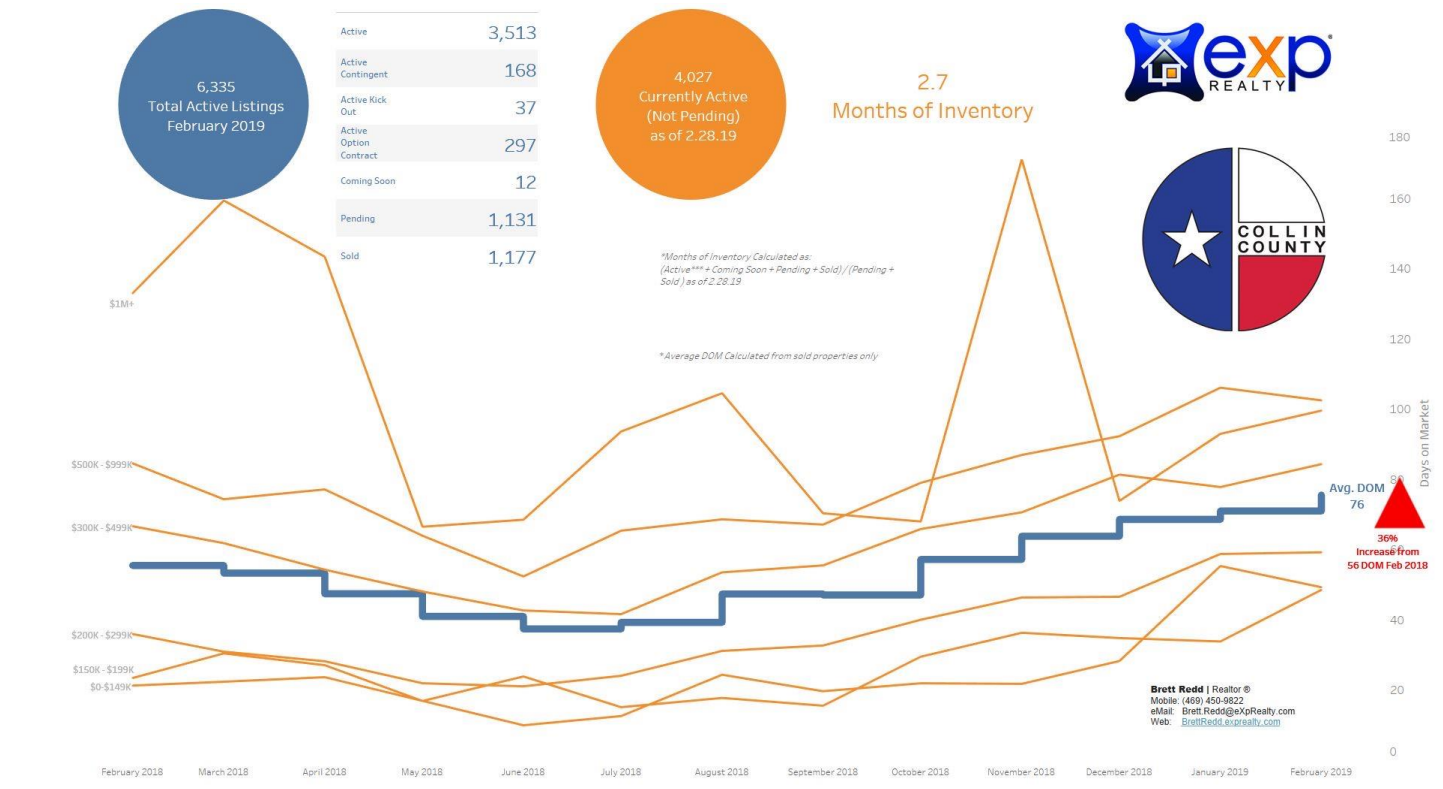
Below, the vertical, sequential marks represent the cost/SF range. The line chart represents average total cost within each SF range. vvv



Average Home Price & Cost/SF by Subdivision

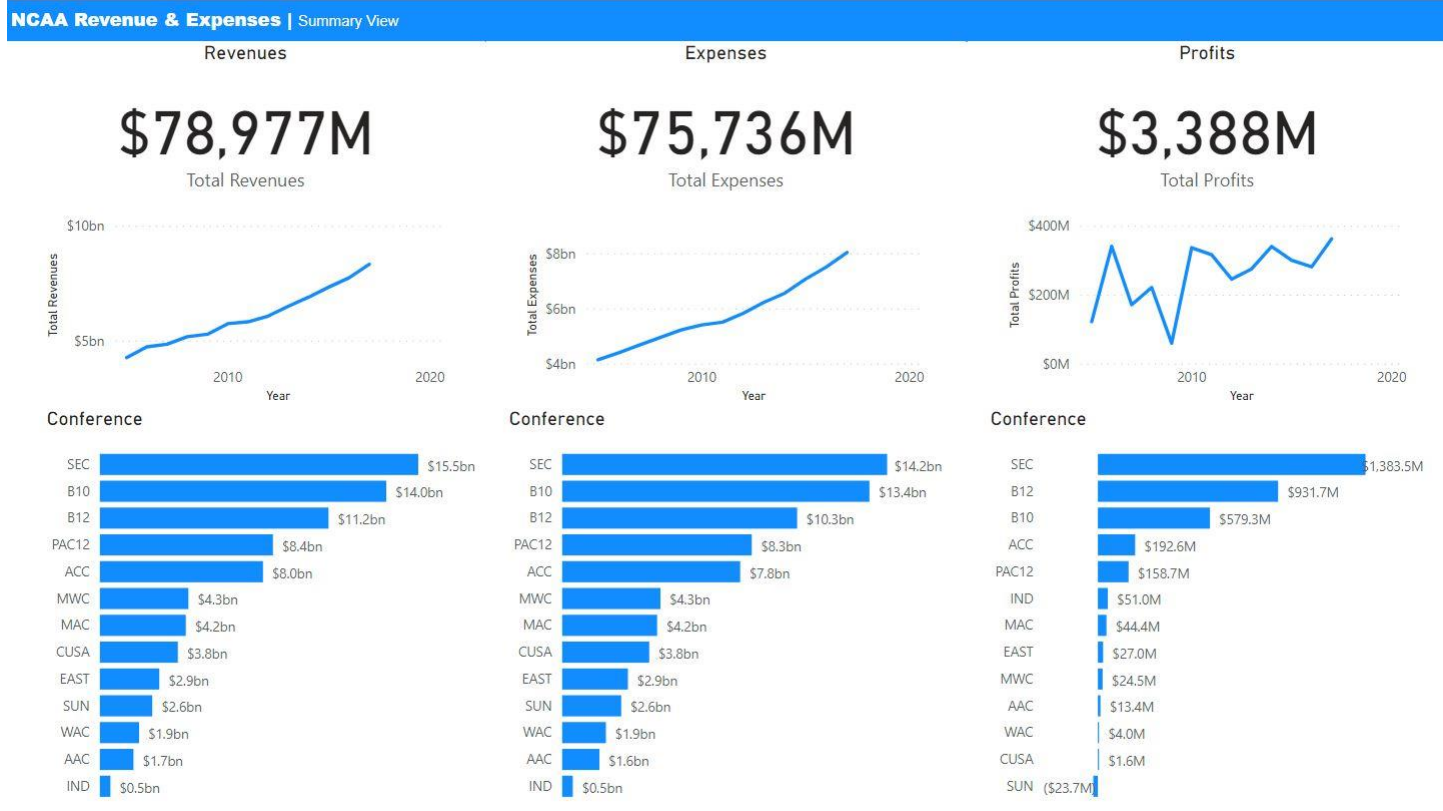


*Developed when between careers in 2019, leveraging my real estate license to sale homes. I still hold a Texas real estate license, but it is currently inactive while I place my full focus into obtaining another career in analytics.

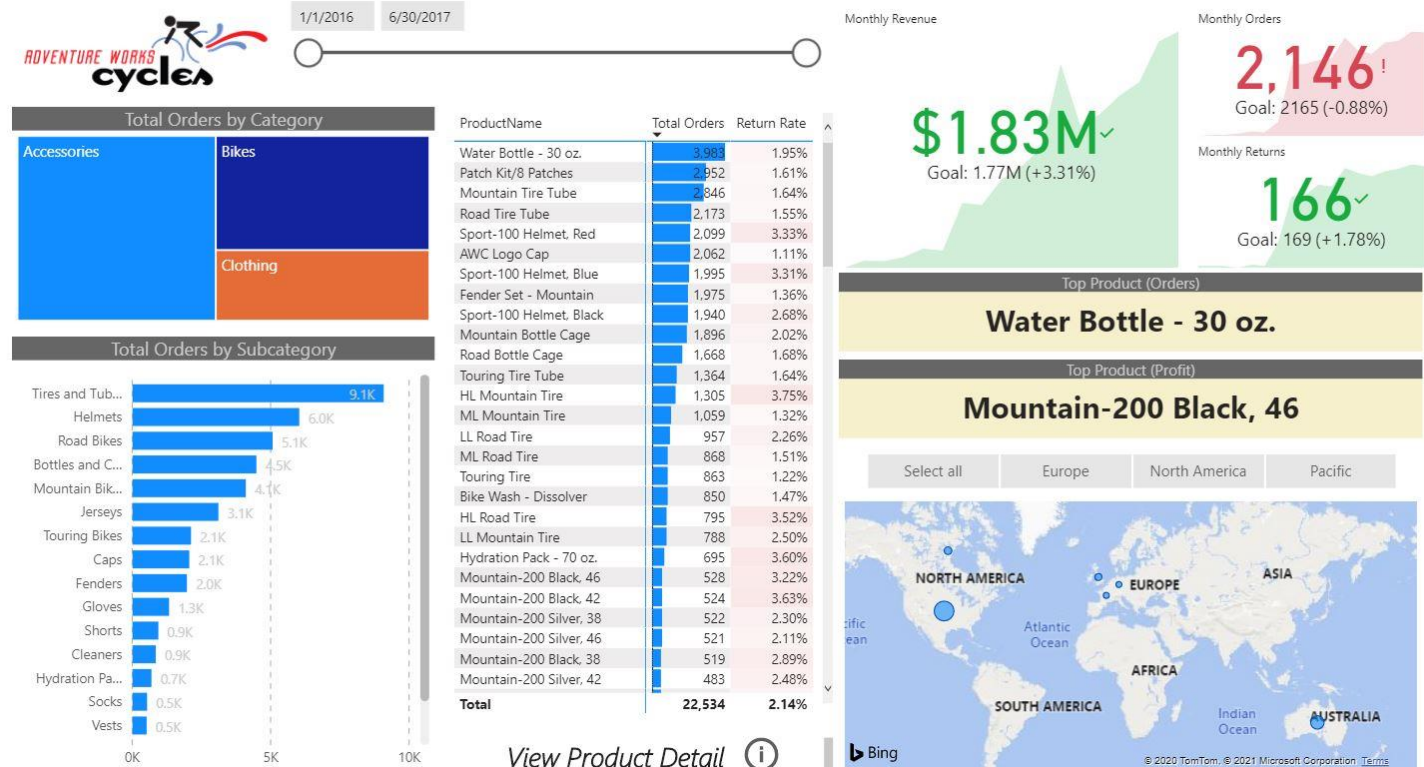


*Also developed when between careers in 2019, leveraging my real estate license to sale homes. I still hold a Texas real estate license, but it is currently inactive while I place my full focus into obtaining another career in analytics.

POWERBI



*Simple dashboard developed through the new PowerBI section, as of 2021, on the Workout Wednesday website.



*Developed as a milestone project within the Maven Analytics Up & Running PowerBI Udemey course.



*Also developed as a milestone project within the Up & Running Maven Analytics PowerBI Udemey course.



CLEAR
FILTERS

Current Month Transactions

18,325 ✓

Goal: 17,339 (+5.69%)

Current Month Profit

\$71,682 ✓

Goal: \$67,872 (+5.61%)

Current Month Returns

496 !

Goal: 482 (-2.9%)

product_brand Total Transactions Total Profit Profit Margin Return Rate

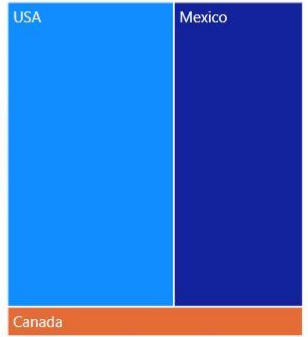
Hermanos	5,342	\$21,753	58.64%	0.95%
Ebony	5,238	\$20,354	59.81%	0.96%
Tell Tale	5,112	\$19,982	58.05%	0.99%
Tri-State	5,099	\$19,980	58.91%	1.10%
High Top	4,940	\$19,810	60.42%	1.01%
Nationeel	4,408	\$18,617	60.44%	1.18%
Best Choice	4,218	\$18,355	60.64%	0.81%
Horatio	4,195	\$17,737	58.42%	1.26%
Fort West	4,108	\$15,834	59.80%	0.97%
Fast	4,097	\$16,469	61.03%	1.07%
Sunset	3,953	\$14,018	60.45%	1.03%
Carrington	3,891	\$14,883	59.52%	0.78%
Red Wing	3,870	\$15,870	59.36%	1.06%
Big Time	3,816	\$15,560	60.20%	1.05%
Cormorant	3,744	\$15,749	61.60%	0.87%
Imagine	3,634	\$15,102	61.40%	1.06%
Super	3,618	\$13,868	60.59%	0.96%
Denny	3,584	\$16,015	58.02%	0.99%
High Quality	3,577	\$16,139	59.98%	1.13%
Golden	3,550	\$13,256	58.72%	0.88%
BBB Best	3,514	\$12,991	62.12%	0.80%
PigTail	3,467	\$11,617	60.68%	1.04%
Plato	3,352	\$12,748	63.55%	1.06%
Landslide	3,270	\$10,647	58.65%	0.98%
CDR	3,078	\$12,062	58.98%	1.11%
Better	2,823	\$9,179	61.15%	1.07%
Carlson	2,564	\$10,534	61.20%	0.97%
Pleasant	2,564	\$10,187	60.18%	0.92%
Just Right	2,558	\$9,283	59.54%	0.83%
Bravo	2,484	\$11,027	59.15%	0.82%
Total	113,668	\$449,627	59.94%	1.00%

Select all

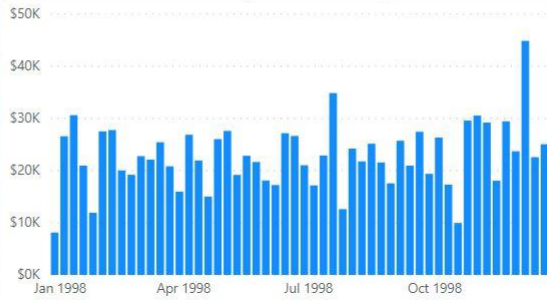
Canada

Mexico

USA



Weekly Revenue Trending



Revenue vs. Target



*Also developed as a milestone project within the Up & Running Maven Analytics PowerBI Udemy course.

PYTHON SCRIPTING EXERCISE MIMICKING COFFEE MACHINE OPERATION

Course: 100 days of Code – The Complete Python Pro Bootcamp

```
MENU = {  
    "espresso": {  
        "ingredients": {  
            "water": 50,  
            "coffee": 18,  
        },  
        "cost": 1.5,  
    },  
    "latte": {  
        "ingredients": {  
            "water": 200,  
            "milk": 150,  
            "coffee": 24,  
        },  
        "cost": 2.5,  
    },  
    "cappuccino": {  
        "ingredients": {  
            "water": 250,  
            "milk": 100,  
            "coffee": 24,  
        },  
        "cost": 3.0,  
    },  
}
```

```
RESOURCES = {  
    "water": 300,  
    "milk": 200,  
    "coffee": 100,  
    "money": 0  
}
```

```
COIN_VALUES = {  
    'penny': .01,  
    'nickel': .05,  
    'dime': .10,  
    'quarter': .25  
}
```

```
MACHINE = 'on'
```

```

def report():
    """Returns machine resource inventory"""
    print(f"Water: {RESOURCES['water']}ml\nMilk: {RESOURCES['milk']}ml\n"
          f"Coffee: {RESOURCES['coffee']}g\nMoney: ${format(RESOURCES['money'], '.2f')}")

def money_inserted():
    """Returns the total calculated from coins inserted."""
    global COIN_VALUES
    print('Please insert coins.')
    while True:
        try:
            quarter = int(input('How many quarters?: '))
            dime = int(input('How many dimes?: '))
            nickel = int(input('How many nickels?: '))
            penny = int(input('How many pennies?: '))
            break
        except ValueError:
            print('Invalid selection. Please try again.')
    deposited = (quarter * COIN_VALUES['quarter']) \
        + (dime * COIN_VALUES['dime']) \
        + (nickel * COIN_VALUES['nickel']) \
        + (penny * COIN_VALUES['penny'])
    return deposited

def check_resources(order_ingredients):
    """Returns True when order can be made, False if resources are insufficient."""
    global RESOURCES
    for item in order_ingredients:
        if order_ingredients[item] > RESOURCES[item]:
            print(f'Sorry, there is not enough {item}.')
    for item in order_ingredients:
        if order_ingredients[item] > RESOURCES[item]:
            return False
    return True

```

```

def check_amt(paid, cost):
    """Returns refund amount if coins inserted meet or exceed item cost,
    returns -1 and prints not enough statement if not."""
    if paid >= cost:
        refund = paid - cost
        return refund
    else:
        print("Sorry that's not enough money. Money refunded.")
        refund = -1
        return refund

def update_resources(order_ingredients, cost):
    """Updates the values in the resources dictionary after successful transaction."""
    global RESOURCES
    for item in order_ingredients:
        RESOURCES[item] -= order_ingredients[item]
    RESOURCES['money'] += cost

```

```

def use_machine():
    """Full code to operate the coffee machine"""
    global MACHINE
    global MENU
    while MACHINE == 'on':
        user_input = input('What would you like? (espresso/latte/cappuccino): ').lower()
        if user_input == 'report':
            report()
        elif user_input == 'off':
            MACHINE = 'off'
            print('Powering down.')
        elif user_input == 'espresso' or user_input == 'latte' or user_input == 'cappuccino':
            drink = MENU[user_input]
            cost = MENU[user_input]['cost']
            is_available = check_resources(drink['ingredients'])
            if is_available:
                money_in = money_inserted()
                refund = check_amt(money_in, cost)
                if refund >= 0:
                    print(f"Here is ${format(refund, '.2f')} in change.")
                    print(f'Here is your {user_input} ☕ Enjoy!')
                    update_resources(drink['ingredients'], cost)
            else:
                print('Invalid selection. Please try again.')

use_machine()

```