**Individual Report Based on Group Project Data Model**

To start off, I will be talking about the three data visualizations I made using the Power BI Software. I’ve even added another visualization that I thought was useful however, I was having trouble understanding what the chart was showing so I will not be giving an depth analysis on the last chart.

The first page I will be discussing is a Pie Chart that I created which involve PriorityTix and ReassignCount. Both values are taken straight from the ServiceNow database that I created in the group project portion of the class. ReassignCount is the number of times an incident (ticket) changes analysts hands and is tracked by a “Int” datatype. PriorityTix is calculated by taking the “Impact” column and the “Urgency” column and creating a number and an associated tag, which is tracked by a “varchar” datatype. I chose to use these two values to analyze because I wanted to see the percentage of the type of priority tickets that were reassigned to other workers. What I gained from this visualization is that 92% of reassigned tickets were moderate, while the other categories fell to around 2% reassigned count for low, high and critical priority. From a managerial perspective this would make sense since the highest volume of tickets would account for most of the reassigns, this chart is just used to show the data in visually appealing format.

On page two of my PowerBI report, I decided to implement a Stacked Bar Chart. I chose this visualization because I liked how you could easily see which CloseCode was getting the most use. Also, the use of multiple colors to show the percentage of priority of each close code fit nicely into the design. The values that I pulled from the ServiceNow database were CloseCode, ReopenCount, and PriorityTix. As discussed earlier, we already know what PriorityTix is, however CloseCode is the ID that is assigned when a ticket is resolved, which is tracked by a “varchar” datatype. ReopenCount is the number of times the ticket was reopened, which is tracked by a “Int” datatype. What I thought was interesting was to see the gap from obviously the first Code 6 and the rest of the codes, but even from Code 9 to 4 and how they all seem close then there is a drop off. A manager can use this chart to determine a few things that can be beneficial for their department. They can determine which close codes are the most common meaning what type of work are they receiving the most reopens on, which in turn can mean more training on these types of tickets. Also using this chart, you can see what type of priority each ticket that reopened have, so insight can be gained as well.

The last chart I will go over includes some of the Python code I used to create a visualization. I decided I would try out a Violin plot using the MatPlotLib package and the Seaborn package of Python. Using the code snippet below I was able to create this visualization that shows a visually appealing design that also allows for insight.

import matplotlib.pyplot as plt

import seaborn as sns

sns.violinplot(x=dataset["PriorityTix"], y=dataset["SysModCount"], palette="Blues",bw=.1)

plt.show()

I used the PriorityTix value again but decided to use SysModCount which is the number of modifications a ticket receives and is tracked by a “Int” data type. On the Y-axis SysModCount is tracked to show how much of each PriorityTix is listed. The way a Violin plot works is it takes the number of times an event happens and creates a shape based off where the bulk of the data is and the distribution. I thought this was a great idea to use because a manager could quickly see which priority has a certain distribution. With this visualization, Critical mainly had the bulk of their distribution in the 0-40 modifications range, with the median around 20. The High priority had a very similar distribution with the only difference being in the number of modifications occurring. Moderate was fairly balanced with the median around 60, while low was really all over the board with their main distribution between 0-60 and their median around 40. Overall, I thought this was a great visualization and I had a lot of fun dealing with the Python code.

I decided to run a bonus chart using python and the seaborn visualization known as a PairPlot. I liked this chart because I know it is supposed to show the correlation between the values, but I was having trouble understanding what was displayed. I would love to have a class in the future that we go over more advanced visualizations, their descriptions, and their applications. Overall, I really liked this assignment and the PowerBI tool.







