Task Start Here

In [ ]:
```
#Importing libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [ ]:
```
import warnings
warnings.filterwarnings("ignore")
```

In [ ]:
```
benefits = pd.read_csv('BenefitsCostSharing.csv') #loading data
```

In [ ]:
```
rate = pd.read_csv('Rate.csv')
```

In [ ]:
```
benefits.info() #data information - original
```

In [ ]:
```
benefits.isnull().sum() #checking for null values
```

In [ ]:
```
benefits['StateCode'].equals(benefits['StateCode2']) #checking for duplicate column (different column name)
```

In [ ]:
```
benefits['IssuerId'].equals(benefits['IssuerId2']) #checking for duplicate column (different column name)
```

In [ ]:
```
benefits['BenefitName'].describe() #detail information about targeted column for analysis.
```

In [ ]:
```
benefits['StateCode'].describe()
```

In [ ]:
```
benefits['PlanId'].describe() #detail information about targeted column for analysis.
```

In [ ]:
```
benefits['IsCovered'].unique() #unique values present in the column
```

In [ ]:
```
#converting value to lowercase to avoid same values with different cases
benefits['IsCovered'] = benefits['IsCovered'].str.lower()
print(benefits['IsCovered'].unique())
```

In [ ]:
```
benefits['IssuerId'].nunique()
```

In [ ]:
```
benefits['ImportDate'].describe() #detail information about targeted column for analysis.
```

In [ ]:
```
benefits['SourceName'].describe()
```

In [ ]:
```
benefits['SourceName'].nunique() #count of unique value present in the data
```

In [ ]:
```
#creating new dataframe for the targted column only for analysis from the benefits dataframe
selected_columns = ['StateCode', 'BusinessYear','BenefitName', 'IssuerId', 'PlanId', 'IsCovered', 'ImportDate', 'SourceName']
benefits_new = benefits[selected_columns]
```

In [ ]:
```
benefits_new
```

In [ ]:
```
benefits_new.info() #new derived dataframe information
```

In [ ]:
```
benefits_new.isnull().sum() #checking for null values
```

In [ ]:
```
# missing values in 'IsCovered' with a 'Unknown' (not droping null values purposely)
benefits_new['IsCovered'].fillna('Unknown', inplace=True)
```

```
print(benefits_new.isnull().sum())
```

In [ ]:
```
#maximum presence of 'Unknown' in IsCovered column
state_max = benefits_new.groupby('StateCode')['IsCovered'].apply(lambda x: (x == 'Unknown').sum()).idxmax()

# minimum presence of 'Unknown' in IsCovered column
state_min = benefits_new.groupby('StateCode')['IsCovered'].apply(lambda x: (x == 'Unknown').sum()).idxmin()

#top 5 state codes with the highest presence of 'Unknown'
top_unknown = benefits_new.groupby('StateCode')['IsCovered'].apply(lambda x: (x == 'Unknown').sum()).nlargest(5)

print("Top 5 state codes with the highest occurrences of 'Unknown' values:\n")
print(top_unknown)

print(f"\nState code with maximum 'Unknown': {state_max}")
print(f"State code with minimum 'Unknown': {state_min}")
```

In [ ]:
```
benefits_new['ImportDate'].describe()
```

In [ ]:
```
#ImportDate to datetime format for better analysis and getting some key insight w.r.t. data and time
benefits_new['ImportDate'] = pd.to_datetime(benefits_new['ImportDate'])
```

In [ ]:
```
print(benefits_new['ImportDate'].describe()) #deatail information about the column
```

In [ ]:
```
benefits_new.info() #info of new dataframe after all neccessary transformation
```

In [ ]:
```
benefits_new
```

Finding - 1 :

In [ ]:
```
#count each businessyear in the overall data column
business_year_counts = benefits_new['BusinessYear'].value_counts().sort_index()
```

```
print(business_year_counts)
```

In [ ]:

```
plt.figure(figsize=(10, 6))
sns.barplot(x=business_year_counts.index, y=business_year_counts.values, palette='viridis')
plt.title('Distribution of Business Years')
plt.xlabel('Business Year')
plt.ylabel('Count')
plt.show()
```

Finding - 2 :

In [ ]:

```
#value count of each unique value present in the column
covered_counts = benefits_new['IsCovered'].value_counts()
print(covered_counts)
```

In [ ]:

```
plt.figure(figsize=(8, 6))
sns.barplot(x=covered_counts.index, y=covered_counts.values, palette='Set2')
plt.title('Distribution of Covered vs. Not Covered')
plt.xlabel('IsCovered')
plt.ylabel('Count')
plt.show()
```

Finding - 3 :

In [ ]:

```
#top 10 issuers
issuer_counts = benefits_new['IssuerId'].value_counts().sort_values(ascending=False)[:10]
print(issuer_counts)
```

In [ ]:
```
plt.figure(figsize=(12, 6))
sns.barplot(x=issuer_counts.index, y=issuer_counts.values, palette='Set3')
plt.title('Top 10 Issuers')
plt.xlabel('IssuerId')
plt.ylabel('Count')
plt.show()
```

Finding - 4 :

In [ ]:
```
#value count of each state / distiribution
state_counts = benefits_new['StateCode'].value_counts()
print(state_counts.head()) #viewing top 5 only
```

In [ ]:
```
plt.figure(figsize=(15, 8))
sns.barplot(x=state_counts.index, y=state_counts.values, palette='viridis')
plt.title('Distribution of StateCode')
plt.xlabel('StateCode')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.show()
```

Finding - 5 :

In [ ]:
```
#distribution of IsCovered by StateCode
iscovered_state_distribution = pd.crosstab(benefits_new['StateCode'], benefits_new['IsCovered'], margins=True, margins_name="Total
print(iscovered_state_distribution.head())
```

In [ ]:
```
plt.figure(figsize=(20, 20))
iscovered_state_distribution.drop("Total").plot(kind='bar', stacked=True, colormap='viridis')
plt.title('IsCovered Distribution by StateCode')
plt.xlabel('StateCode')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')  # Rotate x-axis labels
plt.show()
```

Finding - 6 :

In [ ]:
```
#top 5 most common benefit name values
top_5_benefit_names = benefits_new['BenefitName'].value_counts().head(5)

print(top_5_benefit_names)
```

Finding - 7 :

In [ ]:
```
#highest used benefit name for each state
highest_benefit_by_state = benefits_new.groupby('StateCode')['BenefitName'].agg(lambda x: x.value_counts().idxmax()).reset_index()

highest_benefit_by_state.columns = ['StateCode', 'Highest BenefitName']
print(highest_benefit_by_state.head()) #top 5 from result
```

Finding - 8 :

In [ ]:
```
#unique plans for each SourceName
plans_by_source = benefits_new.groupby('SourceName')['PlanId'].nunique()
print(plans_by_source)
```

In [ ]:
```
plt.figure(figsize=(15, 8))
plans_by_source.sort_values(ascending=False).plot(kind='bar', colormap='Set3')
plt.title('Count of Plans by SourceName')
plt.xlabel('SourceName')
plt.ylabel('Count of Plans')
plt.xticks(rotation=45, ha='right')
plt.show()
```

Finding - 9 :

In [ ]:
```
#count of unique plans for each SourceName and StateCode
plans_by_source_state = benefits_new.groupby(['SourceName', 'StateCode'])['PlanId'].nunique().unstack()
```

In [ ]:
```
plt.figure(figsize=(15, 10))
sns.heatmap(plans_by_source_state, cmap='viridis', annot=True, fmt='g', cbar_kws={'label': 'Count'})
plt.title('Count of Plans by SourceName and StateCode')
plt.xlabel('StateCode')
plt.ylabel('SourceName')
plt.show()
```

Finding - 10 :

In [ ]:
```
#count of frequency between StateCode and SourceName
state_source_cross_tab = pd.crosstab(benefits_new['StateCode'], benefits_new['SourceName'])
```

In [ ]:
```
plt.figure(figsize=(15, 10))
sns.heatmap(state_source_cross_tab, cmap='viridis', annot=True, fmt='g', cbar_kws={'label': 'Count'})
plt.title('Count of Occurrences between StateCode and SourceName')
plt.xlabel('SourceName')
plt.ylabel('StateCode')
plt.show()
```

Conclusion 1

Based on the above analysis and visualization, below are some key findings from that.

The analysis of the health insurance benefits dataset provides valuable insights into various aspects of the insurance plans.

1. Business Year Distribution:

   The dataset spans business years 2014 to 2016. The distribution of plans across these years is as follows: 2014: 1164869 plans 2015: 2079286 plans 2016: 1804253 plans

---

1. Coverage Distribution:

   The majority of plans are labeled as covered (3934525 plans), followed by not covered (897903 plans), and Unknown status (215980 plans).

---

1. Top Issuers:

   The top 10 issuers by Issuer ID are identified based on the number of plans they offer. The issuer with ID 33602 leads with 241412 plans, followed by others in descending order.

---

1. State-wise Distribution:

   The dataset covers plans from various states, with WI having the highest representation (512587 plans), followed by TX, FL, OH, and IL.

---

1. IsCovered Distribution by State:

   IsCovered against StateCode provides a breakdown of covered, not covered, and unknown status for each state.

---

1. Top 5 Benefit Names:

   The most common benefit names and their respective counts are identified. The top 5 benefits are related to dental care and orthodontia.

---

1. Plan Count by SourceName:

   The count of unique plans for each SourceName is analyzed. HIOS has the highest number of plans (31541), followed by SERFF, and OPM.

In [ ]:

Understand the Market Dynamics for Health Insurance Plans & develop a Targeted Marketing Strategy

In [ ]:
```
rate.head()
```

In [ ]:
```
rate.info() #data info - original
```

In [ ]:
```
rate.isnull().sum()
```

In [ ]:
```
rate
```

In [ ]:
```
rate['BusinessYear'].value_counts()
```

In [ ]:
```
rate['StateCode'].describe() #detail information about targeted column for analysis.
```

In [ ]:
```
rate['IssuerId'].nunique() #count of unique value present in the data
```

In [ ]:
```
rate['SourceName'].value_counts() #value count of each unique value present in the data
```

In [ ]:
```
rate['PlanId'].describe()
```

In [ ]:
```
rate['Tobacco'].describe() #detail information about targeted column for analysis.
```

In [ ]:
```
rate['Tobacco'].value_counts()
```

In [ ]:
```
rate['Age'].describe()
```

In [ ]:
```
rate['Age'].value_counts().head()
```

In [ ]:
```
rate['ImportDate'].describe()
```

In [ ]:
```
rate['IssuerId'].equals(rate['IssuerId2']) #checking for duplicate column (different column name)
```

In [ ]:
```
rate['RatingAreaId'].describe()
```

In [ ]:
```
rate['RatingAreaId'].nunique()
```

In [ ]:
```
rate['RatingAreaId'].value_counts()  #value count of each unique value present in the data
```

In [ ]:
```
rate['IndividualRate'].nunique()
```

In [ ]:
```
rate['IndividualTobaccoRate'].value_counts()
```

In [ ]:
```
#targeted columns from the rate DataFrame into the new dataframe
rate_new = rate[['BusinessYear', 'StateCode', 'IssuerId', 'SourceName', 'PlanId',
                 'Tobacco', 'Age', 'ImportDate', 'IndividualRate']]
```

In [ ]:
```
rate_new.head()
```

In [ ]:
```
rate_new.info() #new derived dataframe information
```

In [ ]:
```
rate_new['Age'].unique()
```

In [ ]:
```
#ImportDate to datetime format
rate_new['ImportDate'] = pd.to_datetime(rate_new['ImportDate'])
```

In [ ]:
```
rate_new.info()
```

Analysis Start Here:

Understand the distribution and characteristics of health insurance plans across different states and age groups to identify potential markets to target or areas where the company could improve its offerings. Understand the value proposition of different plans and recommend the top 5 avenues where marketing effort should be spent by plan, age, and state.

Note: I intentionally not including other parameter as task is focused majorly on Age, State and Plan.

Analyzing Individual Rates by Plan (Report - 1):

The distribution of individual rates for each health insurance plan.

In [ ]:
```
individual_rates_stats = rate_new.groupby('PlanId')['IndividualRate'].describe()

print("Summary Statistics for Individual Rates by Plan:")
print(individual_rates_stats.to_string())
```

Plans with competitive pricing or unique value propositions.

In [ ]:
```
competitive_plans = individual_rates_stats.sort_values('mean').head(10)

print("Top 10 Plans with Competitive Pricing:\n")
print(competitive_plans.to_string())
```

Plan Distribution Across Age Groups (Report - 2):

The distribution of health insurance plans across different age groups.

In [ ]:
```
plans_distribution_age = rate_new.groupby('Age')['PlanId'].nunique()

print("Plan Distribution Across Age Groups:")
print(plans_distribution_age.to_frame().to_string())
```

Plans that are popular among specific age demographics.

In [ ]:
```
popular_plans_by_age = rate_new.groupby(['Age', 'PlanId']).size().groupby('Age').idxmax().to_frame().reset_index()
popular_plans_by_age.columns = ['Age', 'Most Popular Plan']

print("Popular Plans by Age:\n")
print(popular_plans_by_age.to_string(index=False))
```

Plan Distribution Across States (Report - 3):

Geographical distribution of health insurance plans across states.

In [ ]:
```
plans_distribution_state = rate_new.groupby('StateCode')['PlanId'].nunique()

print("Plan Distribution Across States:")
print(plans_distribution_state.to_frame().to_string())
```

States where specific plans have a higher market share.

In [ ]:
```
popular_plans_by_state = rate_new.groupby(['StateCode', 'PlanId']).size().groupby('StateCode').idxmax().to_frame().reset_index()
popular_plans_by_state.columns = ['StateCode', 'Most Popular Plan']

print("\nPopular Plans by State:")
print(popular_plans_by_state.to_string(index=False))
```

Top Plans by Age and State (Report - 4):

Top health insurance plans for each age group within each state.

In [ ]:
```
top_plans_by_age_state = rate_new.groupby(['Age', 'StateCode', 'PlanId'])['IndividualRate'].mean().groupby(['Age', 'StateCode']).i

print("Top Plans by Age and State:")
print(top_plans_by_age_state.to_frame().to_string())
```

Overall Market Presence (Report - 5):

Assess the overall market presence of each health insurance plan.

In [ ]:
```
overall_market_presence = rate_new.groupby('PlanId')['IndividualRate'].count()

print("Overall Market Presence of Each Plan:")
print(overall_market_presence.to_frame().to_string())
```

Plans that have a strong presence across various age groups and states.

In [ ]:
```
strong_presence_plans = overall_market_presence.sort_values(ascending=False).head(10)

print("\nTop 10 Plans with Strong Market Presence:")
print(strong_presence_plans.to_frame().to_string())
```

Conculsion : Targeted Marketing Strategy and Planning based on Report Generated.

1. Individual Rates by Plan:

   (Refer to analysis Report - 1 for actual generated information and detail for the same.)

   Aanalysis of individual rates for various health insurance plans. This involved exploring the distribution of rates for each plan, identifying plans with competitive pricing or unique value propositions. The analysis provides insights into the affordability and distinctiveness of different health insurance offerings.

---

1. Plan Distribution Across Age Groups:

   (Refer to analysis Report - 2 for actual generated information and detail for the same.)

   Health insurance plans are distributed across different age groups. By identifying plans that are popular among specific age demographics, we gained valuable insights into the preferences and needs of various age brackets. This information is crucial for tailoring marketing strategies and

plan designs to specific age-related requirements.

---

1. Plan Distribution Across States:

   (Refer to analysis Report - 3 for actual generated information and detail for the same.)

   The geographical distribution of health insurance plans across states was explored, helping us identify regions where specific plans have a higher market share. Understanding the state-wise popularity of plans is vital for targeted marketing efforts and strategic planning to meet the diverse needs of different regions.

---

1. Top Plans by Age and State:

   (Refer to analysis Report - 4 for actual generated information and detail for the same.)

   The top health insurance plans for each age group within each state allows us to focus on plans that resonate well with specific age demographics in different regions. This information is instrumental in designing regional marketing campaigns and optimizing plans based on local preferences.

---

1. Overall Market Presence:

   (Refer to analysis Report - 5 for actual generated information and detail for the same.)

   The overall market presence of each health insurance plan, identifying plans that have a strong presence across various age groups and states. This global perspective is valuable for making decisions that impact the entire market and can guide efforts to strengthen the market position of specific plans.

In summary, this analysis provides a comprehensive view of the health insurance landscape, offering actionable insights for marketing strategies, plan optimization, and regional customization to meet the diverse needs of the market.

Task Ends Here