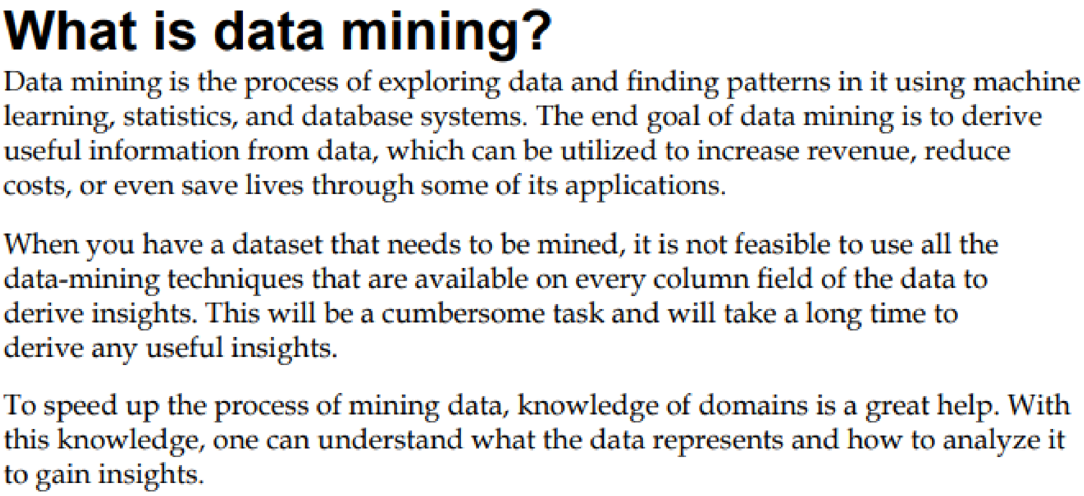
Start from Today - 3

Data mining

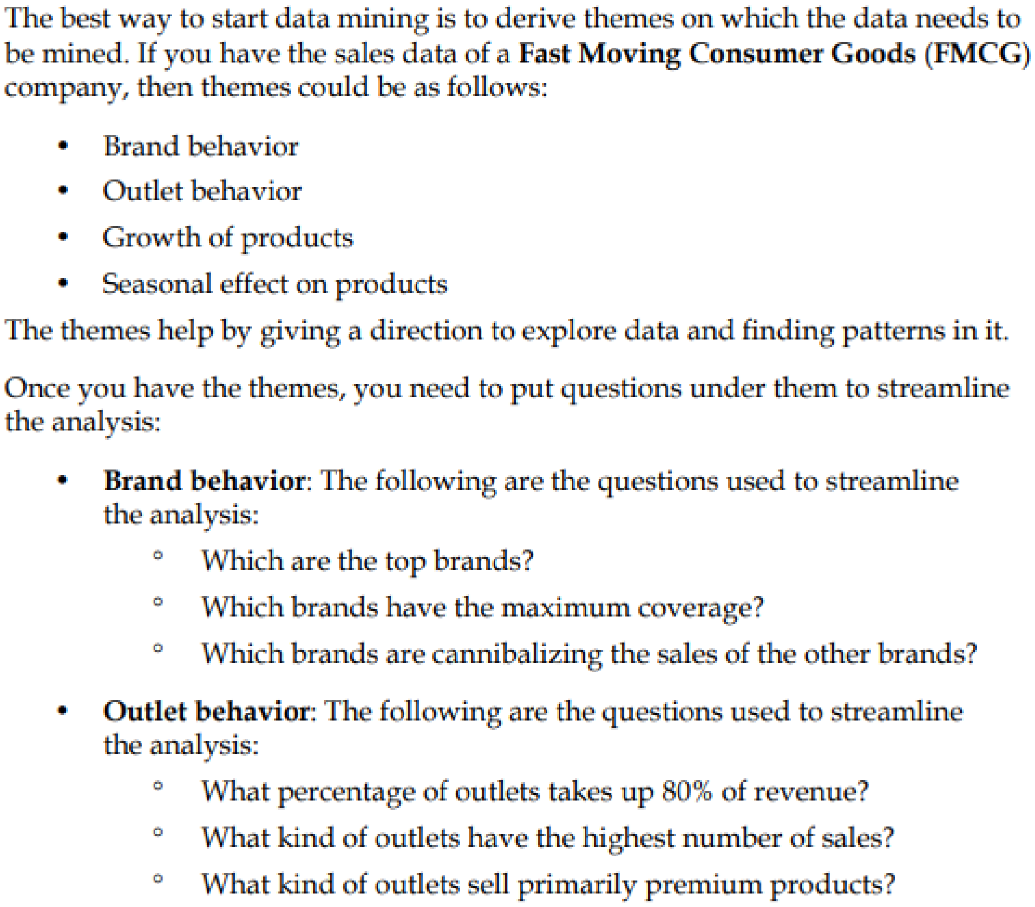


What is data mining?

Data mining is the process of exploring data and finding patterns in it using machine learning, statistics, and database systems. The end goal of data mining is to derive useful information from data, which can be utilized to increase revenue, reduce costs, or even save lives through some of its applications.

When you have a dataset that needs to be mined, it is not feasible to use all the data-mining techniques that are available on every column field of the data to derive insights. This will be a cumbersome task and will take a long time to derive any useful insights.

To speed up the process of mining data, knowledge of domains is a great help. With this knowledge, one can understand what the data represents and how to analyze it to gain insights.



The best way to start data mining is to derive themes on which the data needs to be mined. If you have the sales data of a **Fast Moving Consumer Goods** (**FMCG**) company, then themes could be as follows:

- Brand behavior
- Outlet behavior
- Growth of products
- Seasonal effect on products

The themes help by giving a direction to explore data and finding patterns in it.

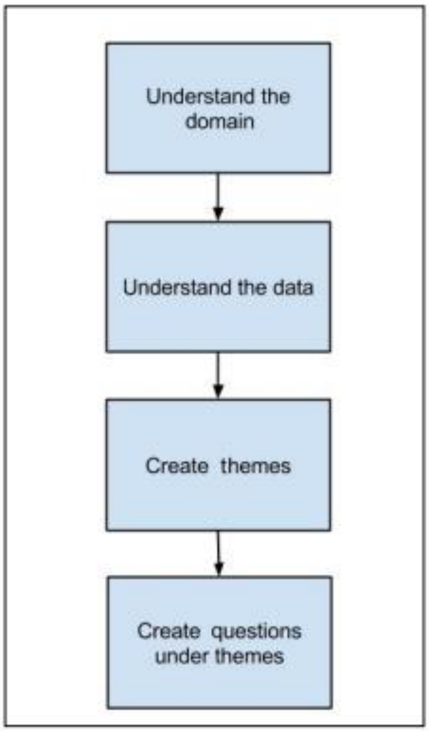
Once you have the themes, you need to put questions under them to streamline the analysis:

- **Brand behavior**: The following are the questions used to streamline the analysis:
    - Which are the top brands?
    - Which brands have the maximum coverage?
    - Which brands are cannibalizing the sales of the other brands?
- **Outlet behavior**: The following are the questions used to streamline the analysis:
    - What percentage of outlets takes up 80% of revenue?
    - What kind of outlets have the highest number of sales?
    - What kind of outlets sell primarily premium products?

- **Growth of products**: The following are the questions used to streamline the analysis:
    - Which are the fastest growing brands in terms of sale?
    - Which are the fastest growing brands in terms of volume?
    - Which brand's growth has flattened out?
- **Seasonal effect of the products**: The following are the questions used to streamline the analysis:
    - How many brands are seasonal?
    - What is the difference in terms of sales during seasonal and nonseasonal periods?
    - Which holiday brings in the maximum amount of sales for a particular brand?

The preceding questions under these themes give pinpointed directions to find patterns and perform an analysis that gives some quality results.

The process of exploring data can be summarized by the following flow chart:



Presenting an analysis

After performing the analysis, you would need to present some observations. The most commonly used medium for doing this is through Microsoft PowerPoint presentations. The result of your analysis could be a construct in the form of a chart or table. When presenting these constructs, there is certain information that should be added to your slides. This is one of the most common templates used:

Here are the different sections of the preceding image:

- **Question**: The topmost part of the template should describe the problem statement that the particular analysis is trying to address.
- **Observation**: Here, the observations from the construct are listed in a vertical column. Sometimes, the observations can be marked over the construct using arrow marks or dialog boxes.
- **Key Takeaway**: Toward the bottom of the image, you can describe what is concluded from the chart.

Studying the Titanic

To perform the data analysis, we'll be using the Titanic dataset from Kaggle.

This dataset is simple to understand and does not require any domain understanding to derive insights.

This dataset contains the details of each passenger on the Titanic and also whether they survived or not.

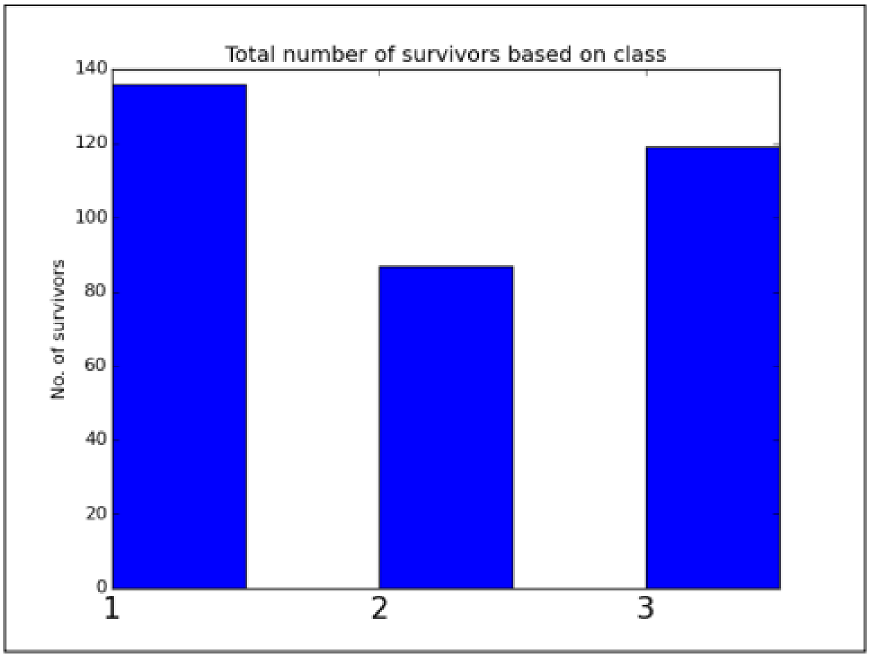
The following are the field descriptions:

Field	**Descriptions**
`survival`	Survival(`0` = No, `1` = Yes)
`pclass`	Passenger class(`1` = 1st, `2` = 2nd, `3` = 3rd)
`name`	Name of the passenger
`sex`	Gender of the passenger
`age`	Age of the passenger
`sibsp`	Number of siblings/spouses aboard
`parch`	Number of parents/children aboard
`ticket`	Ticket number
`fare`	Passenger fare
`cabin`	Cabin
`embarked`	Port of embarkation (`C` = Cherbourg, `Q` = Queenstown, `S` = Southampton)

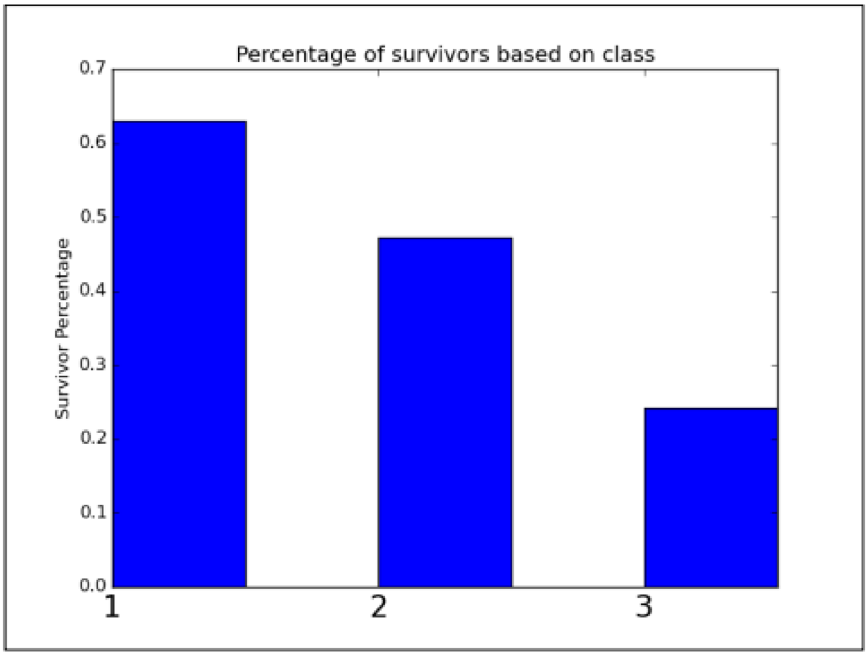
Which passenger class has the maximum number of survivors?

To answer this question, we'll construct a simple bar plot of the number of survivors and the percentage of survivors in each class, respectively. You can do this using the following command:

```
>>> import pandas as pd
>>> import pylab as plt
>>> import numpy as np

>>> df = pd.read_csv('Data/titanic data.csv')

>>> df['Pclass'].isnull().value_counts()
>>> False    891
>>> dtype: int64

>>> df['Survived'].isnull().value_counts()
>>> False    891
>>> dtype: int64

>>> #Passengers survived in each class
>>> survivors = df.groupby('Pclass')['Survived'].agg(sum)


>>> #Total Passengers in each class
>>> total_passengers = df.groupby('Pclass')['PassengerId'].count()
>>> survivor_percentage = survivors / total_passengers

>>> #Plotting the Total number of survivors
>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)
>>> rect = ax.bar(survivors.index.values.tolist(),
          survivors, color='blue', width=0.5)
>>> ax.set_ylabel('No. of survivors')
>>> ax.set_title('Total number of survivors based on class')
>>> xTickMarks = survivors.index.values.tolist()
>>> ax.set_xticks(survivors.index.values.tolist())
>>> xtickNames = ax.set_xticklabels(xTickMarks)
>>> plt.setp(xtickNames, fontsize=20)
>>> plt.show()
```



```
>>> #Plotting the percentage of survivors in each class

>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)


    >>> rect = ax.bar(survivor_percentage.index.values.tolist(),
              survivor_percentage, color='blue', width=0.5)
    >>> ax.set_ylabel('Survivor Percentage')
    >>> ax.set_title('Percentage of survivors based on class')
    >>> xTickMarks = survivors.index.values.tolist()
    >>> ax.set_xticks(survivors.index.values.tolist())
    >>> xtickNames = ax.set_xticklabels(xTickMarks)
    >>> plt.setp(xtickNames, fontsize=20)
    >>> plt.show()
```



In the preceding code, we performed a preliminary check for null values on the fields that are utilized. After this, we calculated the number of survivors and the percentage of survivors in each class. Then, we plotted two bar charts for the total number of survivors and the percentage of survivors.

These are our observations:

- The maximum number of survivors are in the first and third class, respectively
- With respect to the total number of passengers in each class, first class has the maximum survivors at around 61%

- With respect to the total number of passengers in each class, third class has the minimum number of survivors at around 25%

This is our key takeaway:

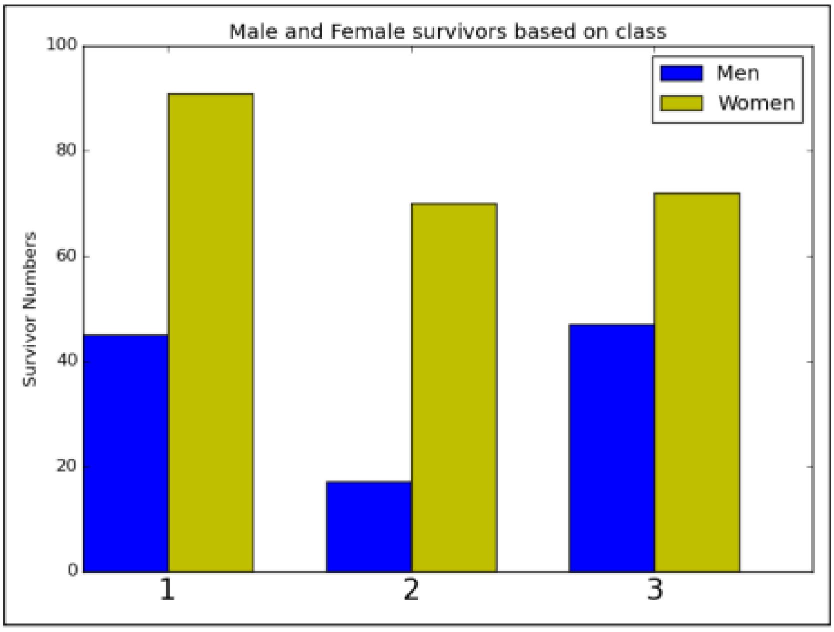
- There was clearly a preference toward saving those from the first class as the ship was drowning. It also had the maximum percentage of survivors

What is the distribution of survivors based on gender among the various classes?

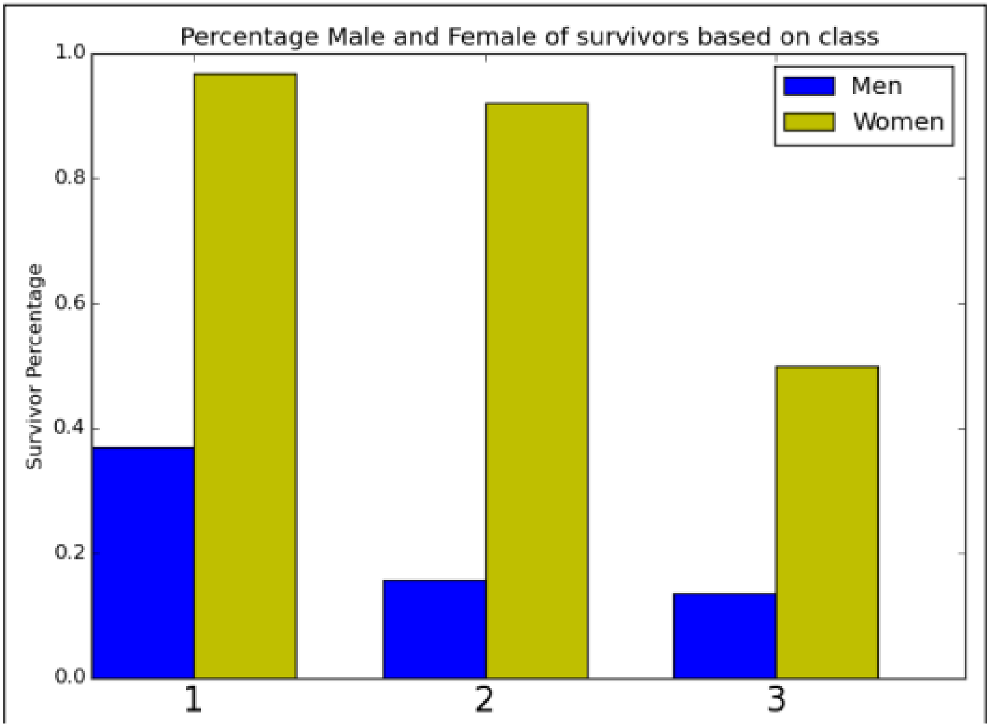
To answer this question, we'll use the following code to plot a side-by-side bar chart to compare the survival rate and percentage among men and women with respect to the class they were in.

```
>>> #Checking for any null values
>>> df['Sex'].isnull().value_counts()
>>> False    891
>>> dtype: int64

>>> # Male Passengers survived in each class
>>> male_survivors = df[df['Sex'] == 'male']
                   .groupby('Pclass')['Survived'].agg(sum)

>>> #Total Male Passengers in each class
>>> male_total_passengers = df[df['Sex'] == 'male']
                          .groupby('Pclass')['PassengerId'].count()
>>> male_survivor_percentage = male_survivors / male_total_passengers

>>> # Female Passengers survived in each class
>>> female_survivors = df[df['Sex'] == 'female']
                     .groupby('Pclass')['Survived'].agg(sum)


>>> #Total Female Passengers in each class
>>> female_total_passengers = df[df['Sex'] == 'female']
                            .groupby('Pclass')['PassengerId'].count()
>>> female_survivor_percentage = female_survivors /
                                 female_total_passengers

>>> #Plotting the total passengers who survived based on Gender
>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)
>>> index = np.arange(male_survivors.count())
>>> bar_width = 0.35
>>> rect1 = ax.bar(index, male_survivors, bar_width, color='blue',
     label='Men')
>>> rect2 = ax.bar(index + bar_width, female_survivors, bar_width,
     color='y', label='Women')
```

```
>>> ax.set_ylabel('Survivor Numbers')
>>> ax.set_title('Male and Female survivors based on class')
>>> xTickMarks = male_survivors.index.values.tolist()
>>> ax.set_xticks(index + bar_width)
>>> xtickNames = ax.set_xticklabels(xTickMarks)
>>> plt.setp(xtickNames, fontsize=20)
>>> plt.legend()
>>> plt.tight_layout()
>>> plt.show()
```



```
>>> #Plotting the percentage of passengers who survived based on Gender
>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)
>>> index = np.arange(male_survivor_percentage.count())
>>> bar_width = 0.35
>>> rect1 = ax.bar(index, male_survivor_percentage, bar_width,
     color='blue', label='Men')
>>> rect2 = ax.bar(index + bar_width, female_survivor_percentage,
     bar_width, color='y', label='Women')
>>> ax.set_ylabel('Survivor Percentage')
>>> ax.set_title('Percentage Male and Female of
     survivors based on class')
>>> xTickMarks = male_survivor_percentage.index.values.tolist()
>>> ax.set_xticks(index + bar_width)
>>> xtickNames = ax.set_xticklabels(xTickMarks)
>>> plt.setp(xtickNames, fontsize=20)
>>> plt.legend()
>>> plt.tight_layout()
>>> plt.show()
```



In the preceding code, the number of male and female survivors is calculated and then a side-by-side bar plot is plotted. After this, the percentage of male and female survivors with respect to the total number of males and females in their respective classes are taken and then plotted.

These are our observations:

- The majority of survivors are females in all the classes
- More than 90% of female passengers in first and second class survived
- The percentage of male passengers who survived in first and third class, respectively, are comparable

This is our key takeaway:

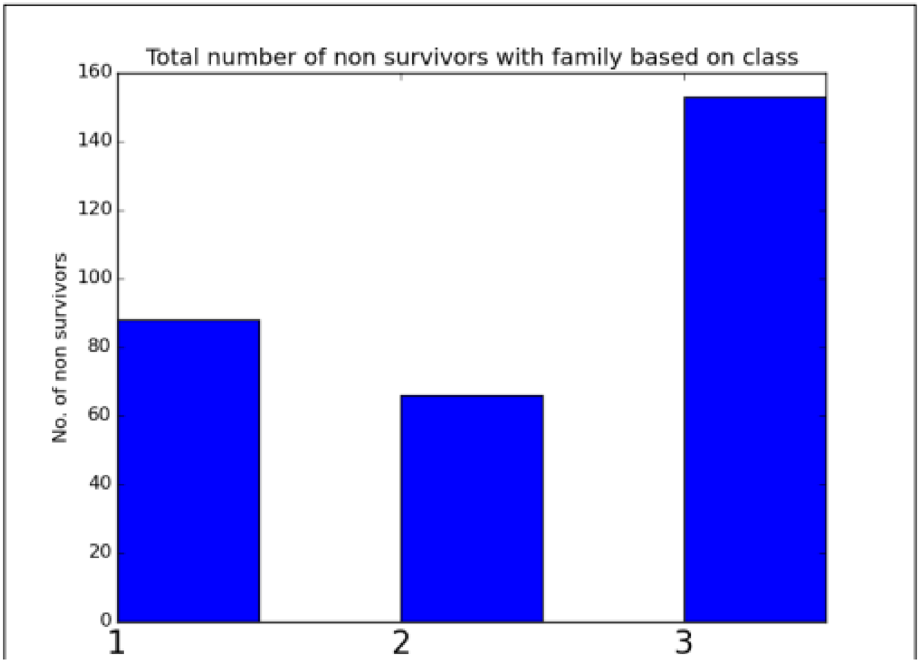
- Female passengers were given preference for lifeboats and the majority were saved.

What is the distribution of nonsurvivors among the various classes who have family aboard the ship?

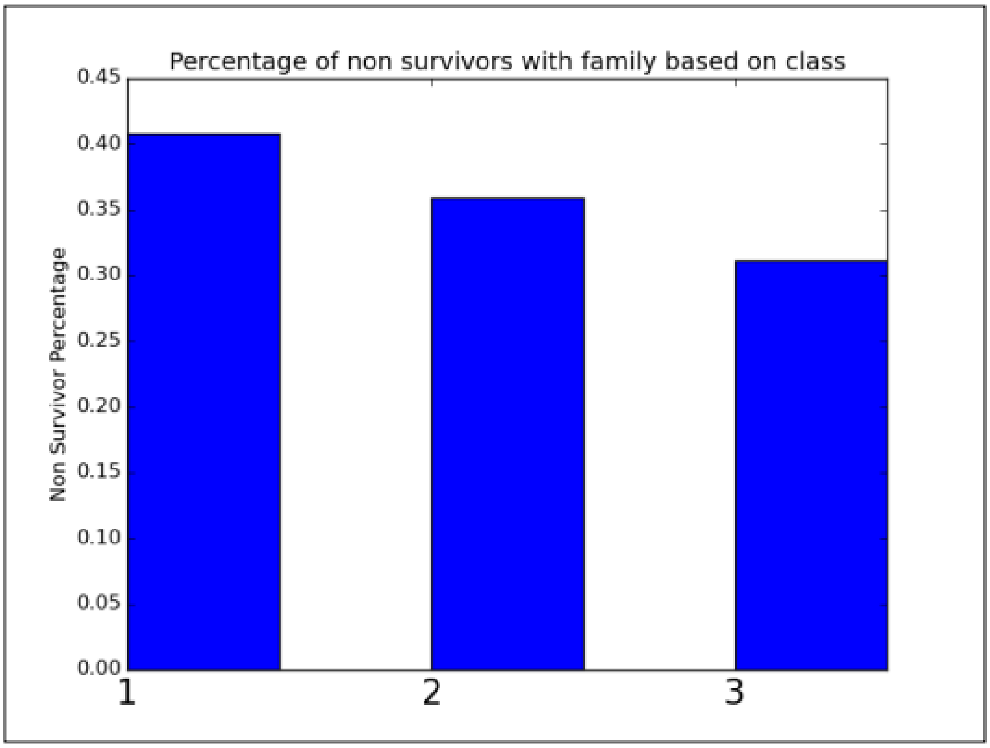
To answer this question, we'll use the following code to plot bar charts again using the total number of nonsurvivors in each class who each had family aboard, and the percentage with respect to the total number of passengers:

```
>>> #Checking for the null values
>>> df['SibSp'].isnull().value_counts()
>>> False    891
>>> dtype: int64

>>> #Checking for the null values
>>> df['Parch'].isnull().value_counts()
>>> False    891
>>> dtype: int64
```

```
>>> #Total number of non-survivors in each class
>>> non_survivors = df[(df['SibSp'] > 0) | (df['Parch'] > 0) &
       (df['Survived'] == 0)].groupby('Pclass')['Survived'].agg('count')
>>> #Total passengers in each class
>>> total_passengers = df.groupby('Pclass')['PassengerId'].count()
>>> non_survivor_percentage = non_survivors / total_passengers
>>> #Total number of non survivors with family based on class
>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)
>>> rect = ax.bar(non_survivors.index.values.tolist(), non_survivors,
       color='blue', width=0.5)
>>> ax.set_ylabel('No. of non survivors')
>>> ax.set_title('Total number of non survivors with
                   family based on class')
>>> xTickMarks = non_survivors.index.values.tolist()
>>> ax.set_xticks(non_survivors.index.values.tolist())
>>> xtickNames = ax.set_xticklabels(xTickMarks)
>>> plt.setp(xtickNames, fontsize=20)
>>> plt.show()
```



```
>>> #Plot of percentage of non survivors with family based on class
>>> fig = plt.figure()
>>> ax = fig.add_subplot(111)
>>> rect = ax.bar(non_survivor_percentage.index.values.tolist(),
     non_survivor_percentage, color='blue', width=0.5)
>>> ax.set_ylabel('Non Survivor Percentage')
>>> ax.set_title('Percentage of non survivors with
                 family based on class')
>>> xTickMarks = non_survivor_percentage.index.values.tolist()
>>> ax.set_xticks(non_survivor_percentage.index.values.tolist())
>>> xtickNames = ax.set_xticklabels(xTickMarks)
>>> plt.setp(xtickNames, fontsize=20)
>>> plt.show()
```



The code here is pretty similar to the code used in the previous questions. Here, we can get the number of the nonsurvivors who have a family and then perform the usual bar plots.

These are our observations:

- There are lot of nonsurvivors in the third class
- Second class has the least number of nonsurvivors with relatives
- With respect to the total number of passengers, the first class, who had relatives aboard, has the maximum nonsurvivor percentage and the third class has the least

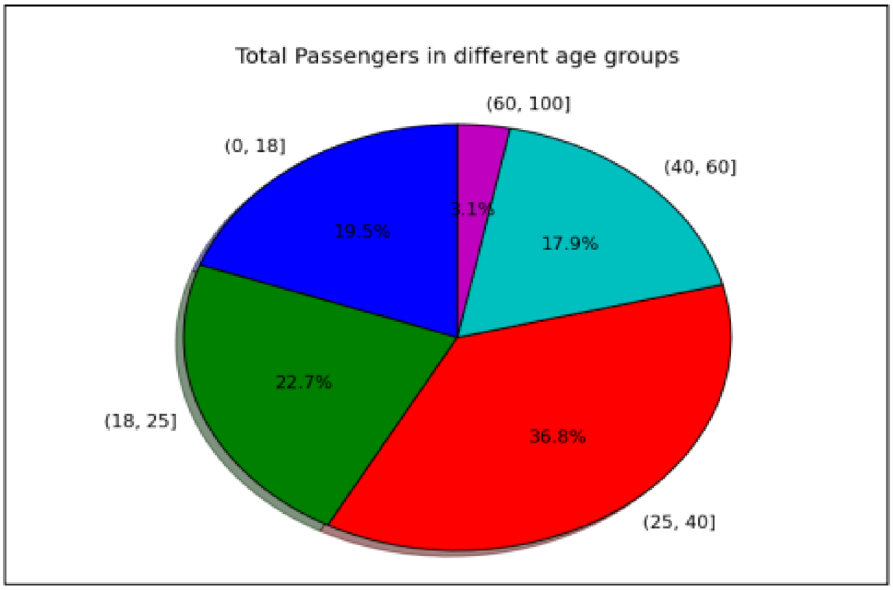
This is our key takeaway:

- Even though third class has the highest number of nonsurvivors with relatives aboard, it primarily had passengers who did not have relatives on the ship, whereas in first class, most of the people had relatives aboard the ship

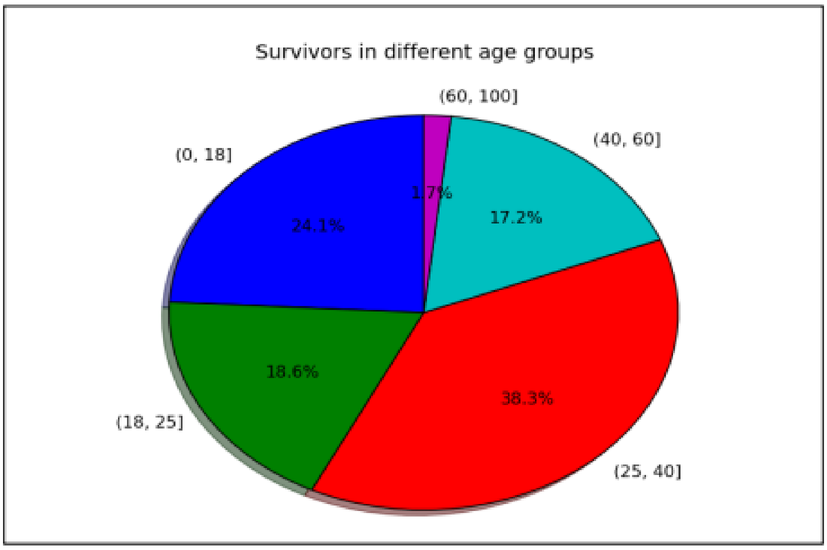
What was the survival percentage among different age groups?

For this question, we'll use the following code to plot pie charts to compare the proportion of survivors in terms of number and percentage with respect to the different age groups:

```
>>> #Checking for null values
>>> df['Age'].isnull().value_counts()
>>> False    714
>>> True     177
>>> dtype: int64

>>> #Defining the age binning interval
>>> age_bin = [0, 18, 25, 40, 60, 100]
>>> #Creating the bins
>>> df['AgeBin'] = pd.cut(df.Age, bins=age_bin)
>>> #Removing the null rows
>>> d_temp = df[np.isfinite(df['Age'])]  # removing all na instances
>>> #Number of survivors based on Age bin
>>> survivors = d_temp.groupby('AgeBin')['Survived'].agg(sum)
>>> #Total passengers in each bin

>>> total_passengers = d_temp.groupby('AgeBin')['Survived'].agg('count')
>>> #Plotting the pie chart of total passengers in each bin
>>> plt.pie(total_passengers,
     labels=total_passengers.index.values.tolist(),
     autopct='%1.1f%%', shadow=True, startangle=90)
>>> plt.title('Total Passengers in different age groups')
>>> plt.show()
```



```
>>> #Plotting the pie chart of percentage passengers in each bin
>>> plt.pie(survivors, labels=survivors.index.values.tolist(),
     autopct='%1.1f%%', shadow=True, startangle=90)
>>> plt.title('Survivors in different age groups')
>>> plt.show()
```



In the code, we defined the bin with the age_bin variable and then added a column called AgeBin, where bin values are filled using the cut function. After this, we filtered out all the rows with the age set as null. After this, we created two pie charts: one for the total number of passengers in each age group and another for the number of survivors in each age group.

These are our observations:

- The 25-40 age group has the maximum number of passengers, and 0-18 has the second highest number of passengers
- Among the people who survived, the 18-25 age group has the second highest number of survivors
- The 60-100 age group has a lower proportion among the survivors

This is our key takeaway:

- The 25-40 age group had the maximum number of survivors compared to any other age group, and people who were old were either not lucky enough or made way for the younger people to the lifeboats.