1. Describe a data project you worked on recently.

I recently worked on OpenStreetMap (OSM) data project. I wanted to parse the raw OSM data of San Jose, CA in XML format to the tabular format for entry into MongoDB database. The data set was large, so I created a random subset of the data first , and later applied the scripts I wrote to full data set

I was particularly interested in node and way tags. Nodes are point features defined by its latitude, longitude and node id. Ways are paths through a city of one kind or another like Street, Avenue, Drive, Boulevard etc.

While auditing the OSM data set for validity, accuracy, completeness, consistency, and uniformity I found that there were several problems with the map. There were Over abbreviated Street Names, Inconsistent and Incorrect postal codes, Inconsistent state name, Inconsistent phone number format, Username error etc. I wrote scripts to fix problems with the data, parsed the data into a tabular format and imported it into the MongoDB database following a schema.

I learned how to clean messy data using Python, source the information into the database and query the database. I was able to draw meaningful insights about the Greater San Jose city and present the results visually in a simple, engaging manner. Moreover working on the large-scale data science project has made me comfortable that I am a valuable addition to a data science team.

---

2. You are given a ten-piece box of chocolate truffles. You know based on the label that six of the pieces have an orange cream filling and four of the pieces have a coconut filling. If you were to eat four pieces in a row, what is the probability that the first two pieces you eat have an orange cream filling and the last two have a coconut filling?

The probability of first orange cream filling is 6/10. The probability of second orange cream filling is 5/9. The probability of third coconut filling is 4/8. The probability of the last coconut filling is 3/7. The probability that the first two pieces you eat have an orange cream filling and the last two have a coconut filling = 6/10 * 5/9 * 4/8 * 3/7 = 0.0714

Follow-up question: If you were given an identical box of chocolates and again eat four pieces in a row, what is the probability that exactly two contain coconut filling?

In this question the order in which chocolates are eaten are irrelevant

The probability that exactly two contain coconut filing = (Number of different ways in which two coconut chocolates can be chosen from four * Number of different ways in which two orange chocolates can be chosen from six)/ Total number of ways four chocolates can be chosen from ten = (4C2 * 6C2)/10C4 = 0.428

---

3. Construct a query to find the top 5 states with the highest number of active users. Include the number for each state in the query result.

```
SELECT state, SUM(active)
FROM users
GROUP BY state
ORDER BY SUM(active) DESC
LIMIT 5;
```

---

4. Define a function first_unique that takes a string as input and returns the first non-repeated (unique) character in the input string. If there are no unique characters return None.

Initialize an array of size 256 for holding the count frequency of characters in the string. Scan the string from left to right and update the count frequency in the array. Scan the string once again to search for the first character with count frequency of 1 and return it.

Space Complexity: O(1) Time Complexity: 0(N), where N is the length of string.

```
def count_array(string):
  '''Returns an array of size 256 containing count of the characters in the string.'''
  array = [0] * 256
  for i in string:
    array[ord(i)] += 1
  return array

def first_unique(string):
  '''Input: String
  Output: The first non-repeated (unique) character in the input string.'''
  count = count_array(string)
  unique_char = None
  for i in string:
    if count[ord(i)] == 1:
      unique_char = i
      break
  return unique_char
```

---

5. What are under fitting and over fitting in the context of Machine Learning? How might you balance them?

Nobel laureate Ronald Coase said, "If you torture the data long enough, it will confess."Over fitting occurs when data mining procedures perform too well on the training data set, however, fails to generalize on unseen data sets. As the model becomes more complex it tends to pick up spurious correlations (noise) in the data that are not the characteristics of the population in general. For instance in a tree-based model well, if we allow the tree to grow until the leaves are pure, we increase the complexity of the model drastically. The model in these cases is purely based on memorization. In case of models based on mathematical functions, we make the model more complex by adding more features/attributes. Now if we examine the accuracy of the model on the data it was trained on it will be very good. However, will the model generalize? Possibly No. In order to deal with over fitting, we have to first recognize it. To identify over fitting we need to evaluate the performance of the model on unseen data- the data model was not trained on. It is an essential practice in data mining procedures to keep a subset of data as holdout data- test data. We train our model on training data and examine the generalization performance of the model on the test data. We hide the label for target variable of the test data from the model and let the model predict the values for target variable. Then we compare the values predicted by the model with the hidden true values. We can also use a more sophisticated holdout training and testing procedure called Cross-validation.

In order to avoid over fitting, we need to control for the complexity of the model. This process is called model regularization. We can reduce the complexity of the model by pruning the classification tree (cutting the tree back when it becomes too large), limiting the number of features used, and including explicit complexity penalties into the mathematical functions used for modeling.

Under fitting occurs when the model performs well neither on the training data set nor generalize well to the unseen data sets. The under fitting model will have poor performance on the training data. It is because the model is too simple and the input features are not expressive enough to describe the target variable very well. To increase the model flexibility we can add new domain-specific features, decrease the amount of regularization used, and try alternate machine learning algorithms.

Ideally, we want to balance over fitting and under fitting and select a sweet spot between them. To understand this it will be best to plot the complexity of the model against the accuracy of the model. As we increase the complexity of the model, the accuracy of the model increases on both training and holdout data. If we train for too long, the performance on training data continues to increase because the model is over fitting. At the same time, the performance on the holdout data starts to decrease as the model's ability to generalize decreases. There is no one way to determine the exact sweet spot theoretically, so we have to rely on empirical approaches.

The models that tend to over fit have low bias and high variance. Non-parametric machine learning algorithms like Decision Tree, kNN often have a low bias, but high variance. The models that tend to under fit have high bias and low variance. Parametric machine learning algorithms like Linear Regression, Log Regression, and Linear Discriminant function often have a high bias, but low variance.

The goal of the supervised classification algorithm is to achieve low bias and low variance.

---

6. If you were to start your data analyst position today, what would be your goals a year from now?

I am really excited about the idea of translating my technical and problem-solving skills into products/insights that can have a lasting impact on the business. I am also passionate about using those insights to sell subscriptions and make the experience of reading The Wall Street Journal better.

My goal for the first three months will be to be:

1. Make myself comfortable with the company's ongoing business intelligence efforts, and large-scale data science projects.
2. Source and query customer database, transaction database and marketing response database.
3. Gather detailed intelligence on everything from when, how often, and where customers are using products.
4. Learn about the customer metrics that Dow Jones tracks.

We can move onto different avenues once I am comfortable with the team and ongoing operations. In order to determine the exact goal for a year – I will need to understand the exact business problem the team is trying to solve and decide whether a data science solution can be appropriately formulated to solve this business problem. Some of the possible directions we can look into are:

1. Identify new customer segments similar to the current best business customers, and target sales on the best opportunities. In order to do that we can use similarity matching based on "firmographic"(Firmographics are sets of characteristics to segment prospect organizations. What demographics are to people, firmographics are to organizations.) data describing characteristics of the companies.

2. Do our customers form natural clusters? This can, in turn, be used to aid decision-making process such as: What products should we offer? How should our customer care team be structured?

3. Co-occurrence grouping finds associations between products based on transactions involving them. Which of the Dow Jones products are commonly purchased together?

I envision myself as a data scientist and an invaluable asset to the team who constantly puts himself out of comfort zone. I understand lifelong learning will be an important part of realizing these goals. My learning goals are -

1. Learn to apply predictive models to massive data sets. Successfully develop and deploy machine learning applications.
2. Enroll and complete Machine Learning and Deep Learning Nanodegree programs from Udacity.