1. You are tasked with migrating a legacy website to Sitecore that contains over 100GB of image assets. How would you migrate and store the images? These images are all required to improve customer experience.

This is a two part answer. Firstly we will tackle the migration of the images themselves. Any way we migrate that amount of images will take a fairly long amount of time. There is a feature in Sitecore Media Library which allows you to unpack a zip file into the media library, but given the amount of images, I would strongly suggest against using the browser to do this. My recommendation would be to use the uploads folder in the webroot on the server which can be configured in the Sitecore settings. This feature will allow you to place all of your images in this folder and let Sitecore do the work in loading them into the media library.

The second part is whether or not you want to store the images as blobs in the database, or alternatively configure Sitecore to store the images on the server. I wouldn’t recommend storing that amount of images in the database, so in this case my recommendation would be to use server storage, but this of course can be problematic with syncing over multiple servers. So if it is available to you, I would actually recommend Azure Blob storage for the best performance for images.

1. One of your clients has a requirement to create multiple forms in Sitecore (Contact Us, Book an adviser, etc.). What is your recommendation for this?

Depending on your version of Sitecore, I would recommend using one of Sitecore’s form products. Prior to Sitecore 9, the option is to use Web Forms For Marketers. For Sitecore 9, WFFM was replaced with Sitecore Forms. So rather than custom building forms, I would use one of these products to set up the forms to allow easy configuration of forms, form events, tracking and notifications.

1. Your website collates data from multiple systems to display on screen. You are tasked with building a search that indexes the data from all these separate system to display accurate search results. What are our options?

There are a couple of options. The first is the use of Solr as an indexing platform as it allows for fast queries and scalability.

You could also use Azure search however, for Sitecore, it does have limitations as to how many indexes you can store on certain azure search packages but does allow for some neat features like real time search suggestions much like a search engine.

There is also Lucene indexing, but this is not recommended as this is not scalable and all the indexes are stored in files on a server.

1. You are tasked to create show/hide the tiles on the dashboard based on member's age. Member's age is available through member's API. What are the high level steps you will carry out to accomplish this?
   * Create a service method to fetch the dashboard tile class objects from a CMS/Datasource
   * Create another service method to call the members api and fetch the members age.
   * Link the first method to the second method and filter out all the tiles which don’t meet the members age requirements
   * Create a controller method for the dashboard tiles view and call the service method to get the tile class objects
   * Pass the tile results to the container view
   * Create a container view for the tiles on the dashboard
   * Create a partial view which will be used to display each tile on the dashboard
   * In the container view loop through the tiles in the tile results and for each tile call the partial view to show the tile in the container