Technical Assessment

1) Define the database tables with its corresponding data types and relationships for the following case:
   A customer wants to sell his car so we need the car information (car year, make, model, and submodel), the location of the car (zip code) and the list of buyers that are willing to buy the car in that zip code. Each buyer will have a quote (amount) and only one buyer will be marked as the current one (not necessarily the one with the highest quote). We also want to track the progress of the case using different statuses (Pending Acceptance, Accepted, Picked Up, etc.) and considering that we care about the current status and the status history (previous statuses indicate when it happened, who changed it, etc.). The status “Picked Up” has a mandatory status date but the rest of the statuses don’t.
   Write a SQL query to show the car information, current buyer name with its quote and current status name with its status date. Do the same thing using Entity Framework. Make sure your queries don’t have any unnecessary data.

2) What would you do if you had data that doesn’t change often but it’s used pretty much all the time?

3) Analyze the following method and make changes to make it better. Explain your changes.

```
public void UpdateCustomersBalanceByInvoices(List<Invoice> invoices)
{
        foreach (var invoice in invoices)
        {
                var customer =
dbContext.Customers.SingleOrDefault(invoice.CustomerId.Value);
                customer.Balance -= invoice.Total;
                dbContext.SaveChanges();
        }
}
```

4) Implement the following method using Entity Framework, making sure your query is efficient in all the cases (when all the parameters are set, when some of them are or when none of them are). If a “filter” is not set it means that it will not apply any filtering over that field (no ids provided for customer ids it means we don’t want to filter by customer).

```
public async Task<List<OrderDTO>> GetOrders(DateTime dateFrom, DateTime dateTo,
List<int> customerIds, List<int> statusIds, bool? isActive)
{
    // your implementation
}
```

5) Bill, from the QA Department, assigned you a high priority task indicating there's a bug when someone changes the status from "Accepted" to "Picked Up".
   Define how you would proceed, step by step, until you create the Pull Request.