Utilization Of Algorithms, Dynamic Programming, Optimal Memory Utilization

In this activity you are expected to develop & submit the developed code by testing it.

Utilizing algorithms, dynamic programming, and optimal memory utilization in a Salesforce application typically involves implementing efficient data processing and management techniques. While Salesforce development is primarily focused on the Apex programming language, complex algorithmic implementations may not always be necessary for typical Salesforce use cases. However, you can optimize code for memory utilization and performance. Here's an example of optimizing memory utilization in Salesforce:

```
// Class Description: This class demonstrates optimal memory utilization in Salesforce


public class MemoryUtilizationExample {


    // Method to demonstrate optimal memory utilization
    public static void optimizeMemoryUtilization() {
        // Initialize a list to store a large amount of data
        List<String> largeDataList = new List<String>();

        // Add elements to the list
        for (Integer i = 0; i < 1000000; i++) {
            largeDataList.add('Data_' + String.valueOf(i));
        }

        // Process the data in batches to optimize memory usage
        Integer batchSize = 1000;
        Integer numberOfBatches = largeDataList.size() / batchSize;

        for (Integer j = 0; j < numberOfBatches; j++) {
            List<String> batchData = new List<String>();
            for (Integer k = 0; k < batchSize; k++) {
                batchData.add(largeDataList[j * batchSize + k]);
            }
            // Process the batch data here
            // Example: You can perform database operations or other data manipulations on the batch data
        }
    }
}
```