Day 12  27/06/2025

**Data Structures & Algorithms**

**Task 1**: What do you understand about data structures?

Data structures in common terms means organizing and storing data in an efficient way. If we can do this efficiently this will make our lives easier. By using data structures we can make the code faster, efficient and scalable.

**Task 2**: What are the types of data structures you know .. list them out..

Linear Data structures:

1. Arrays
2. Linked Lists
3. Stacks
4. Queues

Non-Linear Data structures

1. Trees
2. Graphs

**Task 3:** What all operations can we do in Data structures?

Core operations include insertion, deletion, traversal, searching, sorting, and merging

**Task 4**: What are static and dynamic arrays? Explain or summarize key points in a table like

Size, performance, memory, flexibility, limitations

|  |  |  |
| --- | --- | --- |
| Features | Static Arrays | Dynamic Arrays |
| Size | Size is fixed, should be defined during compiling | Size is variable, can be changed in runtime |
| Performance (Access) | Faster | Slower |
| Performance (Insertion/Deletion) | Slower | Faster |
| Memory | Stack memory, it is fixed.  Memory can be wasted if allocated memory is not used | Heap memory, it is dynamic.  Here memory is allocated on demand |
| Flexibility | Less Flexible | More flexible |
| Limitations | Cannot handle datasets which exceeds the pre defined size | Resizing can be done, but it is time consuming |

**Task 5**: What is the binary value of a?

01100001

Hint ascii value is 97..

**Task 6:** Types of Computer memory with examples.. Explain ..

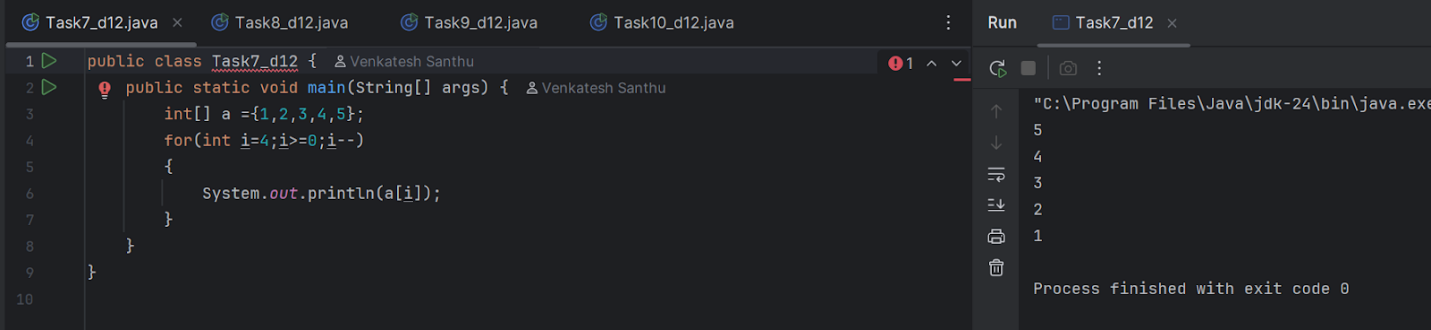
1. RAM (Random Access Memory): It is volatile, the data is lost once the system is shut down or powered off.
2. DRAM (Dynamic RAM): Requires periodic refreshing to maintain data, commonly used in PCs
3. SRAM (Static RAM): Does not require refreshing, often used in cache memory, offering faster access speeds.

2. ROM (Read only Memory): It is not volatile, the contents are there even if the system is shut down. Basically ROM stores data essential for the system operations.

3. Cache Memory: A smaller, faster type of memory that stores frequently accessed data, enabling the CPU to retrieve information quickly.

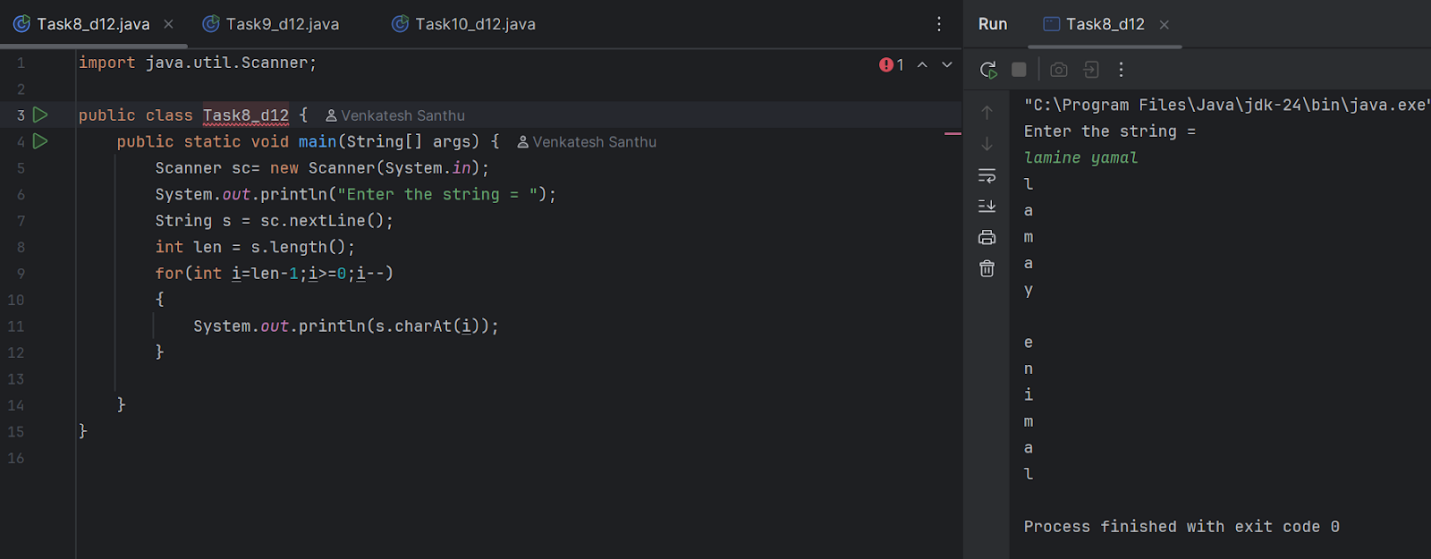
**Task 7:** Reverse an array. write a code.

Hint : take a list of nos and display in reverse order..



**Task 8:** Reverse a string .. write a code

Hint: take a name from the user and display the name in reverse order..



**Task 9**: Create hackerrank, leetcode account if not already done.

**Task 10**: Understand the below code

public class Task010 {

   public static void main (String[] args) {

       int[] arr1 = {11, 34, 66, 75};

       int n1 = arr1.length;

       int[] arr2 = {1, 5, 19, 50, 89, 100};

       int n2 = arr2.length;

       int[] merge = new int[n1 + n2];

       int i = 0, j = 0, k = 0, x;

       System.*out*.print("Array 1: ");

       for (x = 0; x < n1; x++)

           System.*out*.print(arr1[x] + " ");

       System.*out*.print("\nArray 2: ");

       for (x = 0; x < n2; x++)

           System.*out*.print(arr2[x] + " ");

       while (i < n1 && j < n2) {

           if (arr1[i] < arr2[j])

               merge[k++] = arr1[i++];

           else

               merge[k++] = arr2[j++];

       }

       while (i < n1)

           merge[k++] = arr1[i++];

       while (j < n2)

           merge[k++] = arr2[j++];

       System.*out*.print("\nArray after merging: ");

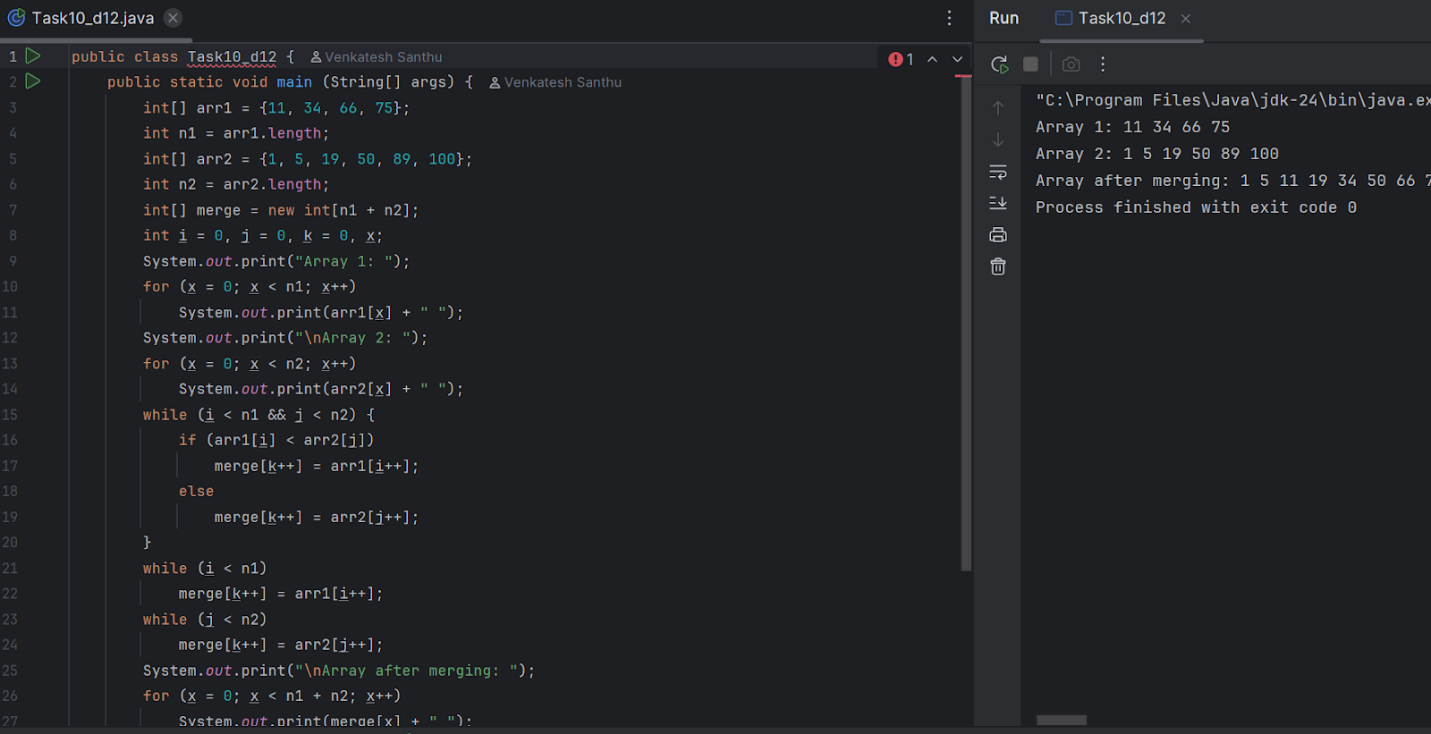
       for (x = 0; x < n1 + n2; x++)

           System.*out*.print(merge[x] + " ");

   }

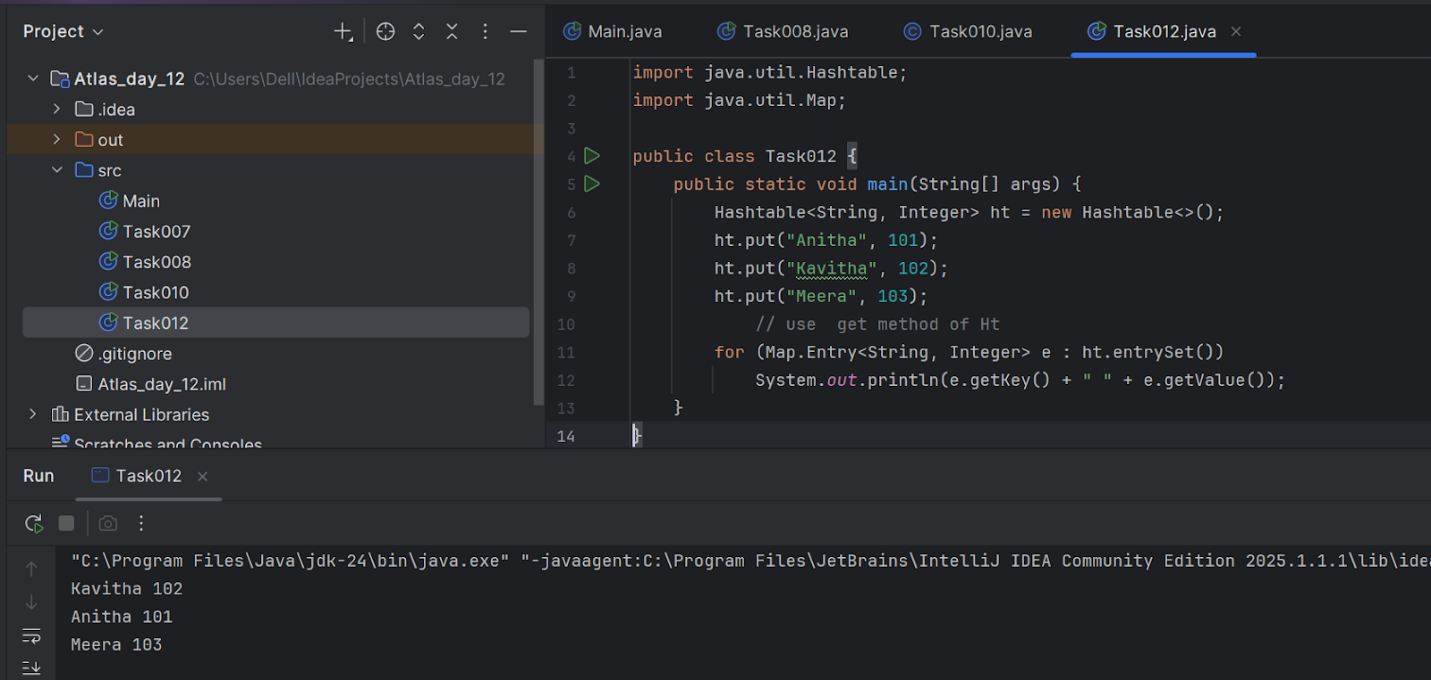
}

Home task : Task 10  rewrite the code in such a way that it has to take unsorted list and then ,merge in an array the sorted list.

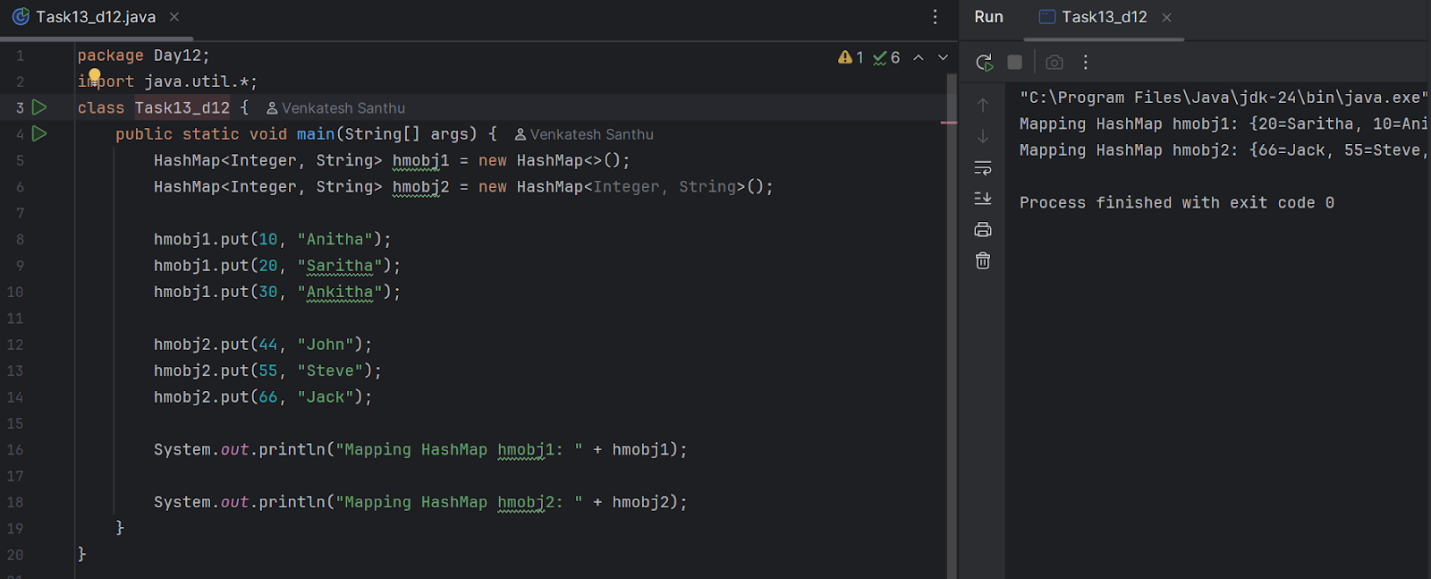


**Task 11**: What do you understand about the Hash table?

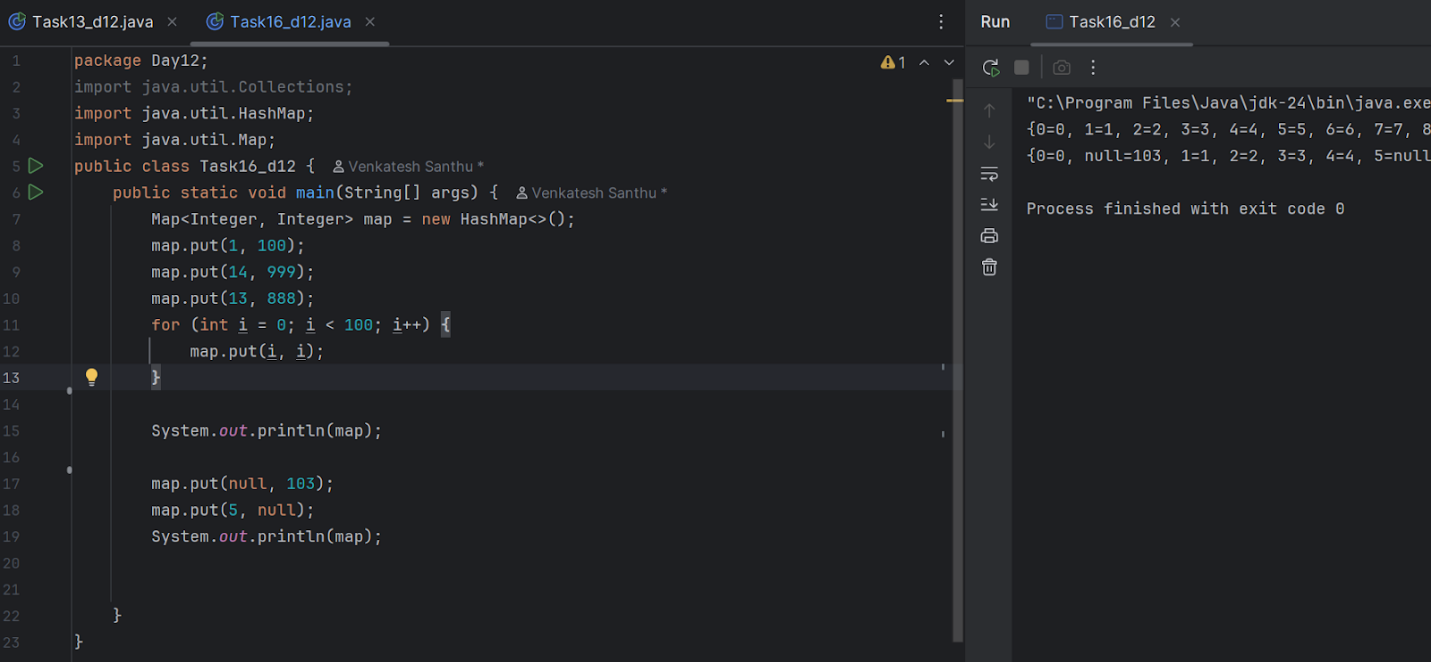
**Task 12**: Understand the below Hash table code and try to print values using get method of Hash table



**Task 13**:



**Task 16**:



**Task 17**:

