**Algorithms and Codes**

1. 5 Employees data should be store in array(ID,Name,Address) and display data value

**Algorithm:**

1. Declare object named Employee with variables ID, Name and Address.
2. Create an array of objects with object type Employee
3. Store the values of 5 employees’ data in array objects
4. Run a loop upto 5 numbers and display objects one by one.

**Code:**

class Employee {

public int employeeID;

public String employeeName;

public String employeeAddress;

Employee(int employeeID, String employeeName, String employeeAddress)

{

this.employeeID = employeeID;

this.employeeName = employeeName;

this.employeeAddress = employeeAddress;

}

}

public class Demo {

public static void main(String args[])

{

Employee[] arr;

arr = new Employee[5];

arr[0] = new Employee(1701289270, "Tukaram", "Demo Address1");

arr[1] = new Employee(1701289219, "Om Prasad", "Demo Address2");

arr[2] = new Employee(1701289265, "Jai Bhatt", "Demo Address3");

arr[3] = new Employee(1701289221, "Farhan Qureshi", "Demo Address4");

arr[4] = new Employee(1701289213, "Raju Rastogi", "Demo Address5");

for(int i=0;i<5;i++)

{

System.out.println("Employee ID: "+arr[i].employeeID+", Name: "+arr[i].employeeName+ ", Address: "+ arr[i].employeeAddress);

}

}

}

1. Sort using bubble sort based based on its employeeid

**Algorithm:**

1. The concept of bubble sort here will work by comparing 2 consequent Employee IDs
2. If the first employee ID is bigger than the second, swap objects 1 and 2.
3. If not, let them be and compare employee ID of objects 2 and 3.
4. At the end of each iteration of this loop, the biggest employeeID will be placed at the bottom-most position in the array.
5. Do this for all the objects. At the end, all the objects will be sorted according to employee ID’s

**Code:**

Employee(int employeeID, String employeeName, String employeeAddress)

{

this.employeeID = employeeID;

this.employeeName = employeeName;

this.employeeAddress = employeeAddress;

}

}

public class Demo {

public static void main(String args[])

{

// Declaring an array of student

Employee[] arr;

arr = new Employee[5];

arr[0] = new Employee(1701289270, "Satyabrata", "Demo Address1");

arr[1] = new Employee(1701289219, "Omm Prasad", "Demo Address2");

arr[2] = new Employee(1701289265, "Jai Bhatt", "Demo Address3");

arr[3] = new Employee(1701289221, "Farhan Qureshi", "Demo Address4");

arr[4] = new Employee(1701289213, "Raju Rastogi", "Demo Address5");

for(int i=0;i<5;i++)

{

System.out.println("Employee ID: "+arr[i].employeeID+", Name: "+arr[i].employeeName+", Address: "+ arr[i].employeeAddress);

}

//bubblesort

for(int i=0;i<5;i++)

{

Employee temp = new Employee(0, "", "");

for(int j=1;j<(5-i);j++)

{

if(arr[j-1].employeeID>arr[j].employeeID)

{

//swapping

temp.employeeID = arr[j-1].employeeID;

temp.employeeName = arr[j-1].employeeName;

temp.employeeAddress = arr[j-1].employeeAddress;

arr[j-1].employeeID = arr[j].employeeID;

arr[j-1].employeeName = arr[j].employeeName;

arr[j-1].employeeAddress = arr[j].employeeAddress;

arr[j].employeeID = temp.employeeID;

arr[j].employeeName = temp.employeeName;

arr[j].employeeAddress = temp.employeeAddress;

}

}

}

System.out.println("After Bubble Sort:");

for(int i=0;i<5;i++)

{

System.out.println("Employee ID: "+arr[i].employeeID+", Name: "+arr[i].employeeName+", Address: "+ arr[i].employeeAddress);

}

}

}

class Employee {

public int employeeID;

public String employeeName;

public String employeeAddress;

1. Sort using merge sort based on its employeeid

**Algorithm:**

1. Merge sort here will work in 2 phases, divide and merge.
2. The divide phase will divide/separate the array items one by one into individual elements/objects.
3. Then in merge phase, Compare atomic items one by one.
4. Comparing in pairs and sorting internally will create pairs of sorted objects
5. Then comparing items in pairs with other pairs, and combining these pairs, creating next level of sorted blocks further.
6. This way once all the items in the array are sorted and merged, this will lead to the entire array being sorted.

**Code:**

class Employee {

public int employeeID;

public String employeeName;

public String employeeAddress;

// Student class constructor

Employee(int employeeID, String employeeName, String employeeAddress)

{

this.employeeID = employeeID;

this.employeeName = employeeName;

this.employeeAddress = employeeAddress;

}

}

class Merge

{

void merge(Employee arr[], int begin, int middle, int end)

{

int p, q, r;

int n1 = middle - begin + 1;

int n2 = end - middle;

Employee LeftArray[] = new Employee[n1];

Employee RightArray[] = new Employee[n2];

for (p = 0; p < n1; p++)

LeftArray[p] = arr[begin + p];

for (q = 0; q < n2; q++)

RightArray[q] = arr[middle + 1 + q];

p = 0; /\* initial index of first sub-array \*/

q = 0; /\* initial index of second sub-array \*/

r = begin; /\* initial index of merged sub-array \*/

while (p < n1 && q < n2)

{

if(LeftArray[p].employeeID <= RightArray[q].employeeID)

{

arr[r] = LeftArray[p];

p++;

}

else

{

arr[r] = RightArray[q];

q++;

}

r++;

}

while (p<n1)

{

arr[r] = LeftArray[p];

p++;

r++;

}

while (q<n2)

{

arr[r] = RightArray[q];

q++;

r++;

}

}

void mergeSort(Employee arr[], int begin, int end)

{

if (begin < end)

{

int middle = (begin + end) / 2;

mergeSort(arr, begin, middle);

mergeSort(arr, middle + 1, end);

merge(arr, begin, middle, end);

}

}

}

public class Demo {

public static void main(String args[])

{

// Declaring an array of Employee

Employee[] arr;

arr = new Employee[5];

arr[0] = new Employee(1701289270, "Satyabrata", "Demo Address1");

arr[1] = new Employee(1701289219, "Omm Prasad", "Demo Address2");

arr[2] = new Employee(1701289218, "Jai Bhatt", "Demo Address3");

arr[3] = new Employee(1701289221, "Farhan Qureshi", "Demo Address4");

arr[4] = new Employee(1701289213, "Raju Rastogi", "Demo Address5");

for(int i=0;i<5;i++)

{

System.out.println("Employee ID: "+arr[i].employeeID+", Name: "+arr[i].employeeName+", Address: "+ arr[i].employeeAddress);

}

Merge m1 = new Merge();

m1.mergeSort(arr,0,5-1);

System.out.println("After Merge Sort:");

for(int i=0;i<5;i++)

{

System.out.println("Employee ID: "+arr[i].employeeID+", Name: "+arr[i].employeeName+", Address: "+ arr[i].employeeAddress);

}

}

}