|  |
| --- |
| DSA Minor-1 Project |
| Subhranshu Sekhar Choudhury  SEC - 45  **REGD NO: 2241003051**  [Website] |

Description

The program creates employee[] array, which will hold at most 500 employee. From the user we will get the details using Scanner class. We have created MyDate, Strings, Employee, Address class for storing with our defined datatype. MyDate will store the date in Date class format.

String will store all the string (print lines) used in our program. Employee class will used to make array of employees.

* The first method, **arrangeEmployeeBySalary**, sorts an array of Employee objects in descending order by their salary field. It uses the Arrays.sort method with a custom comparator that handles null values and compares the salaries of two Employee objects using the Double.compare method.
* The second method, **getEmployeesByJobPosition**, prints the details of all Employee objects in an array that have a specific job position. It uses a for-each loop to iterate over the array and checks if the employee object is not null and has the same job position as the parameter. If so, it prints the employee object using the System.out.println method.
* The third method, **getEmployeesByHireDate**, prints the details of all Employee objects in an array that have a hire date between two given dates. It uses a try-catch block to convert the parameters of type Date to java.util.Date objects. Then, it uses a for-each loop to iterate over the array and checks if the employee object is not null and has a hire date that is after the start date and before the end date. If so, it prints the employee object using the System.out.println method.
* The fourth method, **getEmployeesBySalary**, prints the details of all Employee objects in an array that have a salary between two given values. It uses a for-each loop to iterate over the array and checks if the employee object is not null and has a salary that is greater than or equal to the lower bound and less than or equal to the upper bound. If so, it prints the employee object using the System.out.println method.
* The fifth method, **foreignEmployeeCount**, returns the number of Employee objects in an array that have a foreign contact number. It uses a for-each loop to iterate over the array and checks if the employee object is not null and has a contact number that does not start with “+91”. If so, it increments a count variable. Finally, it returns the count variable.

Subhranshu Choudhury  
2241003051