**שיעורי בית – Java Interface**

**חלק א'**

כתבו מחלקה עובד Employee ע"פ הדרישות הבאות:

1. Attributes – מאפיינים
   * name: String
   * age: int
   * salary: double
2. Methods – מתודות
   * Employee(name,age, salary)
   * toString()

בצעו את השינויים הנדרשים והשתמשו בממשק Comarable כדי שנוכל לבצע מיון של עובדים לפי השכר.

**public** **static** **void** main(String[] args) {

Employee[] employees = **new** Employee[] {

**new** Employee("Kobi", 28,8000),

**new** Employee("Dan", 29, 6000),

**new** Employee("Taly", 23, 7000)

};

*sortEmployeesBySalary*(employees);

}

**private** **static** **void** sortEmployeesBySalary(Employee[] employees) {

System.***out***.println("before sorting by salary");

*print*(employees);

Arrays.*sort*(employees);

System.***out***.println("\nafter sorting by salary");

*print*(employees);

}

**private** **static** **void** print(Employee[] employees) {

**for** (Employee employee: employees) {

System.***out***.println(employee);

}

}

הפלט הנדרש:

before sorting by salary

Kobi,28, 8000.0

Dan,29, 6000.0

Taly,23, 7000.0

after sorting by salary

Dan,29, 6000.0

Taly,23, 7000.0

Kobi,28, 8000.0

**חלק ב'**

נמשיך עם המחלקה עובד Employee, הרחיבו את התמיכה במיון כך שיתמוך גם במיון לפי גיל.

בצעו את השינויים הנדרשים והשתמשו בממשק Comarator בכדי לתמוך במיונים שונים

System.***out***.println("before sorting");

*print*(employees);

Arrays.*sort*(employees, **new** AgeComparator());

System.***out***.println("\nafter sorting by age");

*print*(employees);

Arrays.*sort*(employees, **new** SalaryComparator());

System.***out***.println("\nafter sorting by salary");

*print*(employees);

הפלט הנדרש:

before sorting

Kobi,28, 8000.0

Dan,29, 6000.0

Taly,23, 7000.0

after sorting by age

Taly,23, 7000.0

Kobi,28, 8000.0

Dan,29, 6000.0

after sorting by salary

Dan,29, 6000.0

Taly,23, 7000.0

Kobi,28, 8000.0

**חלק ג'**

השתמשו במיון הבינארי של Arrays וחפשו את אחד העובדים לפי שם העובד.  
הוסיפו ממשק Comarator נוסף בכדי לתמוך בחיפוש.

Employee e1 = **new** Employee("Taly", 23, 7000);

Comparator<Employee> c = **new** NameComparator();

Arrays.*sort*(employees, c);

**int** index = Arrays.*binarySearch*(employees, e1, c );

System.***out***.println("Employee " + employees[index] + " found in index " + index);

הפלט הנדרש:

Employee Taly,23, 7000.0 found in index 2

**חלק ד'**

הוסיפו ממשק Cloneable בכדי לתמוך בשכפול עובד.

Employee e = employee.clone();

employee.setName("Kobi change");

System.***out***.println("Original employee : " + employee);

System.***out***.println("Clone employee : " + e);

הפלט הנדרש:

Original employee : Kobi change,28, 8000.0

Clone employee : Kobi,28, 8000.0