**משימה – JAVA API**

**מטרה**

במשימה זו נתרגל תוכנה מודרנית עם APIs נפוצים בשפת Java

**דרישות**

* עליכם ליצור מחלקה עם מתודה ראשית main()
* במתודה main יש להדפיס אובייקטים מהמחלקות השונות
* אין צורך לבצע בקרת קלט, הניחו כי הקלט תקין

**שלב 3**

בשלב זה נתרגל את האלמנטים הבאים:

* עבודה נכונה עם Enum
* עבודה נכונה עם Exceptions
* עבודה נכונה עם Time Api

ממשו את ה-Enum **Error** באופן הבא:

1. Enum Values

*NULL\_OR\_EMPTY*("the list is null or empty"),  
*EMPLOYEE\_NOT\_FOUND*("the given employee not found"),  
*EMPLOYEE\_ALREADY\_EXIST*("the given employee is already exist");

1. Attributes – מאפיינים
   * - **final** message: String
2. Constructors – בנאים
   * **-** Error(message: Message)
3. Methods – מתודות
   * + getMessage(): String
     + Return the message

עדכנו את המחלקה **Helper** באופן הבא:

1. Methods – מתודות
   * + static **String** format(time: LocalDateTime): String
     + Return a format String for e.g.: 19/03/2022 – 15:55:22

ממשו את המחלקה **CompanySystemException** באופן הבא:

1. Class Level
   * **Extends** Exception
2. Attributes – מאפיינים
   * No needed
3. Constructors – בנאים
   * + CompanySystemException(msg: String)
     + Send the super constructor the next message:

{msg} has occurred in {formatted current LoaclDateTime }

1. Methods – מתודות
   * No needed

עדכנו את המחלקה **Company** באופן הבא:

1. Attributes – מאפיינים
   * No changes
2. Constructors – בנאים
   * No changes
3. Methods – מתודות
   * + addEmployee(employee: Employee): void
     + It the given employee already exist **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Add the given employee to the collection
   * + removeEmployee (employee: Employee): void
     + It the given company is not exist **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Remove the given company from the collection
   * + getAverageSalaray(): double
     + If the collection is null or empty **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Return the average salary
   * + getManagementAverageSalaray(): double
     + If the collection is null or empty **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Return the average salary of Managers in the company
   * + getMonthlyPayment(): double
     + If the collection is null or empty **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Return the sum of the salaries
   * + getYearlyPayment(): double
     + If the collection is null or empty **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Return the sum of the salaries per 1 year
   * + getTotalNumOfEmployees(): double
     + If the collection is null or empty **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Return the number of employees
   * + getTotalNumOfManagers (): double
     + If the collection is null or empty **throw** **CompnaySystemExcetion** with the relevant Error string using the Enum Error we created previous
     + Return the number of employees which are managers

עדכנו את המחלקה **Helper** באופן הבא:

עדכנו את המתודה report – קיימת זריקה אפשרית של חריגה באמצעות throws declaration

ממשו את המחלקה **Test** באופן הבא:

עדכנו את המתודה main קיימת זריקה אפשרית של חריגה באמצעות throws declaration

**בהצלחה!**