

המכללה האקדמית צפת, מערכות מידע ניהוליות

פרויקט: סוף קורס Java מתקדם

תשפ"ה - 2025

מרצה: פרופ' מאלכ יוסף

מתרגל: מר מוחמד נאסר

## תכנות מתקדם ב-JAVA

### פרויקט סוף קורס

#### מערכת לניהול רכבים

מגישים:

פז ויקטור מאור

בן-ציון אפוטה

מערכות מידע ניהוליות

שנה ב' - סמסטר א'

18 בפברואר 2025

המכללה האקדמית צפת

קריירה עם משמעות



## ראשי פרקים

1.	מבוא	2
2.	פרק א' - תפריטים	3
2.1.	ההירארכיה של התפריט	3
2.2.	תפריט מנהל (Admin Menu)	4
2.3.	תפריט משתמש (User Menu)	5
3.	פרק ב' - מבנה המערכת	6
3.1.	הקבצים המרכזיים	6
3.2.	User.java	7
3.3.	AdminUser.java	8
3.4.	RegularUser.java	9
3.5.	Vehicle.java	10
3.6.	VehicleOpertations.java	11
3.7.	VehicleManager.java	12
3.8.	Rental.java	15
3.9.	Logger.java	16
3.10.	InputValidator.java	17
3.11.	MenuManager.java	18
3.12.	Main.java	19
4.	פרק ג' - תהליכי המערכת	20
4.1.	פעולות המערכת	20
4.2.	טיפול בשגיאות	24

במסגרת קורס תכנות מתקדם ב-JAVA שנלמד במכללה האקדמית צפת, פותחה מערכת לניהול רכבים שמיועדת לשימוש על ידי חברות השכרת רכב. פרויקט זה נועד להפגין את היכולות שרכשנו במהלך הלימודים בקורס ולהחיל את התיאוריות והמתודולוגיות שנלמדו על תרחיש מעשי.

המערכת נבנתה עם מטרה לאפשר ניהול יעיל ומקיף של פרטי רכב, החל מקליטת רכבים חדשים, דרך עדכון מצבם ומחירם, ועד לטיפול בהשכרת רכבים והחזרתם. המערכת מספקת פתרונות לשני סוגי משתמשים: מנהלים, שמופקדים על ניהול המערכת והפעלתה, ולקוחות, שמחפשים להשכיר רכבים לטווח קצר או ארוך.

במהלך פיתוח המערכת, התמקדנו בפיתוח ממשק משתמש נוח ואינטואיטיבי, כאשר הדגש הושם על אפיון ובניית מודולים שמאפשרים ניווט פשוט ויעיל בין התפריטים השונים והפונקציונליות שהמערכת מציעה. כל חלקי המערכת והקוד שפותחו מותאמים לעקרונות של תכנות אובייקט מונחה עצמים, תוך שימוש במחלקות, ירושות ופולימורפיזם.

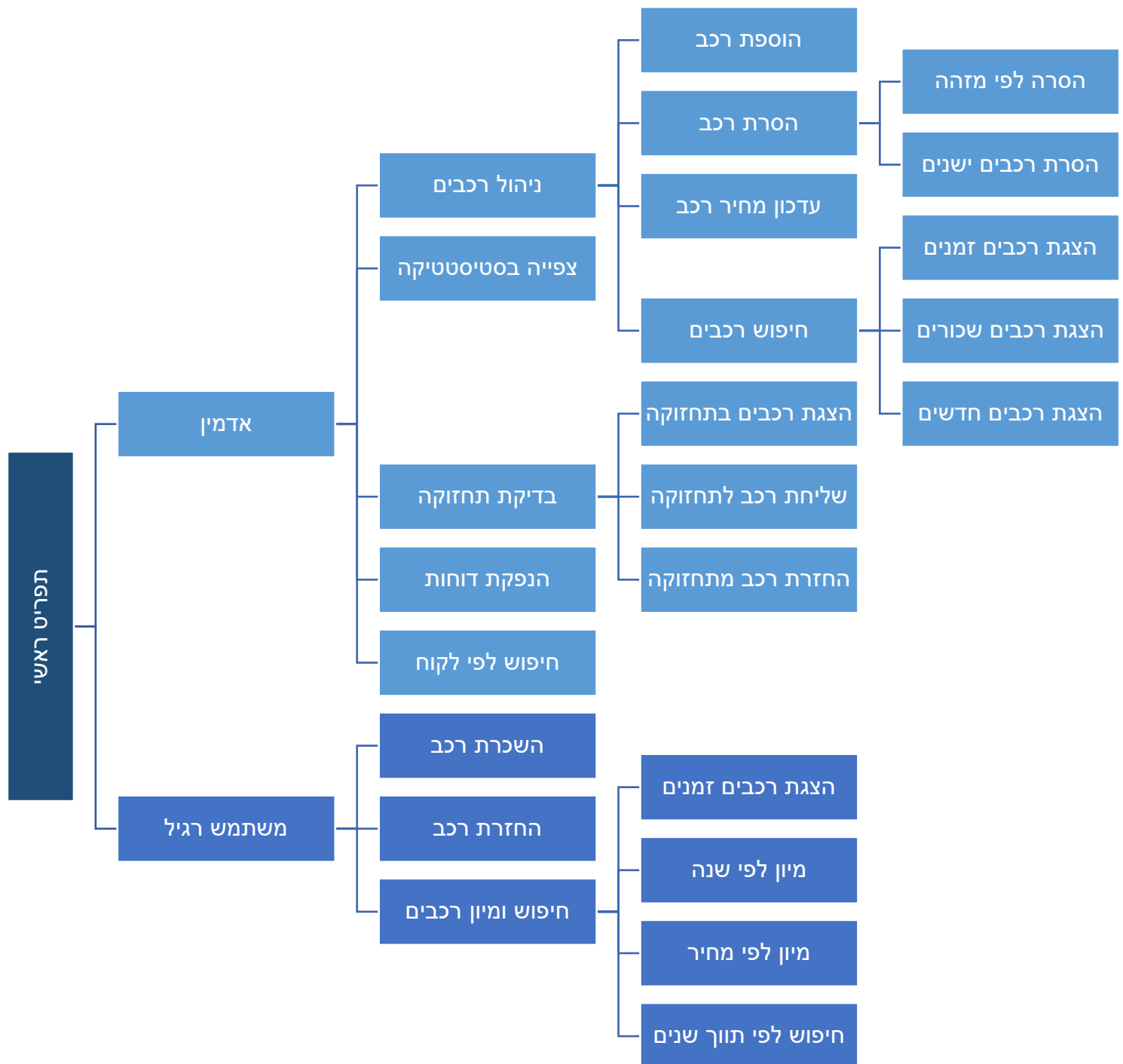
המערכת מעקבת אחר פעילות המשתמשים ומייצרת דוחות ונתונים סטטיסטיים המאפשרים לנו להבין את תפוקת המערכת ולשפר אותה בהתאם לצרכים המשתנים. כל זאת תוך כדי שמירה על רמת אבטחה גבוהה ומנגנונים להגנה על נתוני המשתמשים והרכבים.

הפרויקט כולל ניתוח של מצבי שגיאה ותקלות פוטנציאליות ומתן פתרונות לטיפול בהם, כדי להבטיח שהמערכת תעבוד בצורה חלקה וללא תקלות. אנו מקווים שתמצאו את העבודה שביצענו כמועילה ומתוך כך תוכלו לראות את ההתקדמות שעשינו במהלך הקורס.

## 2. פרק א' - תפריטים

### 2.1. ההירארכיה של התפריט

תרשים זרימה זה מתאר את התפריט של המערכת. התפריט מראה את הפעולות האפשריות של המערכת, המחולק לפי גישות משתמש.



## 2.2. תפריט מנהל (Admin Menu)

תפריט המנהל במערכת ניהול הרכבים מאפשר למנהל לבצע מגוון רחב של פעולות שמחולקות לקטגוריות מרכזיות: ניהול רכבים, תחזוקה, דוחות וסטטיסטיקה.

### ניהול רכבים

המנהל יכול להוסיף רכבים חדשים למערכת, תהליך שכולל את קליטת פרטי הרכב כמו מזהה, דגם, שנת ייצור ומחיר השכרה. כל רכב שנוסף מבוקר באופן מלא על ידי המנהל ונשמר בקובץ המידע.

בנוסף, ישנה האפשרות להסיר רכבים מהמערכת. פעולה זו נחוצה כאשר רכבים מוחלפים, נמכרים או יוצאים משימוש. המנהל צריך להזין את מזהה הרכב כדי לבצע הסרה.

עדכון פרטי רכב מתבצע גם הוא בקלות, כאשר המנהל יכול לשנות את מחיר ההשכרה של רכב קיים בהתאם לצרכים העסקיים.

### ניהול תחזוקה

תחזוקת רכבים היא חלק חיוני בהבטחת תקינותם ובטיחותם. המנהל מנהל את לוח הזמנים לתחזוקה ויכול לשלוח רכבים לתחזוקה כשצריך. רכב שנשלח לתחזוקה יוסר מרשימת הרכבים הזמינים להשכרה עד לשובו.

### דוחות וסטטיסטיקה

היכולת לייצר דוחות היא כלי חשוב בידי המנהל לניטור וניתוח הפעילות במערכת. דוחות אלה כוללים נתונים על מספר הרכבים הזמינים, רכבים בתחזוקה, הכנסות מהשכרות והיסטוריית השכרות.

המנהל מסוגל לעקוב אחרי פעילות המשתמשים ולהבין דפוסים שיכולים להוביל לשיפור בשירות ובהצעות המחיר.

### 2.3. תפריט משתמש (User Menu)

תפריט המשתמש במערכת ניהול הרכבים מעניק למשתמשים הרגילים את היכולת לבצע פעולות שונות בקלות וביעילות, תוך כדי שמירה על נוחות וגישה פשוטה למידע ולשירותים נדרשים.

#### השכרת רכבים

משתמשים יכולים להשכיר רכבים באופן עצמאי דרך המערכת. התהליך כולל בחירת רכב מרשימת הרכבים הזמינים, קביעת תאריכי ההשכרה והחזרה, ואישור סופי של ההזמנה. המערכת מדריכה את המשתמש דרך כל שלב, מהבחירה ועד לאישור הזמנה.

#### החזרת רכבים

לאחר סיום תקופת ההשכרה, המשתמשים מחזירים את הרכב לפי המזהה שלהם. המערכת מאמתת את הפרטים ומאפשרת רישום החזרה עצמאי, כולל עדכון תאריך ושעת החזרה. המערכת מחשבת אוטומטית כל דמי עיכוב אפשריים ומעדכנת את הלקוח בעלות הסופית.

#### חיפוש ומיון רכבים

משתמשים יכולים לחפש רכבים לפי פרמטרים שונים כמו דגם, שנה, מחיר או קטגוריה. המערכת מציעה אפשרויות מיון שונות שמאפשרות למשתמש לסנן ולמצוא בקלות את הרכב המתאים ביותר לצרכיו.

#### צפייה ברכבים זמינים

המערכת מספקת למשתמש תצוגה מיידיית של כל הרכבים הפנויים להשכרה בזמן אמת. זהו כלי חיוני שמאפשר ללקוחות לתכנן את השכרתם בצורה יעילה ומהירה.

### 3. פרק ב' - מבנה המערכת

#### 3.1. הקבצים המרכזיים

המחלקות השונות וכיצד הן תורמות לפעולת המערכת הכללית.

מס'	שם מחלקה	תיאור המחלקה
1	User	מחלקה מופשטת המהווה בסיס ליצירת סוגי משתמשים שונים במערכת.
2	AdminUser	פעולות הניהול המרכזיות במערכת כולל הוספה, הסרה ועדכון פרטי רכב, ניהול תחזוקה, והנפקת דוחות וסטטיסטיקות.
3	RegularUser	מאפשרת למשתמשים הרגילים לבצע פעולות השכרה והחזרת רכבים, וגם חיפוש ומיון רכבים לפני פרמטרים שונים.
4	Vehicle	מייצגת רכב בודד ומחזיקה פרטים כגון מזהה, דגם, שנת ייצור, מחיר השכרה מצב נוכחי (פנוי   מושכר   בתחזוקה).
5	VehicleOperations	ממשק המגדיר את כלל הפעולות הסטנדרטיות לניהול רכבים, כגון הוספה, הסרה, עדכון מחירים והצגת רכבים פנויים.
6	VehicleManager	מנהלת את כלל הפעולות הקשורות לרכבים במערכת כולל שמירה וטעינה של נתונים, ניהול השכרות, ותחזוקת רשימת הרכבים.
7	Rental	מייצגת השכרה פעילה של רכב, כולל פרטי הרכב, מזהה המשתמש השוכר, תאריכי התחלה וסיום, והעלות הכוללת של ההשכרה.
8	Logger	אחראית על רישום פעולות ושגיאות במערכת לקובץ לוג, לצורך ניטור ואחזקת המערכת.
9	InputValidator	מספקת כלים ואימות קלט מהמשתמש למניעת שגיאות ובעיות נתונים, כולל אימות פורמטים של מזהה רכב, מחירים, תאריכים ועוד.
10	MenuManager	מנהלת את התפריטים והניווט במערכת, אחראית על הצגת האפשרויות השונות והפעלת הפונקציות בהתאם לבחירות המשתמש.
11	Main	מחלקת ההרצה של התוכנית.

### *User.java 3.2*

מחלקה מופשטת ומשתמשת כבסיס למשתמשים במערכת, בתוכה נמצאות פונקציות המהוות את היסוד לפעולה של כל המשתמשים במערכת, ומאפשרות פולימורפיזם וגמישות בהתאמת הממשק לסוגי משתמשים שונים.

פונקציה	תיאור
User(String id, String name)	פונקציית בנאי שמאתחלת מופע של מחלקת User עם מזהה ייחודי ושם. מזהה המשתמש ושמו מאוחסנים כשדות פנימיים במחלקה ומשמשים לזיהוי והצגה במערכת.
getId()	מחזירה את מזהה המשתמש. משמשת לאימות וגישה לנתוני המשתמש במערכת.
getName()	מחזירה את שם המשתמש. הפונקציה משמשת להצגה ולהדפסה בממשק המשתמש.
displayMenu()	פונקציה מופשטת במחלקה וממוששת במחלקות היורשות שלה. מטרתה היא להציג את התפריט הרלוונטי לסוג המשתמש.



### AdminUser.java 3.3

מספקת יכולות ניהול ודיווח עבור מנהל המערכת, ומאפשרת ניהול יעיל ומדויק של הרכבים והשכרות במערכת.

פונקציה	תיאור
AdminUser(String id, String name, VehicleManager vehicleManager)	פונקציית בנאי של המחלקה. מקבלת מזהה משתמש, שם משתמש ומופע של VehicleManager. משמש לניהול הרכבים במערכת.
addVehicle(Vehicle vehicle)	מאפשרת למנהל להוסיף רכב חדש למערכת. הפונקציה מבצעת בדיקת תקינות ומוודאת שהרכב לא קיים כבר במערכת לפי ההוספה.
removeVehicle(String vehicleId)	מאפשרת למנהל להסיר רכב מהמערכת לפי מזהה הרכב. הפונקציה מבצעת בדיקה שהרכב קיים לפני ההסרה.
updateVehicle(String id, double price)	מאפשר למנהל לעדכן את מחיר ההשכרה של רכב קיים. היא מקבלת את מזהה הרכב והמחיר החדש לעדכון.
getVehicleManager()	מחזירה את מופע VehicleManager שמשויך למנהל. מאפשרת גישה לפעולות ניהול רכבים.
viewStatistics()	מציגה סטטיסטיקות כלליות על מצב הרכבים במערכת, כולל מספר הרכבים הפנויים והמושכרים.
showLateFees()	מציגה דוח על דמי איחור השכרות שלא הוחזרו בזמן, מחשבת את דמי האיחור עבור כל השכרה על פי מספרים הימים שחלפו מעבר למועד ההחזרה.
generateRentalReport()	מייצרת דוח מופרט של כל השכרות המערכת, כולל השכרות פעילות והיסטוריית השכרות מלאה.
generateReport()	מייצרת דוח כולל של הפעילות במערכת, כולל דוח השכרות ודוח הכנסות מכל ההכנסות.
displayMenu()	מציגה את התפריט המנהלי, הכולל אפשרויות לניהול הרכבים, צפייה בסטטיסטיקות, ניהול תחזוקה, יצירת דוחות וחיפוש אחר משתמש.

### *RegularUser.java 3.4*

מספקת את הכלים הנדרשים למשתמש הרגיל לביצוע הפעולות השוטפות במערכת השכרות. כל פונקציה מותאמת לצרכים הספציפיים של המשתמש ומספקת ממשק פשוט ונוח לשימוש.

פונקציה	תיאור
RegularUser(String id, String name, VehicleManager vehicleManager)	פונקציית בנאי של המחלקה. מקבלת מזהה משתמש, שם משתמש ומופע של VehicleManager. vehicleManager משמש לניהול פעולות הקשורות לרכבים עבור המשתמש.
rentVehicle(Scanner scanner)	מאפשרת למשתמש להשכיר רכב. המשתמש נדרש להזין את מזהה המשתמש, מזהה הרכב ותאריך התחלת ההשכרה. הפונקציה בודקת זמינות ומבצעת את ההשכרה אם הרכב פנוי.
returnVehicle(Scanner scanner)	מאפשרת למשתמש להחזיר רכב שהושכר. המשתמש נדרש להזין את מזהה המשתמש, מזהה הרכב ותאריך ההחזרה. הפונקציה מחשבת את כלל העלות המצטברת להשכרה ומעדכנת את מצב הרכב לפנוי.
searchAndSortVehicles()	מאפשרת למשתמש לחפש ולמיין רכבים לפי מספר פרמטרים כגון שנה, דגם או מחיר. התוצאות ממוינות ומוצגות למשתמש.
getVehicleManager()	מחזירה את המופע VehicleManager המשווין למשתמש. מאפשר גישה ישירה לפעולות הקשורות לניהול רכבים.
displayAvailableVehicles()	מציגה את כל הרכבים הפנויים להשכרה ברגע נתון. משמשת לסקירה מהירה של רכבים שהמשתמש יכול להשכיר.
displayMenu()	מציגה את תפריט המשתמש למשתמשים רגילים, כולל אפשרויות להשכרה, החזרה, חיפוש ומיון רכבים.

מספקת את כל הפונקציונליות הנדרשת לניהול רכב בודד במערכת, כולל ניהול מחיר, מצב, וסוגי מיון רכבים. זו מחלקה מרכזית שמאפשרת פעולה יעילה ומאורגנת של פרטי הרכבים במערכת.

פונקציה	תיאור
Vehicle(String id, String model, int year, double rentalPrice, String status)	פונקציית בנאי, מקבלת מזהה ייחודי, דגם, שנת יצור, מחיר השכרה, ומצב הרכב (פנוי   מושכר   בתחזוקה).
getId()	מחזירה את מזהה הרכב. פונקציה זו משמשת לזיהוי ייחודי של הרכב במערכת.
getModel()	מחזירה את דגם הרכב. ניתן להשתמש במידע זה לחיפושים ולהשוואות בין רכבים.
getYear()	מחזירה את שנת היצור של הרכב. מידע זה יכול לעזור בהערכת ערך הרכב ובחיפושים לפי קריטריונים של גיל הרכב.
getRentalPrice()	מחזירה את מחיר ההשכרה של הרכב. חשוב להצגת מחירים ללקוחות ולחישוב הכנסות מהשכרות.
getStatus()	מחזירה את מצב הרכב (פנוי   מושכר   בתחזוקה). פונקציה זו חשובה לניהול הזמינות ותפעול הרכב.
setRentalPrice(double rentalPrice)	מעדכנת את מחיר ההשכרה של הרכב. מאפשרת לנהל את מחירי ההשכרה.
setStatus(String status)	מעדכנת את מצב הרכב. פונקציה זו מאפשרת לשנות את מצב הרכב במערכת לפי צרכי התפעול והתחזוקה.
setSortType(SortType type)	מגדירה את סוג המיון לרשימת הרכבים. מאפשרת למשתמש לבחור באופן דינאמי את הקריטריון למיון הרכבים (למשל, לפי מחיר, שנה, דגם).
compareTo(Vehicle other)	משווה בין הרכב הנוכחי לרכב אחר לפי הקריטריון שהוגדר ב-setSortType מאפשרת השוואה ומיון רכבים ברשימה.
toString()	מחזירה מחרוזת המייצגת את הרכב עם כל הפרטים הרלוונטיים. שימושית להדפסות לוג ולהצגה גרפית של נתוני הרכב.

### *VehicleOperations.java 3.6*

הממשק משמש כבסיס להגדרת פעולות הניהול של רכבים במערכת. כל אחת מהפונקציות אלו נועדה לספק ניהול יעיל ונוח של הרכבים, מהשכרה ועד תחזוקה.

פונקציה	תיאור
<code>addVehicle(Vehicle vehicle)</code>	מאפשרת להוסיף רכב חדש למערכת. דורשת מופע של רכב ומוסיפה אותו לרשימת הרכבים הזמינים, בתנאי שהמזהה של הרכב אינו כבר קיים במערכת.
<code>removeVehicle(String vehicleId)</code>	מאפשרת להסיר רכב מהמערכת. דורשת מזהה של רכב ומבצעת את הסרתו מרשימת הרכבים, אם הרכב נמצא במערכת.
<code>updateVehicle(String vehicleId)</code>	מאפשרת לעדכן פרטים של רכב קיים, כמו מחיר ההשכרה. דורשת מזהה של רכב והפרטים החדשים לעדכון. פעולה זו מבטיחה שהמידע על הרכב יישאר מעודכן ומדויק.
<code>displayAvailableVehicles()</code>	מציגה את כל הרכבים הפנויים להשכרה. פונקציה זו משמשת להצגת רשימת כל הרכבים שלא נמצאים כרגע בהשכרה, ומאפשרת למשתמשים לראות במהירות אילו רכבים זמינים.
<code>searchAndSortVehicles()</code>	מאפשרת חיפוש ומיון של רכבים לפי קריטריונים שונים כגון שנה, דגם, ומחיר. המשתמשים יכולים לבחור את הקריטריונים הרצויים והמערכת תציג רשימה ממוינת של רכבים שמתאימים לדרישות.

### VehicleManager.java 3.7

תפקידה לנהל את כל הפעולות הקשורות לרכבים במערכת, היא מספקת את כל הכלים הדרושים לניהול מלא ויעיל של הרכבים במערכת, מהשכרה דרך תחזוקה ועד ניהול מלאי והצגת נתונים סטטיסטיים.

פונקציה	תיאור
VehicleManager()	פונקציית בנאי, היא מאתחלת את רשימות הרכבים, הרכבים המושכרים ומכינה את המערכת לפעילות.
preloadVehicles()	טוענת את הרכבים לרשימת הרכבים במערכת. מבטיחה שהרכבים יהיו זמינים לשימוש מיד עם הפעלת המערכת.
addVehicle(Vehicle vehicle)	מוסיפה רכב חדש למערכת. מבצעת בדיקה שלא קיים רכב עם אותו מזהה לפני ההוספה.
removeVehicle(String id)	מסירה רכב מהמערכת לפי מזהה רכב נתון. בודקת קודם שהרכב קיים במערכת.
updateVehicle(String id, double price)	מעדכנת את מחיר ההשכרה של רכב קיים לפי המחיר החדש שהוזן.
displayAvailableVehicles()	מציגה את כל הרכבים הפנויים להשכרה. יעיל לצפייה מהירה של הרכבים הזמינים ללקוחות.
displayRentedVehicles()	מציגה את כל הרכבים שכרגע מושכרים. מאפשרת ניהול ומעקב על רכבים שנמצאים בשימוש.
searchAndSortVehicles()	מאפשרת חיפוש ומיון של רכבים לפי מאפיינים שונים כגון דגם, שנה, ומחיר.
findByYear(int start, int end)	מחפשת רכבים ששנת ייצורם נמצאת בתוך טווח שנים נתון. מאפשרת חיפוש ממוקד של רכבים לפי גיל.
findVehicleById(String vehicleId)	מחפשת ומחזירה רכב לפי מזהה ייחודי. משמשת לגישה מהירה וקלה לרכב ספציפי.
showTop3Newest()	מציגה את שלושת הרכבים החדשים ביותר במערכת. שימושי לקידום מכירות או השכרות של דגמים חדשים.

## המשך - VehicleManager.java

פונקציה	תיאור
removeOld(int currentYear)	מסירה רכבים ישנים מהמערכת ששנת ייצורם מתחת לשנה הנתונה. פונקציה זו מאפשרת ניקוי ועדכון של הרשימה לכלול רק רכבים עדכניים יותר.
calculateDays(String startDate, String endDate)	מחשבת את מספר הימים בין שתי תאריכים. משמשת לחישוב תקופות השכרה ודמי איחור.
displayStatistics()	מציגה סטטיסטיקות כלליות על פעילות הרכבים במערכת, כולל מספר השכרות, הרכבים הפנויים והמושכרים.
rentVehicle(String vehicleId, String userId, String startDate)	מתבצעת השכרה של רכב למשתמש. דורשת מזהה רכב, מזהה משתמש ותאריך התחלה. מעדכנת את מצב הרכב למושכר.
returnVehicle(String vehicleId, String userId, String returnDate)	מתבצעת החזרה של רכב שהושכר. דורשת מזהה רכב, מזהה משתמש ותאריך החזרה. מעדכנת את מצב הרכב ומחשבת עלות כוללת של ההשכרה.
calculateTotalCost(Rental rental, int rentalDays)	מחשבת את העלות הכוללת של השכרה, כולל דמי איחור אם יש. מבוססת על מחיר ההשכרה ליום ומספר הימים.
displayEarningsReport()	מציגה דוח הכנסות מהשכרות הרכבים. כולל נתונים על כל השכרות וההכנסות שנצברו מהן.
showUserTotalCost(String userId)	מציגה את סך ההוצאות של משתמש ספציפי על השכרות רכב. מאפשרת ניטור וניהול חשבונות לקוחות.
checkMaintenance()	בודקת אילו רכבים נמצאים בתחזוקה ומציגה אותם. משמשת לניהול תחזוקת רכבים במערכת.
sendToMaintenance(String id)	שולחת רכב לתחזוקה. מקבלת מזהה רכב ומעדכנת את מצבו ל"בתחזוקה".
restoreVehicle(String id)	משחזרת רכב מתחזוקה ומחזירה אותו לסטטוס פנוי או זמין להשכרה. מאפשרת שימוש חוזר ברכב לאחר תחזוקה.

## המשך - VehicleManager.java

פונקציה	תיאור
getVehicles()	מחזירה את רשימת כל הרכבים במערכת. משמשת לגישה נוחה ומהירה לכל הרכבים ומאפשרת פעולות ניהול ועדכון על כלל הרכבים.
getRentalHistory()	מחזירה את היסטוריית השכרות לכל הרכבים. מאפשרת צפייה בכל השכרות שבוצעו, כולל תאריכים ופרטי משתמשים שהשכירו את הרכבים.
getTotalRevenue()	מחזירה את ההכנסה הכוללת מהשכרות הרכבים. פונקציה זו מספקת נתונים חיוניים לניתוח כלכלי של העסק ולתכנון עתידי.
saveToFile()	שומרת את כל הנתונים הרלוונטיים של המערכת לקובץ, כולל רשימת הרכבים והיסטוריית השכרות. מבטיחה שהנתונים לא יאבדו בין הפעלות של המערכת.
loadFromFile()	טוענת את כל הנתונים הרלוונטיים מקובץ, כולל רשימת הרכבים והיסטוריית השכרות. מאפשרת שחזור של מצב המערכת לאחר הפעלה מחודשת.

תורמת לניהול יעיל ומסודר של השכרות במערכת, מספקת ממשק נוח להצגה ועדכון של נתוני השכרות, ומאפשרת חישובים שוטפים של עלויות ופרטי ההשכרה.

פונקציה	תיאור
<code>Rental(String id, String model, int year, double rentalPrice, String userId, String startDate)</code>	פונקציית בנאי להשכרת רכב. מקבלת מזהה רכב, דגם, שנת ייצור, מחיר השכרה, מזהה המשתמש השוכר ותאריך התחלת ההשכרה.
<code>getUserId()</code>	מחזירה את מזהה המשתמש השוכר. מאפשרת זיהוי וגישה לנתוני המשתמש השוכר.
<code>getStartDate()</code>	מחזירה את תאריך התחלת ההשכרה. שימושית לחישוב תקופת ההשכרה ולניתוחים נוספים.
<code>getEndDate()</code>	מחזירה את תאריך סיום ההשכרה. זהו תאריך שנקבע לאחר החזרת הרכב ומשמש לחישוב העלות הכוללת של ההשכרה.
<code>setEndDate(String endDate)</code>	מגדירה את תאריך סיום ההשכרה. מאפשרת עדכון של התאריך במידת הצורך, למשל כאשר הרכב מוחזר מאוחר יותר מהצפוי.
<code>setTotalCost(double totalCost)</code>	מגדירה את העלות הכוללת של ההשכרה. שימושית לעדכון סופי של העלות לאחר חישוב של דמי איחור או שינויים אחרים.
<code>getTotalCost()</code>	מחזירה את העלות הכוללת של ההשכרה. חשובה לניהול חשבונות ולדיווח למשתמשים.
<code>toString()</code>	מחזירה מחרוזת המייצגת את השכרה, כולל כל הפרטים הרלוונטיים. מאוד שימושי להדפסות ולהצגת נתונים למשתמש.



מאפשרת למפתחים ולמנהלי המערכת לקבל תובנות על פעולות המשתמשים, לזהות ולטפל בבעיות תפעוליות ובאירועי שגיאה, ולשפר את המערכת לאורך זמן.

פונקציה	תיאור
<code>log(String message)</code>	מוסיפה רשומת לוג כללית לקובץ הלוג או למערכת הלוגים. משמשת לתיעוד פעולות רגילות ומידע נוסף במערכת.
<code>logError(String errorMessage)</code>	מוסיפה רשומת לוג עבור שגיאות במערכת. מסייעת באיתור בעיות ובטיפול בהן, ובניהול נתוני איכות ואמינות של המערכת.
<code>logAction(String actionMessage)</code>	מוסיפה רשומת לוג עבור פעולות משתמש או פעולות אוטומטיות במערכת. משמשת למעקב אחר פעילות המשתמשים ולניתוח תהליכים במערכת.

כלי מרכזי לאימות נתוני קלט ולהבטחת אינטגרציה ותקינות של הנתונים המוזנים למערכת. השימוש בחריגות מאפשר ניהול מרוכז ובקרה על שגיאות ובעיות נתונים בזמן ריצת המערכת.

פונקציה	תיאור
<code>readPositiveInt(Scanner scanner, String errorMessage)</code>	קוראת מספר שלם חיובי מהמשתמש. מציגה הודעת שגיאה נתונה אם הקלט אינו חוקי ומבקשת קלט מחדש.
<code>readPositiveDouble(Scanner scanner, String errorMessage)</code>	קוראת מספר עשרוני חיובי מהמשתמש. עובדת באופן דומה ל <code>readPositiveInt</code> עם התאמה לקליטיים עשרוניים.
<code>validateUniqueVehicleId(String vehicleId, ArrayList&lt;Vehicle&gt; vehicles)</code>	בודקת אם מזהה רכב נתון כבר קיים במערכת. מחזירה <code>True</code> אם המזהה ייחודי, אחרת היא מחזירה <code>False</code> .
<code>validateVehicleId(String vehicleId)</code>	בודקת אם מזהה רכב נתון תקין לפי הפורמט הנדרש במערכת.
<code>validateModelName(String modelName)</code>	בודקת את תקינות שם הדגם של הרכב. עשויה לכלול בדיקות למניעת תווים לא חוקיים או אורך לא תקין.
<code>validatePrice(String price)</code>	בודקת אם מחיר נתון תקין ומהווה מספר חיובי.
<code>validateDate(String date)</code>	בודקת אם תאריך נתון מתאים לפורמט תאריך תקני במערכת.
<code>validateStatus(String status)</code>	בודקת אם מצב רכב נתון (פנוי   מושכר   בתחזוקה) תקין וקיים ברשימת המצבים המותרים.
קלאס	תיאור
<code>InvalidInputException(String message)</code>	חריגה שמופעלת כאשר קלט מהמשתמש אינו תקין. מקבלת הודעת שגיאה להצגה.
<code>VehicleNotFoundException(String message)</code>	חריגה שמופעלת כאשר רכב במזהה נתון אינו נמצא במערכת.
<code>InvalidRentalDateException(String message)</code>	חריגה שמופעלת כאשר תאריכי השכרה או החזרה אינם תקינים או לא הגיוניים (לדוגמה, תאריך התחלה אחרי תאריך סיום).

### MenuManager.java 3.11

מהווה את החלק המרכזי בממשק המשתמש, ומאפשרת ניווט נוח ומסודר בין האפשרויות השונות שהמערכת מציעה. היא מספקת תפריטים מגוונים עבור סוגי משתמשים שונים ומאפשרת ניהול יעיל של כל פעילויות המערכת.

פונקציה	תיאור
MenuManager()	פונקציית בנאי, אחראית לאתחול המחלקה והכנתה לתפקוד.
run()	מתחילה את תהליך התפריט והניווט במערכת. מציגה את התפריט הראשי ומנתבת לתת-תפריטים בהתאם לבחירת המשתמש.
adminMenu()	מציגה את תפריט המנהל, אשר כולל אפשרויות כגון ניהול רכבים, תחזוקה, דוחות ועוד.
userMenu()	מציגה את תפריט המשתמש הרגיל, עם אפשרויות כגון השכרת רכב, החזרת רכב וחיפוש רכבים.
manageVehiclesMenu()	תפריט ניהול רכבים שמאפשר למנהל להוסיף, לעדכן או להסיר רכבים מהמערכת.
removeVehicleMenu()	תפריט משני שמאפשר הסרת רכב מהמערכת. דורש מהמנהל להזין את מזהה הרכב להסרה.
searchVehiclesMenu()	תפריט שמאפשר חיפוש רכבים לפי מספר פרמטרים ומיון התוצאות.
maintenanceMenu()	תפריט המאפשר למנהל לנהל את תחזוקת הרכבים, כולל שליחה לתחזוקה ושחזור מתחזוקה.
searchAndSortMenu()	תפריט שמאפשר מיון וחיפוש מתקדמים של רכבים במערכת, לפי קריטריונים שונים.
inputVehicle()	פונקציה להזנת נתוני רכב חדש או עדכון רכב קיים. משמשת בתוך תפריטים של הוספה ועדכון רכבים.

מחלקה זו מתחילה את תהליך הריצה של היישום ומהווה את הבסיס לכל הזרימה של היישום. כאן מתרחשים אתחול המערכת והתחלת האינטראקציה עם המשתמש.

פונקציה	תיאור
main(String[] args)	פונקציה זו משמשת כנקודת הכניסה ליישום. היא אחראית לאתחול ולהפעלה של המערכת. בתוך הפונקציה נוצר מופע של MenuManager אשר מתחיל את זרימת התפריטים ומאפשרת למשתמשים לבצע אינטראקציה עם המערכת.

#### 4. פרק ג' - תהליכי המערכת

##### 4.1. פעולות המערכת

הוספת רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	הזנת פרטי רכב חדש דרך ממשק המשתמש.	AdminUser, MenuManager
עיבוד	בדיקת ייחודיות הרכב במערכת והוספה אם לא קיים.	VehicleManager
פלט	החזרת תוצאת הבדיקה והודעה למשתמש על תוצאת ההוספה.	AdminUser, MenuManager, Logger

הסרת רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	הזנת מזהה הרכב למחיקה דרך ממשק המשתמש.	AdminUser, MenuManager
עיבוד	בדיקת קיום הרכב במערכת ומחיקתו אם נמצא.	VehicleManager
פלט	החזרת תוצאת הבדיקה והודעה למשתמש על תוצאת המחיקה.	AdminUser, MenuManager, Logger

עדכון רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	הזנת מזהה רכב ופרטים לעדכון דרך ממשק המשתמש.	AdminUser, MenuManager
עיבוד	בדיקת קיום הרכב במערכת ועדכון הפרטים בהתאם לקלט.	VehicleManager
פלט	החזרת תוצאת העדכון והודעה למשתמש על תוצאת העדכון.	AdminUser, MenuManager, Logger

חיפוש רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	הזנת פרמטרים לחיפוש רכב (לדוגמה, דגם, שנה, מצב).	AdminUser, MenuManager
עיבוד	ביצוע חיפוש במאגר הרכבים על פי הפרמטרים הנתונים.	VehicleManager
פלט	החזרת רשימת רכבים שמתאימים לקריטריונים והצגתם לאדמין.	AdminUser, MenuManager

הצגת סטטיסטיקה		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	בקשה לצפייה בסטטיסטיקות רכב מהממשק.	AdminUser, MenuManager
עיבוד	איסוף ועיבוד נתונים סטטיסטיים מהמאגר.	VehicleManager
פלט	הצגת הסטטיסטיקות לאדמין, כולל נתוני שימוש, השכרות ותקלות.	AdminUser, MenuManager

הנפקת דוחות		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	בקשה לצפייה בדוחות רכב מהממשק.	AdminUser, MenuManager
עיבוד	איסוף ועיבוד נתוני דוחות מהמאגר.	VehicleManager
פלט	הצגת הדוחות לאדמין, כולל נתוני שימוש, השכרות ותקלות.	AdminUser, MenuManager

הצגת רכבים בתחזוקה		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	בקשה לרשימת רכבים בתחזוקה מהממשק.	AdminUser, MenuManager
עיבוד	איסוף נתונים על רכבים הנמצאים בתחזוקה.	VehicleManager
פלט	הצגת רשימת הרכבים לאדמין.	MenuManager

הוספת רכב לתחזוקה		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	בחירת רכב מהרשימה ושליחתו לתחזוקה.	AdminUser, MenuManager
עיבוד	עדכון מערכת לסטטוס רכב "בתחזוקה".	VehicleManager
פלט	אישור עדכון והצגת הודעה לאדמין על שליחת הרכב לתחזוקה.	MenuManager

החזרת רכב מתחזוקה		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	בחירת רכב מרשימת רכבים בתחזוקה להחזרה.	AdminUser, MenuManager
עיבוד	עדכון מערכת להסרת סטטוס "בתחזוקה" מהרכב.	VehicleManager
פלט	אישור עדכון והצגת הודעה לאדמין על החזרת הרכב מתחזוקה.	MenuManager

השכרת רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	המשתמש מקליד את מזהה הרכב שהוא רוצה להשכיר.	RegularUser, VehicleManager
עיבוד	המערכת מעדכנת את מצב הרכב ל"מושכר" ותאריך החזרה משוער.	VehicleManager
פלט	עדכון מצב הרכב במערכת ל"מושכר"	RegularUser, MenuManager

החזרת רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	המשתמש מקליד את מזהה הרכב שהוא רוצה להחזיר לאחר השכרה.	RegularUser, VehicleManager
עיבוד	המשתמש מחזיר רכב שהושכר. המערכת מעדכנת את מצב הרכב במערכת ל"זמין".	VehicleManager
פלט	עדכון מצב הרכב במערכת ל"זמין".	RegularUser, MenuManager

חיפוש רכב		
שלב	פעולה ותוצאה	מחלקות מעורבות
קלט	המשתמש בוחר את אפשרות החיפוש	RegularUser, VehicleManager
עיבוד	משתמש מבצע חיפוש רכבים במערכת לפי קריטריונים שונים כמו דגם, שנת ייצור, וטווח שנים.	VehicleManager
פלט	רשימת רכבים שמתאימים לקריטריונים.	RegularUser, MenuManager



## 4.2. טיפול בשגיאות

מערכת זו מצוידת במנגנונים מתקדמים לזיהוי, תיקון ומניעת שגיאות, שמטרתם להבטיח את רציפות העבודה ולמזער הפרעות למשתמשים.

### זיהוי שגיאות

המערכת עושה שימוש במספר שיטות לזיהוי שגיאות :

**בדיקות תקינות קלט** - מחלקת InputValidator בודקת את תקינות הנתונים הנכנסים מהמשתמשים למנוע שגיאות נתונים.

**בדיקות תקינות בזמן ריצה** - המערכת מפעילה בדיקות על תהליכים פנימיים ותקשורת עם מאגרי הנתונים לזיהוי שגיאות פנימיות.

### טיפול ומעקב אחר שגיאות

כאשר שגיאה זוהתה, המערכת פועלת לטפל בה באמצעות מספר גישות :

**תיקון שגיאות** - במקרה של שגיאות נתונים InputValidator, תדרוש מהמשתמש להזין שוב נתונים תקינים.

**דיווח על שגיאות** - כל שגיאות נרשמות על ידי מחלקת Logger, ששומרת יומני מעקב מפורטים על פעולות המערכת והשגיאות שהתרחשו.

### שגיאות בביטויים רגולריים (Regex)

המערכת כוללת מדיניות ברורה לבקרת שגיאות :

**זיהוי שגיאות** - המערכת משתמשת בכלים פנימיים לבדיקת תקינות ה-Regex לפני שהוא מופעל, מה שמונע רוב השגיאות הפוטנציאליות.

**התמודדות עם שגיאות** - במקרה של זיהוי שגיאה ב-Regex, המערכת מפעילה מנגנוני תיקון אוטומטיים או מציגה הודעת שגיאה מפורטת למשתמש, המסבירה את הבעיה ומציעה דרכי פתרון.