תוכנה 1 – סתיו תשע"ה

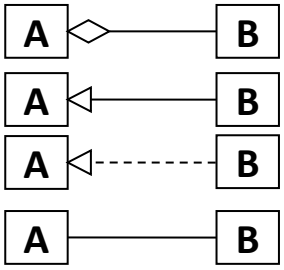
תרגיל מספר 9

הורשה

<u>הנחיות כלליות:</u>
קראו בעיון את קובץ נהלי הגשת התרגילים אשר נמצא באתר הקורס.

- הגשת התרגיל תעשה במערכת ה-moodle בלבד (http://moodle.tau.ac.il/).
- יש להגיש קובץ zip יחיד הנושא את שם המשתמש ומספר התרגיל (לדוגמא, עבור המשתמש aviv יקרא הקובץ aviv_hw9.zip). קובץ ה-zip יכיל:
  א. קובץ פרטים אישיים בשם details.txt המכיל את שמכם ומספר ת.ז.
  ב. קבצי ה- java של התוכניות אותם התבקשתם לממש, כולל תיקיות החבילה.
  ג. קובץ PDF בשם answers.pdf המכיל את התשובות לשאלות.

<u>הערה כללית:</u> בתרגיל זה אתם מתבקשים, בין היתר, לשרטט דיאגרמות של מחלקות. השתמשו בסימונים הבאים בלבד:



- (aggregation) יחס של **הכלה** (למשל, ל-A יש שדה מטיפוס B)
- **ירושה** (B מחלקה היורשת את A):
- **מימוש** (B מחלקה המממשת\ מנשק היורש את המנשק A):
- (association) קשר כללי שאינו נופל בקטגוריות הקודמות. למשל, A משתמש במשתנה מטיפוס B באחת המתודות.

יש לציין: בתוך כל מלבן <<interface>>, <<abstract>> או <<class>>, ואת שם המנשק או המחלקה.

אין צורך לציין: מספרים ושמות שדות על יחסי אגרגציה ואסוציאציה; שמות מתודות ושדות בתוך מלבני המחלקות; יחסים "עקיפים" בין מחלקות (כלומר, אם C יורש מ-B שיורש מ-A, אין צורך לציין קשר בין C ל-A אלא אם יש ביניהם קשר ישיר בנוסף, למשל של הכלה)

יצירת הדיאגרמות: ניתן לעשות זאת דרך PowerPoint, Word, תוכנת הציור המועדפת עליכם או לסרוק שרטוט (בכתב ברור!).

Starfleet Command

בתרגיל זה נבנה מערכת תוכנה לניהול צי חלליות עתידני.

בתחילה נבנה מחלקות שייצגו את אנשי הצוות ואת סוגי החלליות השונים תוך שימוש במנשקים, מחלקות אבסטרקטיות והורשה. לאחר מכן, ניצור אובייקטים של מחלקות אלו ולבסוף נדפיס מספר דוחות המציגים חיתוכי מידע שונים על צי החלליות שלנו כגון עלות אחזקה כוללת, כוח-אש כולל של חלליות הצי ועוד.

Starfleet Personnel

צי החלל של פדרציית הכוכבים המאוחדת כולל 2 סוגי אנשי צוות (Crew-Members):

1. Crewman – איש צוות רגיל.
2. Officer – איש צוות שהינו קצין (בעל דרגת קצונה).

הנה תיאור השירותים בהם תתמוך כל אחת מהמחלקות המייצגות את סוגי אנשי הצוות הנ"ל:

Crewman

שם השירות	טיפוס החזרה	הסבר
getName()	String	שם איש הצוות (מחרוזת המציינת שם ייחודי לכל איש צוות).
getAge()	int	גילו של איש הצוות (בשנות כדור-הארץ, למשל 28).
getYearsInService()	int	מספר שנות השירות של איש הצוות (בשנות כדור-הארץ, למשל 10).

Officer

איש צוות שהינו קצין יכלול את כל התכונות של איש צוות רגיל, ובנוסף תהיה לו גם דרגת קצונה:

שם השירות	טיפוס החזרה	הסבר
getRank()	OfficerRank	הדרגה של הקצין (מיוצג ע"י Enum בשם OfficerRank).

שימו לב:

- בהמשך נגדיר מנשק בשם **CrewMember** אשר ייצג איש צוות מסוג כלשהו.
- הטיפוס **OfficerRank** הוא Enum המגדיר קבועים המציינים את דרגות הקצונה. טיפוס זה נתון לכם.
- ההתייחסות בלשון זכר לאנשי הצוות היא מטעמי קיצור בלבד.

Starfleet Ships

צי החלל של פדרציית הכוכבים המאוחדת כולל 5 סוגי חלליות:

1. Medical Ship – חללית רפואה המעניקה שירותי תמיכה רפואיים לסגל הפדרציה.
2. Transport Ship – חללית תובלה המאפשרת שינוע נוסעים ומטען בין בסיסי חלל.
3. Fighter – חללית קרב (Battleship) קטנה ומהירה.
4. Bomber – חללית קרב (Battleship) גדולה בעלת עוצמת אש אדירה.
5. Stealth Cruiser –חללית קרב (Battleship) מהירה בעלת יכולת חמקנות (Stealth).

להלן פירוט המאפיינים של סוגי החלליות השונות:

Spaceship

נתחיל בתיאור השירותים המשותפים לכל סוגי החלליות. עבור כל חללית (להלן Spaceship) נגדיר את השירותים הבאים:

שם השירות	טיפוס החזרה	הסבר
getName()	String	שם החללית (מחרוזת המציינת שם ייחודי לכל חללית).
getCommissionYear()	int	שנת ייצור (בשנות כדור הארץ, למשל 2241).
getMaximalSpeed()	float	מהירות מקסימלית (שבר בין 0 ל-10).
getFirePower()	int	סכום כוח-אש של כל כלי הנשק המותקנים בחללית (מספרים שלמים, ביחידות של כוח-אש). לכל חללית יש כוח-אש בסיסי מובנה של 10 יחידות כוח-אש. בחלליות קרב מתווסף כוח-אש נוסף מכלי הנשק המותקנים על החללית.
getCrewMembers()	Set<CrewMember>	חברי הצוות המאיישים את החללית (CrewMember הינו מנשק המייצג איש-צוות מסוג כלשהו).
getAnnualMaintenanceCost()	int	עלות אחזקה שנתית כוללת (מספרים שלמים) ביחידות של דולר-פדרציה. נתון זה יחושב באופן שונה לכל סוג חללית על פי המפורט בהמשך.

Medical Ship

עבור חללית רפואה נגדיר את כל השירותים של חללית המובאים לעיל, בתוספת ההגדרות הבאות:

שם השירות	טיפוס החזרה	הסבר
getNumberOfMedicalClinics()	int	מספר המרפאות המותקנות על חללית הרפואה. עלות האחזקה של כל מרפאה היא 3000 דולר לשנה.
getAnnualMaintenanceCost()	int	עלות האחזקה השנתית כוללת של ספינת רפואה מורכבת מסכום הרכיבים הבאים: • עלות אחזקה שנתית בסיסית לספינה רפואה (8000 דולר) • עלות אחזקה שנתית של המרפאות (מספר המרפאות * 3000 דולר).

Transport Ship

עבור חללית תובלה נגדיר את כל השירותים של חללית המובאים לעיל, בתוספת ההגדרות הבאות:

שם השירות	טיפוס החזרה	הסבר
getCargoCapacity()	int	יכולת נשיאת מטען, ביחידות של מגה-טון (מספר שלם)
getPassengerCapacity()	int	יכולת נשיאת נוסעים, ביחידות של מספר נוסעים (מספר שלם)
getAnnualMaintenanceCost()	int	עלות האחזקה השנתית הכוללת של ספינת תובלה מורכבת מסכום הרכיבים הבאים:

	• עלות אחזקה שנתית בסיסית לספינת תובלה (3000 דולר). • עלות של 5 דולר לכל מגה-טון של יכולת נשיאת מטען (כלומר CargoCapacity * 5 דולר). • עלות של 3 דולר פר יכולת נשיאת נוסע (כלומר PassengerCapacity * 3 דולר).

Fighter

חללית קרב מהירה. נגדיר עבורה את כל השירותים של חללית בנוסף להגדרות הבאות:

שם השירות	טיפוס החזרה	הסבר
getWeaponArray()	List<Weapon>	רשימת כלי הנשק המותקנים על חללית הקרב. עבור כל נשק (Weapon) נשמור את הנתונים הבאים: • שם כלי הנשק. • כוח-אש (ביחידות כוח-אש). • עלות אחזקה שנתית (בדולרים).
getFirePower()	int	כוח האש המצטבר של חללית קרב הינו סכום כוח-האש של כל הנשקים המותקנים, בנוסף לכוח האש המובנה של כל חללית.
getAnnualMaintenanceCost()	int	עלות האחזקה השנתית של חללית קרב מסוג Fighter מורכבת מסכום הרכיבים הבאים: • עלות אחזקה שנתית בסיסית לספינת קרב Fighter (2500 דולר). • עלות אחזקה השנתית של כלי הנשק (סכום עלות האחזקה של כל כלי הנשק המותקנים על חללית הקרב). • עלות אחזקת מנועי החללית כתלות במהירות החללית המקסימלית (MaximalSpeed * 1000, מעוגל לשלמים).

Bomber

חללית קרב כבדה בעל יכולת הפצצה מרשימה. נגדיר עבורה את כל השירותים של חללית בנוסף להגדרות הבאות:

שם השירות	טיפוס החזרה	הסבר
getWeaponArray()	List<Weapon>	רשימת כלי הנשק המותקנים על חללית הקרב. עבור כל נשק נשמור את הנתונים הבאים: • שם כלי הנשק. • כוח-אש (ביחידות כוח-אש). • עלות תחזוקה שנתית (בדולרים).
getFirePower()	int	כוח האש המצטבר של חללית קרב הינו סכום כוח-האש של

		כל הנשקים המותקנים, בנוסף לכוח האש המובנה של כל חללית.
getNumberOfTechnicians()	int	מספר הטכנאים המוצבים על החללית (מספר שלם בטווח 0-5).
getAnnualMaintenanceCost()	int	עלות האחזקה השנתית של חללית קרב מסוג Bomber מורכבת מסכום הרכיבים הבאים: • עלות אחזקה שנתית בסיסית לספינת קרב מסוג Bomber (6000 דולר). • עלות האחזקה השנתית של כלי הנשק (עלות האחזקה של כל כלי הנשק המותקנים על חללית הקרב). שימו לב – כל טכנאי המוצב על החללית מוזיל את עלויות האחזקה השנתיות על כלי הנשק ב-10%. כלומר, עלות תחזוקת כלי הנשק מופחתת בשיעור של 0-50% כתלות במספר הטכנאים. יש לעגל את המחיר לשלמים אחרי חישוב ההוזלה ביחס לסכום עלויות כלי הנשק.

StealthCruiser

חללית קרב מהירה הכוללת גם יכולת חמקנות מתקדמת, משמשת למשימות סיור בעומק שטח האויב. נגדיר עבורה את כל השירותים של חללית קרב (Fighter), בנוסף להגדרות הבאות:

שם השירות	טיפוס החזרה	הסבר
getAnnualMaintenanceCost()	int	עלות האחזקה השנתית של חללית קרב מסוג StealthCruiser מורכבת מסכום הרכיבים הבאים: • עלות אחזקה שנתית של חללית קרב (Fighter). • תוספת שנתית בגין תחזוקה של מנוע החמקנות (Cloaking device). ○ על כל חללית מסוג StealthCruiser מותקן מנוע חמקנות יחיד. ○ עלות האחזקה השנתית של מנוע החמקנות תלויה במספר מנועי החמקנות הקיימים בצי. עלות האחזקה של כל מנוע חמקנות מחושבת כמספר המנועים בצי * 100 דולר פדרציה (למשל אם קיימים בצי 2 מנועי חמקנות, עלות האחזקה של כל אחד מהם היא 200 דולר-פדרציה). ○ ניתן להניח שמספר מנועי החמקנות בצי שווה למספר המופעים שנוצרו עבור סוג החללית StealthCruiser.

מה עליכם לעשות ?

1. הגדרת מנשקים (Interfaces) (5%)

הגדרת המנשק CrewMember

- הגדירו מנשק בשם **CrewMember** אשר ייצג איש צוות בצי החלל. על המנשק לכלול את המתודות getName(), getAge(), getYearsInService().
- על כל מחלקה המייצגת איש צוות לממש מנשק זה.

הגדרת המנשק Spaceship

- הגדירו מנשק בשם **Spaceship** אשר ייצג חללית בצי החלל. על המנשק לכלול את המתודות המאפיינות חללית כלשהי (היעזרו בטבלה המתארת Spaceship כללי לעיל).
- על כל מחלקה המייצגת חללית לממש מנשק זה.

✓ הגדרת מנשק מאפשרת לנו לעבוד בצורה אחידה עם מחלקות שונות המממשות אותו. למשל - נוכל ליצור אוסף פולימורפי המכיל אובייקטים של חלליות מסוגים שונים ולגשת אליהם בצורה אחידה דרך המתודות המוגדרות במנשק *Spaceship*.

✓ יש לממש את המנשקים ואת כל שאר המחלקות בתרגיל זה תחת החבילה starfleet.

2. הגדרת עץ ההורשה (5%)

- נתחו את הדמיון בין המחלקות השונות שהוגדרו לעיל עבור אנשי צוות ועבור חלליות, ובנו עצי הורשה מתאימים אשר יכללו מנשקים, מחלקות אבסטרקטיות, מחלקות קונקרטיות ומחלקות עזר אם קיימות.
- יחסי ההורשה בין המחלקות אמורים למנוע שכפול קוד בין מחלקות.
- שרטטו את היחסים בין המחלקות השונות על פי המוגדר בראש התרגיל והגישו את דיאגרמת המחלקות בקובץ התשובות.
- וודאו שהשרטוט שלכם מכיל את כל המחלקות הקונקרטיות המפורטות בתחילת הסעיף הבא.

3. מימוש המחלקות (40%)

בהתבסס על עצי ההורשה אותם הגדרתם ולפי פירוט המתודות שהובא בטבלאות לעיל, ממשו את המחלקות הבאות בתוך החבילה *starfleet*:

1) Crewman
2) Officer
3) MedicalShip
4) TransportShip
5) Fighter
6) Bomber
7) StealthCruiser
8) Weapon

- במידה ובחרתם להגדיר מחלקות אבסטרקטיות, ממשו גם אותן.
- בנאים - לכל מחלקה ניצור בנאי המקבל את כל הפרמטרים הנדרשים לאתחול שדות המחלקה. ממשו את הבנאים על פי החתימות הבאות (ניתן להניח שהקלט לבנאים תקין):

```
public Crewman(String name, int age, int yearsInService)

public Officer(String name, int age, int yearsInService, OfficerRank rank)
```

```
public MedicalShip(String name, int commissionYear, float maximalSpeed,
Set<CrewMember> crewMembers, int numberOfMedicalClinics)

public TransportShip(String name, int commissionYear, float maximalSpeed,
Set<CrewMember> crewMembers, int cargoCapacity, int passengerCapacity)

public Fighter(String name, int commissionYear, float maximalSpeed,
Set<CrewMember> crewMembers, List<Weapon> weaponArray)

public Bomber(String name, int commissionYear, float maximalSpeed,
Set<CrewMember> crewMembers, List<Weapon> weaponArray, int numberOfTechnicians)

public StealthCruiser(String name, int commissionYear, float maximalSpeed,
Set<CrewMember> crewMembers, List<Weapon> weaponArray)

public Weapon(String name, int firePower, int annualMaintenanceCost)
```

- העמסת בנאים – עבור המחלקה `StealthCruiser`, צרו בנאי נוסף בעל החתימה :

```
public StealthCruiser(String name, int commissionYear, float maximalSpeed,
Set<CrewMember> crewMembers)
```

היות ומרבית החלליות מסוג `StealthCruiser` יכללו רק תותחי לייזר סטנדרטיים בתור חימוש, בנאי זה (אשר אינו מקבל את הנשקים כפרמטר) יצור אובייקט המייצג חללית מסוג `StealthCruiser` עם מערך נשקים הכולל את האובייקט הבא בלבד:

```
new Weapon ("Laser Cannons",10,100)
```

על בנאי זה לעשות שימוש בבנאי המלא שהוגדר קודם לכן (שימו לב שקריאה לבנאי אחר של המחלקה יכולה להיעשות רק מהשורה הראשונה של הבנאי...רמז: יתכן ותרצו להשתמש ב-Arrays.asList).

- מתודת toString() – בכל אחת מהמחלקות עליכם לממש מתודת toString() המחזירה מחרוזת המתארת את נתוני המחלקה.
  - המחרוזת תתחיל בשם המחלקה, ואח"כ מוסטים לימין ע"י טאב בודד יופיעו נתוני המחלקה לפי הסדר והפורמט המודגמים בהמשך (סדר הופעת השדות יהיה: שדות המחלקה המשותפים לכל סוגי החלליות, אח"כ תופיע עלות האחזקה השנתית, ולאחר מכן השדות הספציפיים לאותה המחלקה).
  - מתודת ה-toString() עשויה לקרוא למתודה באותו שם במחלקת האם.
  - הקפידו שהמחרוזות שנוצרות יהיו **זהות** לאלו המוצגות בסוף התרגיל או בקובץ הפלט הנלווה.

הנה דוגמא למחרוזת המיוצרת ע" מתודת ה-toString של מחלקת `TransportShip`:

```
TransportShip
	Name=USS Peres
	CommissionYear=2396
	MaximalSpeed=5.1
	FirePower=10
	CrewMembers=23
	AnnualMaintenanceCost=20000
	CargoCapacity=2000
	PassengerCapacity=5000
```

- <u>דריסת המתודות equals() ו-hashCode()</u>

- **public boolean** equals(Object obj)
- **public int** hashCode()

על מנת שנוכל לאחסן אובייקטים של מחלקות שיצרנו במבני נתונים המבוססים על HashTable יש לדרוס את המתודות equals() (הבודקת זהות מול אובייקט אחר) ואת המתודה hashCode() (המחזירה ערך גיבוב). <u>וודאו שכל מחלקה המייצגת איש צוות או חללית מכילה דריסה של 2 מתודות אלו</u> (אך הימנעו משכפול קוד מיותר תוך שימוש בהורשה).

- ✓ שימו לב ששדה השם מהווה ערך מזהה ייחודי עבור אנשי צוות ועבור חלליות.
- ✓ היעזרו באקליפס ליצירה אוטומטית של מתודות אלו (**Source>Generate hashCode() and equals()...**), אך וודאו שאתם מבינים את הקוד שנוצר.

- <u>תמיכה במיון של אובייקטים מסוג איש צוות או חללית</u>

ייתכן ותרצו שהמחלקות שיצרתם יממשו את המנשק Comparable כדי שניתן יהיה להשתמש בהן עם מתודות או מבני נתונים הדורשים הגדרת יחס סדר (כגון Collection.sort או כמפתחות של TreeMap). לחילופין, תוכלו בהמשך להגדיר מחלקת עזר חיצונית המממשת את המנשק Comparator ולספק אותה כמגדירת יחס סדר למתודה או לבנאי של מבנה הנתונים הרלבנטי.

<u>הערות כלליות לסעיף זה:</u>

- אתם רשאים להוסיף שדות, מתודות ומחלקות עזר נוספות בכל אחת מהמחלקות שלכם כל זמן שאתם לא פוגעים בחתימות ובמנשק המוגדרים לעיל.
- שימו לב לנראות השדות בכל אחד משלבי היררכיית הירושה. לא ניתן לגשת לשדות המוגדרים כפרטיים במחלקת האם.
- הקפידו להשתמש בקבועים כשאלו נדרשים.

4. <u>StarfleetManager (50%)</u>

מחלקה זו (עבורה נתון לכם השלד) תכיל מספר מתודות סטטיות המקבלות אוסף חלליות ומחזירות חיתוכים שונים על פי הפירוט הבא:

```
1. public static List<String> getShipDescriptionsSortedByShipName
   (List<Spaceship> fleet)
```

(5%) המתודה תחזיר רשימה של מחרוזות המתארות את חלליות הצי, כאשר החלליות ממוינות בסדר עולה לפי שם החללית. כל איבר ברשימה המוחזרת יהיה מחרוזת שהינה תוצר של מתודת ה-toString() של אובייקט החללית המתאים.

- ✓ לצורך מיון החלליות על פי שם החללית עליכם להגדיר מחלקה שתממש את המנשק Comparator ולספק אותה כפרמטר למתודה Collections.sort.

```
2. public static Map<String, Integer> getInstanceNumberPerClass
   (List<Spaceship> fleet)
```

(5%) המתודה תחזיר מפה המכילה עבור כל שם מחלקה של חללית את מספר האובייקטים שנוצרו מהמחלקה (רק אם נוצרו, אין צורך לכלול מחלקות שלא נוצרו מהן אובייקטים).

- ✓ ניתן להשתמש במתודה getClass() על כל אובייקט כדי לדעת מאיזו מחלקה הוא (מקבלים חזרה אובייקט מסוג Class ואז ניתן לקבל את שם המחלקה באמצעות המתודה getSimpleName()). גם האופרטור instanceof עשוי להיות שימושי במימוש מתודה זו.
- ✓ שימו לב שניתן לממש מתודה זו בשתי דרכים: האחת כוללת מעבר על רשימת החלליות הניתנת כפרמטר למתודה, והשניה כוללת החזקת מונים סטטיים באחת או יותר מהמחלקות הרלבנטיות וקידומם בבנאי(ם) בעת יצירת אובייקט חדש. וודאו שאתם מבינים את שתי הדרכים ובחרו בזו שנראית לכם יותר אלגנטית לצורך המימוש שלכם.

```
3. public static int getTotalMaintenanceCost (List<Spaceship> fleet)
```

(5%) המתודה תחזיר את סך כל עלויות האחזקה של כל חלליות הצי ע"י סכימת עלויות האחזקה של כל חללית בצי.

```
4. public static int getTotalFleetFirePower (List<Spaceship> fleet)
```

(5%) המתודה תחזיר את סך כל כוח-האש של חלליות הצי ע"י סכימת כוח-האש של כל חללית בצי.

```
5. public static Set<String> getFleetWeaponNames (List<Spaceship>
   fleet)
```

(5%) המתודה תחזיר אוסף מסוג קבוצה המכיל מחרוזות המייצגות את שמות כלי הנשק השונים (ללא חזרות) המותקנים על חלליות הצי.

```
6. public static int getTotalNumberOfFleetCrewMembers(List<Spaceship>
   fleet)
```

(5%) המתודה תחזיר את מספר אנשי הצוות הכולל בצי (סכום אנשי הצוות המוצבים בכל חללית)

```
7. public static float getAverageAgeOfFleetOfficers(List<Spaceship>
   fleet)
```

(5%) המתודה תחזיר את הגיל הממוצע של קציני הצי (ניתן להניח שעל כל חללית קיים קצין אחד לפחות).

```
8. public static Map<Officer, Spaceship>
   getHighestRankingOfficerPerShip(List<Spaceship> fleet)
```

(5%) המתודה תמצא את הקצין בעל הדרגה הבכירה ביותר המוצב על כל חללית בצי, ותחזיר מפה הממפה לכל קצין כזה את החללית בה הוא מוצב.

שימו לב שהטיפוס OfficerRank שמסופק לכם, הוא Enum אשר מונה את דרגות הקצונה על פי סדר הבכירות שלהן. Enum בג'אווה מממש את המנשק Comparable ועל כן ניתן למיין לפיו.

```
9.  public static List<Map.Entry<OfficerRank, Integer>>
    getOfficerRanksSortedByPopularity (List<Spaceship> fleet)
```

(10%) המתודה תחזיר רשימה של אובייקטים מסוג Map.Entry המכילים זוגות של דרגה, ומספר המופעים שלה בקרב קציני הצי. הרשימה המוחזרת תהיה ממוינת בסדר יורד לפי מספר מופעי הדרגה בצי.

הנחיה: בנו מפה אשר מאחסנת עבור כל דרגה את מספר המופעים שלה, אח"כ השתמשו ב-Map.getEntries() לקבלת אוסף זוגות המייצג את המפה, העבירו את האוסף לרשימה, מיינו את הרשימה והחזירו אותה.

5. StarfleetManagerTester

המחלקה StarfleetManagerTester מייצרת צי של חלליות על צוותיהן ומשתמשת בכל המחלקות והמתודות שכתבתם כדי להדפיס דוח מסכם למסך. לאחר שסיימתם את מימוש כל המחלקות, הריצו את המחלקה StarfleetManagerTester שמסופקת לכם בשלמותה ובדקו שהפלט המודפס על ידכם זהה לפלט המצורף לקבצי התרגיל (בכל מקום שבו מופיעה הזחה, ניתן להניח שמדובר בטאב בודד).

- ✓ המחלקה StarfleetManagerTester מייצרת צי חלליות וצוותים בצורה אוטומטית (אך לא רנדומאלית, על מנת שתוכלו לקבל פלט זהה לשלנו בכל הרצה).
- ✓ שמות אנשי הצוות והחלליות אמורים להיות ייחודיים ועל כן יצרנו שמות מהצורה "James #121".

בהצלחה !