

# OOP Exercise

## מטרה:

בתרגיל זה נכתוב אפליקציה המתארת משחק מרדף בין שוטרים ופושעים.  
ארכיטקטורת האפליקציה תעזור לנו בהבנת נושא "ירושה" / Inheritance.

## דרישות:

על התרגיל להיות מסודר וקריא. יש להקפיד על תיעוד במקומות הנדרשים - על פי שיקול דעתכם ובהתחשב בעובדה שמתכנת אחר ייקרא את הקוד שלכם בעתיד.  
הקפידו על מוסכמות השפה כפי שנלמדו בשיעורים עד כה.

## הנחיות:

לפניכם הנחיות כלליות לכתיבת האפליקציה. אמנם, על פי שיקול דעתכם תוכלו להשתמש בפונקציות עזר לבחירתכם ובכלים מתקדמים - גם במידה ועדיין לא נלמדו.

האפליקציה תכיל את האובייקטים הבאים:

1. Player - שחקן.

2. Policeman - שוטר ( יורשת מ Player ).

3. Criminal - פושע ( יורשת מ Player ).

כתיבת המחלקה Player:

מחלקה אבסטרקטית המתארת שחקן באפליקציה.

תכיל את השדות הבאים:

1. שדה מטיפוס String בשם name השומר את שם השחקן.

2. שדה מטיפוס int בשם strength השומר את כוחו של השחקן.

תכיל את הפונקציות הבאות:

1. בנאי שחתימתו:

```
public Player(String name, int strength)
```

המאתחל את שדות המחלקה בהתאם.

2. פונקציה שחתימתה:

```
public String toString()
```

המחזירה String המתאר את שמו וכוחו של השחקן.

3. פונקציה שחתימתה:

```
protected boolean isAlive()
```

המחזירה true במידה וכוחו של השחקן גדול מ 0, או false אחרת.

4. פונקציה שחתימתה:

```
protected int knockOut()
```

המביאה את מצב השחקן לנוק-אווט : פונקציה זו תוריד את כוחו של השחקן ל 0 ותחזיר את כוחו שהיה לפני ההורדה. כלומר, לדוגמא: אם היה כוחו של השחקן 6 - תוריד הפונקציה את כוחו ל 0 ותחזיר את הערך 6.

כתיבת המחלקה Criminal:

המחלקה יורשת מהמחלקה Player.

תכיל את הפונקציות הבאות:

1. בנאי שחתימתו:

```
public Criminal(String name, int strength)
```

המאתחל את שדות המחלקה בהתאם.

2. בנאי העתקה שחתימתו:

```
public Criminal(Criminal criminal)
```

המאתחל את שדות המחלקה בהתאם לאלו המתקבלים בארגומנט criminal.

כתיבת המחלקה Policeman:

המחלקה יורשת מהמחלקה Player.

תכיל את השדות הבאים:

1. שדה מטיפוס Criminal[] בשם arrestedCriminals השומר את הפושעים שהצליח לעצור שוטר זה.

2. שדה מטיפוס int בשם totalArrested השומר את מספר הפושעים העצורים במערך.

3. שדה מטיפוס int בשם MAX\_NUM\_CRIMINALS השומר את מספר הפושעים המקסימלי ששוטר זה מסוגל לעצור.

תכיל את הפונקציות הבאות:

1. בנאי שחתימתו:

```
public Policeman(String name, int strength)
```

המאתחל את שדות המחלקה בהתאם.

2. פונקציה שחתימתה:

```
public boolean arrestedAll()
```

המחזירה true במידה ומספר הפושעים העצורים הגיע למקסימום, או מחזירה false אחרת.

3. פונקציה שחתימתה:

```
public boolean arrest(Criminal criminal)
```

המחזירה true במידה והצליח השוטור לעצור את הפושע, או false אחרת.

להלן תיאור של תהליך המעצר:

a. במידה והשוטר בחיים (*isAlive*) וגם הפושע בחיים בדוק:

i. אם עוד נותר מקום במערך העצורים של השוטור בצע:

1. אם כוחו של השוטור גדול מכוחו של הפושע:

a. הבא את הפושע למצב נוק-אווט, החסר את כוחו של

הפושע מכוחו של השוטור, עצור את הפושע והחזר **true**.

2. אחרת, אם כוחו של הפושע גדול מכוחו של השוטור:

a. הבא את השוטור למצב נוק-אווט והחסר את כוחו מכוחו

של הפושע. כלומר, במידה וכוחו של השוטור היה 2 וכוחו

של הפושע היה 6, כעת הכוחות יהיו - שוטור: 0 פושע: 4.

3. אחרת, כלומר - כוחו של הפושע שווה לכוחו של השוטור:

a. הבא את השוטור ואת הפושע למצב נוק-אווט.

b. החזר **false**.

4. פונקציה שחתימתה:

```
public void showArrestedCriminals()
```

במידה והשוטר עצר פושע אחד או יותר, תדפיס הפונקציה את שמו של השוטור ורשימת הפושעים שעצר.

במידה ולא עצר אף פושע, תדפיס הפונקציה את שמו של השוטור ותדווח על כך שלא עצר אף פושע.

מומלץ להשתמש בקוד הבא בפונקציה main בכדי לבדוק את האפליקציה:

```
public static void main(String[] args) {
    Criminal[] criminals = new Criminal[6];
    /*Initialize list of criminals*/
    criminals[0] = new Criminal("Damien", 4);
    criminals[1] = new Criminal("Bella", 5);
    criminals[2] = new Criminal("Griselda", 5);
    criminals[3] = new Criminal("Spike", 6);
    criminals[4] = new Criminal("Lucifer", 8);
    criminals[5] = new Criminal("Baruch", 29);

    Policeman[] policemen = new Policeman[3];
    /*Initialize list of police men*/
    policemen[0] = new Policeman("David", 15);
    policemen[1] = new Policeman("James", 19);
    policemen[2] = new Policeman("Jax", 50);

    /*For every police man, */
    for (int i = 0; i < policemen.length; i++) {
        /*try to arrest all the criminals*/
        for (int j = 0; j < criminals.length; j++) {
            /*If arrested successfully*/
            if (policemen[i].arrest(criminals[j])) {
                /*Print to console*/
                System.out.println(policemen[i] + " arrested " + criminals[j]);
            }
        }
    }

    /*Show arrested criminals of each police man*/
    for (int i = 0; i < policemen.length; i++) {
        policemen[i].showArrestedCriminals();
    }
}
```