

תרגיל 5 : מערכת לניהול חיות בגן חיות 🌸

♦ חלק א – יצירת היררכיה של מחלקות

🌸 שלב 1: הגדר מחלקת בסיס Animal

תכונות (attributes):

- name שם החיה (str)
- species סוג החיה (str)
- age גיל (int)

מתודה:

- describe(): מחזירה מחרוזת עם פרטי החיה, למשל:

```
python
Edit Copy
"Polly is a 3-year-old Parrot."
```

- make_sound(): מחזירה מחרוזת גנרית:

```
python
Edit Copy
"Some generic animal sound"
```

🌸 שלב 2: צור מחלקת בת Bird שיורשת מ־Animal

מאפיינים נוספים:

- wing_span האורך הכולל בין קצה כנף אחת לקצה הכנף השנייה כשהכנפיים פרושות לגמרי (int)

מתודות חדשות:

- describe_wings() מחזירה מחרוזת כמו:

```
python
Edit Copy
"Polly has a wingspan of 25 cm."
```

♦ חלק ב – קומפוזיציה: מחלקה בתוך מחלקה

:Override

- עוקף את make_sound() כך שתדפיס:

```
python
"Tweet tweet!"
```

✿ שלב 3: צור מחלקה Caretaker

מאפיינים:

- name שם המטפל (str)
- experience_years מספר שנות ניסיון (int)

מתודה:

- describe() מחזירה מחרוזת עם פרטי המטפל, למשל:

```
python
"Dina has 5 years of experience caring for animals."
```

✿ שלב 4: הוסף את Caretaker אל תוך Bird

הוסף את המחלקה Caretaker לתוך תכונה של Bird.

♦ חלק ג – ארגון לקבצים (מודולים)

✿ שלב 5: ארגן את המחלקות בקובץ נפרד

1. צור קובץ בשם zoo.py
כלול בו את שלוש המחלקות:

```
python

Animal, Bird, Caretaker
```

2. צור קובץ בשם main.py

יבוא ממנו את המחלקה Bird בלבד וייבצע את הפעולות הבאות:

```
python

from zoo import Bird

parrot = Bird("Polly", "Parrot", 3, 25, "Dina", 5)

print(parrot.describe())           # Polly is a 3-year-old Parrot.
print(parrot.describe_wings())     # Polly has a wingspan of 25 cm.
print(parrot.make_sound())         # Tweet tweet!
print(parrot.caretaker.describe()) # Dina has 5 years of experience caring for animals.
```

חלק ד – שילוב עם OpenAI API

✿ שלב 6: צור קובץ chat.py

- בקש קלט מהמשתמש (שם של חיה)
- תשלח את הקלט ל- ChatGPT דרך API
- הצג את התשובה
- כעת, הוסף מתודה למחלקת animal כך שתוכל לקבל פרטים אודות החיה (כלומר זאת תהא מתודה שמאפשרת לך לקבל פרטים אודות החיה דרך chatgpt).