## Vet

Create a class called **Vet**. Upon initialization it should receive a **name** (string). It should also have an **instance attribute** called **animals** (empty list by default). There should also be **2 class attributes**: **animals** (empty list) which will store the total amount of **animals for all vets**; **space** (5 by default). You should create **3 instance methods**

* **register\_animal(animal\_name)**
  + If there **is space** in the vet clinic add the animal to **both animals' lists** and return a message: **"{name} registered in the clinic"**
  + Otherwise return **"Not enough space"**
* **unregister\_animal(animal\_name)**
  + If the animal is **in the clinic**, **remove** it from **both animals** lists and return **"{animal} unregistered successfully"**
  + Otherwise, return **"{animal} not in the clinic"**
* **info()** – returns: **"{vet\_name1} has {number\_animals} animals. {space\_left\_in\_clinic} space left in clinic"**

### Examples

|  |  |
| --- | --- |
| **Test Code** | **Output** |
| peter = Vet("Peter")  george = Vet("George")  print(peter.register\_animal("Tom"))  print(george.register\_animal("Cory"))  print(peter.register\_animal("Fishy"))  print(peter.register\_animal("Bobby"))  print(george.register\_animal("Kay"))  print(george.unregister\_animal("Cory"))  print(peter.register\_animal("Silky"))  print(peter.unregister\_animal("Molly"))  print(peter.unregister\_animal("Tom"))  print(peter.info())  print(george.info()) | Tom registered in the clinic  Cory registered in the clinic  Fishy registered in the clinic  Bobby registered in the clinic  Kay registered in the clinic  Cory unregistered successfully  Silky registered in the clinic  Molly not in the clinic  Tom unregistered successfully  Peter has 3 animals. 1 space left in clinic  George has 1 animals. 1 space left in clinic |