Code assignment Backend Chapter

Welcome to the code assignment of Singtel.   You are expected to spend 1 to 1.5 hours on the assignment. If you run out of time, please submit the results as far as you got within this amount of time . We value quality over quantity.

Please create a repository in Github   and share the link with results .   To set the correct expectations: There is no "right" or "wrong" answer. We would like to see how you model the objects in the assignment and learn your coding style. Stay close to how you would code if this were an assignment in your regular day job.

Furthermore:

- Please keep a fine grained commit history with separate commits for each question.
- Please create meaningful Unit Tests .
- Please modify the code snippets provided to optimize for maintainability . Be aware that the code snippets are deliberately flawed. We like to see you improve them.
- A README.md with basic documentation and a simple visualisation is appreciated . A picture of a whiteboard or drawing on a paper is sufficient here.
- You are encouraged to be opinionated how you solve this. Show us your way of working! Take creative liberty where you feel it is needed.

A. Let's start with the basics

Given the following code...

```
class Animal {

    void walk(){
        System.out.println("I am walking");
    }

}

class Bird extends Animal {

    void fly() {
        System.out.println("I am flying");
    }

}

public class Solution {

    public static void main(String[] args) {
        Bird bird = new Bird();
```

```
          bird.walk();
          bird.fly();
          bird.sing();
      }
}
```

1. Can you implement the sing() method for the bird?
   1. How did you unit test it?
   2. How did you optimize the code for maintainability? *(Ask yourself the same question for all following exercises)*
2. Now, we have 2 special kinds of birds: the **Duck** and the **Chicken** ... Can you implement them to make their own special sound?
   1. A duck says: "Quack , quack "
   2. A duck can swim
   3. A chicken says: " Cluck, cluck "
   4. A chicken cannot fly (assumption: its wings are clipped but ignore that)
3. Now how would you model a rooster?
   1. A rooster says: " Cock-a-doodle-doo "
   2. How is the rooster related to the chicken?
   3. Can you think of other ways to model a rooster without using inheritance?
4. Can you model a parrot? We are specifically interested in three parrots, one that lived in a house with dogs one in a house with cats , the other lived on a farm next to the rooster.
   1. A parrot living with dogs says: "Woof, woof"
   2. A parrot living with cats says: " Meow"
   3. A parrot living near the rooster says: " Cock-a-doodle-doo "
   4. How do you keep the parrot maintainable? What if we need another parrot lives near a Duck? Or near a phone that rings frequently?

B. Model fish as well as other swimming animals

1. In addition to the birds, can you model a fish?
   1. Fishes don't sing
   2. Fishes don't walk
   3. Fishes can swim
2. Can you specialize the fish as a Shark and as a Clownfish?
   1. Sharks are large and grey
   2. Clownfish are small and colourful (orange)
   3. Clownfish make jokes
   4. Sharks eat other fish
3. Dolphins are not exactly fish, yet, they are good swimmers
   1. Can you model a dolphin that swims without inheriting from a fish class?
   2. How do you avoid duplicating code or introducing unneeded overhead?

C. Model animals that change their behaviour over time

1. Can you model a butterfly?
    1. A butterfly can fly
    2. A butterfly does not make a sound
2. Can you optimize your model to account for the metamorphosis from caterpillar to butterfly?
    1. A caterpillar cannot fly
    2. A caterpillar can walk (crawl)

D. Counting animals

Suppose you have an array of animals, e.g.

```
Animal[] animals = new Animal[]{
        new Bird(),
        new Duck(),
        new Chicken(),
        new Rooster(),
        new Parrot(),
        new Fish(),
        new Shark(),
        new Clownfish(),
        new  Dolhpin (),
        new  Frog (),
        new Dog(),
        new  Butterfly (),
        new Cat()
};
```

Note : The above instantiation may be different if you chose to set up your object model differently…   (hopefully you did)

1. Can you share the code to count:
    1. How many of these animals can fly?
    2. How many of these animals can walk?
    3. How many of these animals can sing?
    4. How many of these animals can swim?

BONUS

If you still have time left, please consider the following:

1. Can you add a second language (if you kno w a language other than English) Use the rooster as a PoC for demonstrating this. For example, this is how the Rooster sounds differently depending on language. Please add the rooster sound in your native tongue.
    - Danish: kykyliky
    - Dutch: kukeleku
    - Finnish: kukko kiekuu
    - French: cocorico
    - German: kikeriki
    - Greek: kikiriki
    - Hebrew: coo- koo - ri-koo
    - Hungarian: kukuriku
    - Italian: chicchirichi
    - Japanese: ko - ke - kok - ko -o
    - Portuguese: cucurucu
    - Russian: kukareku
    - Swedish: kuckeliku
    - Turkish: kuk-kurri-kuuu
    - Urdu: kuklooku
2. Can you design a RESTful API for querying these animals?