Inheritance (subclasses)

Explanation

Ensure you refer to and understand the Classes cheatsheet first!

Subclasses are *extensions* of other classes - known as superclasses. For example, if a class called `Animal` exists, it may have some base attributes like `name`, `age`, `colour`, `owner`, etc. But then, a subclass, say `Cat` might exist. This could have attributes like `lives`, `relationship_to_dogs` etc.

Similarly, the `Animal` class might have getter & setter functions for its properties, and the `Cat` subclass would have getters and setters for its *extra* properties. The subclass would also feature its own constructor.

The subclass would inherit all attributes and methods of its superclass. It could also have its own methods added onto it.

Python

Here is the example used above in Python:

```python
# Superclass
class Animal:
        def __init__(self, name, age, colour, owner):
                self.name = name
                self.age = age
                self.colour = colour
                self.owner = owner

        def get_name(self):
                return self.name

        def get_age(self):
                return self.age

        def get_colour(self):
                return self.colour

        def get_owner(self):
                return self.owner

        def set_name(self, name):
                self.name = name

        def set_age(self, age):
                self.age = age

        def set_colour(self, colour):
                self.colour = colour

        def set_owner(self, owner):
                self.owner = owner
```

```python
class Cat(Pet):
        def __init__(self, name, age, colour, owner, lives, relationship_to_dogs):
                super().__init__(name, age, colour, owner)
                self.lives = lives
                self.relationship_to_dogs = relation_to_dogs

        def get_lives(self):
                return self.lives

        def get_relationship_to_dogs(self):
                return self.relationship_to_dogs

        def set_lives(self, lives):
                self.lives = lives

        def set_relationship_to_dogs(self, relationship_to_dogs):
                self.relationship_to_dogs(relationship_to_dogs)
```

TypeScript

```typescript
class Animal {
    name: string;
    age: number;
    colour: string;
    owner: string;

    constructor(name: string, age: number, colour: string, owner: string) {
        this.name = name;
        this.age = age;
        this.colour = colour;
        this.owner = owner;
    }

    getName(): string {
        return this.name;
    }

    getAge(): number {
        return this.age;
    }

    getColour(): string {
        return this.colour;
    }

    getOwner(): string {
        return this.owner;
    }

    setName(name: string): void {
        this.name = name;
    }

    setAge(age: number): void {
        this.age = age;
    }

    setColour(colour: string): void {
```

```
            this.colour = colour;
        }

        setOwner(owner: string): void {
            this.owner = owner;
        }
    }

    class Cat extends Animal {
        lives: number;
        relationshipToDogs: string;

        constructor(name: string, age: number, colour: string,
                              owner: string, lives: number, relationshipToDogs: string)
            {
            super(name, age, colour, owner);
            this.lives = lives;
            this.relationshipToDogs= relationshipToDogs;
        }

        getLives(): number {
            return this.lives;
        }

        getRelationshipToDogs(): string {
            return this.relationshipToDogs;
        }

        setLives(lives: number): void {
            this.lives = lives;
        }

        setRelationshipToDogs(relationshipToDogs: string): void {
            this.relationshipToDogs = relationshipToDogs;
        }
    }
```

C#

```
using System;

public class Animal {
    public string name;
    public int age;
    public string colour;
    public string owner;

    public Animal(string name, int age, string colour, string owner) {
        this.name = name;
        this.age = age;
        this.colour = colour;
        this.owner = owner;
    }

    public string GetName() {
        return this.name;
    }

    public int GetAge() {
```

```csharp
            return this.age;
    }

    public string GetColour() {
        return this.colour;
    }

    public string GetOwner() {
        return this.owner;
    }

    public void SetName(string name) {
        this.name = name;
    }

    public void SetAge(int age) {
        this.age = age;
    }

    public void SetColour(string colour) {
        this.colour = colour;
    }

    public void SetOwner(string owner) {
        this.owner = owner;
    }
}

public class Cat : Animal {
    public int lives;
    public string relationshipToDogs;

    public Cat(string name, int age, string colour,
               string owner, int lives, string relationshipToDogs)
        : base(name, age, colour, owner) {
        this.lives = lives;
        this.relationshipToDogs = relationshipToDogs;
    }

    public int GetLives() {
        return this.lives;
    }

    public string GetRelationshipToDogs() {
        return this.relationshipToDogs;
    }

    public void SetLives(int lives) {
        this.lives = lives;
    }

        public void SetRelationshipToDogs(string relationshipToDogs)
        {
                this.relationToDogs = relationshipToDogs;
        }
}
```