

Elektronikschule Tett nang	Softwareentwicklung	Datum:	Klasse:
	Übung zu abstrakten Klassen und Methoden überschreiben	Name:	

Die „Animal“ Klasse repräsentiert ein Tier mit einem Hunger-Level der sich verändert, abhängig von den Aktivitäten des Tieres.

<i>{abstract} Animal</i>	
- hungerlevel : int	// 0 bedeutet nicht hungrig
+ Animal() // hungerlevel ist 0	
+ Animal(int aHungerLevel)	
+ <i>abstract</i> play() : void	
+ <i>abstract</i> speak() : String	
+ getHungerLevel() : int	
+ setHungerLevel(int aLevel) : void	
+ eat() : void	// setzt das hungerlevel auf 0
+ toString() : String	

Erstellen Sie zwei Unterklassen („Dog“ und „Cat“) der abstrakten „Animal“-Klasse.

Die abstrakten Methoden play() und speak() müssen sowohl in der Klasse „Dog“ als auch in der Klasse „Cat“ implementiert werden.

Betrachten Sie die anderen Methoden und entscheiden Sie welche Methoden ohne Veränderung vererbt werden können, welche überschrieben werden müssen und welche neuen Methoden hinzugefügt werden müssen um den folgenden Eigenschaften der Tiere Hund und Katze gerecht zu werden:

- Hunde haben einen Namen
- Hunde haben einen anfänglichen Hungerlevel von 30
- Hunde verdoppeln ihren Hungerlevel wenn sie spielen
- Hunde halbieren ihren Hungerlevel wenn sie essen (Benützen Sie Integerdivision)
- Hunde sagen „Grrr“, wenn Ihr Hungerlevel ≥ 50 ist, andernfalls sagen sie „woof“
- Hunde können Tricks ausführen, ein Trick erhöht jedoch den Hungerlevel um 20
- Katzen haben einen Namen
- Katzen haben einen anfänglichen Hungerlevel von 10
- Katzen erhöhen ihren Hungerlevel um 20 wenn sie spielen
- Katzen sind ganz satt, wenn sie einmal essen (Hungerlevel 0)
- Katzen sagen „REOW“, falls hungerlevel ≥ 85
- Katzen sagen „meow“, falls hungerlevel < 85 und hungerlevel ≥ 30
- Katzen sagen ansonsten „purrrr“
- Katzen können Mäuse fangen. Diese Aktion halbiert den Hungerlevel (Integerdivision)

Schreiben Sie ein Demoprogramm.