

# Programmentwicklung Advanced

## Repetition Einführung I

### Aufgaben

---

#### 1. Aufgabe

Was bedeutet Polymorphie?

Das methoden Typenspezifisch neu definiert werden.  
(verhalten ist unterschiedlich = polimorphth)

#### 2. Aufgabe

Was wird auf der Konsole ausgegeben?

```
public class Vehicle {  
    public void Move() {  
        Console.WriteLine("moving");  
    }  
}  
  
public class Boat : Vehicle {  
    public new void Move() {  
        Console.WriteLine("swimming");  
    }  
}  
  
public class Jetski : Boat {  
    public new void Move() {  
        Console.Write("speed");  
        base.Move();  
    }  
}  
  
public static void Main() {  
    Vehicle vehicle = new Jetski();  
    vehicle.Move();  
}  
  
moving
```

**3. Aufgabe****Was wird auf der Konsole ausgegeben?**

```
public class Vehicle {  
    public virtual void Move() {  
        Console.WriteLine("moving");  
    }  
}  
  
public class Boat : Vehicle {  
    public override void Move() {  
        Console.WriteLine("swimming");  
    }  
}  
  
public class Jetski : Boat {  
    public override void Move() {  
        Console.Write("speed");  
        base.Move();  
    }  
}  
  
public static void Main() {  
    Vehicle vehicle = new Jetski();  
    vehicle.Move();  
}  
  
speedswimming
```

**4. Aufgabe****Was ist der Unterschied zwischen einer abstrakten Klasse und einem Interface?**

Eine abstrakte Klasse kann Methoden-Implementiert werden.  
(pure abstract class = interface)

**5. Aufgabe****Was könnte der Grund dafür sein, dass eine Klasse in C# zwar mehrere Interfaces implementieren, jedoch nur von einer einzigen Klasse erben kann?**

Interfaces beschreiben nur was die Klasse haben muss.

**6. Aufgabe****Was ist der Nutzen von abstrakten Klassen, bzw. in welchen Situationen werden Sie verwendet?**

Wenn gemeinsame Methoden und Properties sind, man diese als solches jedoch nicht instanzieren können soll