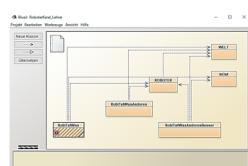


Wir programmieren mit

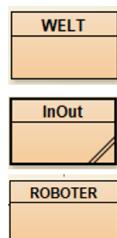
Java , C++, C#, Delphi, Visual Basic

```
System.out.print(info + " ");
try {
    InputStreamReader strom = new InputStreamReader(strom);
    BufferedReader buffer = new BufferedReader(buffer);
    String s = buffer.readLine();
    return s.charAt(0);
}
catch( IOException ex ) {
    System.out.println( ex.getMessage() );
    return ' ';
}
```



erzeugt aus dem Programmcode Java-Programme

Blue-J , Java Editor, Eclipse, NetBeans



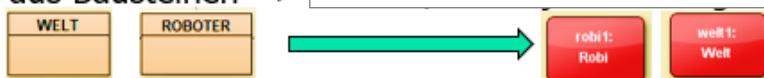
(fertige nutzbare Programmteile)

Karol [Welt, Roboter], InOut, String...

Wir nutzen ihre Funktionen, ohne den Inhalt kennen zu müssen.

Wir programmieren **objektorientiert** - das bedeutet:

(1) Programm besteht aus Bausteinen =>
z.B.



(2) Objekte haben Eigenschaften/Attribute =>
mit konkreten Werten => Attributwerte
z.B. farbeZiegel="blau"

(3) Objekte werden mit oder ohne Attributwerte erzeugt=>
z.B. public Robi (x, y, welt) { }

(4) Objekte können etwas tun =>
z.B. hebeAuf();

(5) Objekte können ihren Zustand anzeigen =>
z.B. int gibZahlZiegelVorn();

Quelltext einer Klasse

```
public class Klassenname          Klassenname beginnt mit Großbuchstaben
{
    // hier Attribute (globale Variablen) festlegen
    int var;                      Variablenname beginnt mit Kleinbuchstaben
    String text;
    public Klassenname()
    {// der Konstruktor erzeugt Objekte
        // kann auch leer bleiben
    }
    public void methodenName1() Methodenname beginnt mit Kleinbuchstaben
    {
        // Vorlage einer verändernden Methode...
    }
    public int methodenName2()
    {// Vorlage einer sondierenden Methode...
        var=5;
        return var;
    }
}
```