

Idee, Zweck, Systembeschreibung:

In der Schule benötigen die *Personen* einen *Geheimnachrichten-Generator*. Hierfür soll es möglich sein, auf einer *Benutzeroberfläche* (*Hauptfenster*) nach der Eingabe der *Nachricht* (im Eingabebereich) eine *geheime Nachricht erstellen* (*verschlüsseln*) und auf der Benutzeroberfläche (*Hauptfenster*) *anzeigen*. Beide Schritte erfolgen durch das Anklicken einer Schaltfläche *Nachricht erstellen* (*verschlüsseln*). Eine weitere Schaltfläche *Eingaben löschen* soll dafür sorgen, dass alle gemachten Eingaben *gelöscht* werden. Die *Person* kann auch geheime *Nachrichten lesen* (*entschlüsseln*), indem Sie die dafür vorgesehene Schaltfläche anklickt. Die entschlüsselte *Nachricht* wird daraufhin im Ausgabebereich *angezeigt*. Die Ver- und Entschlüsselung (sind Methoden) der *Nachricht* soll einem bestimmten Schema (ROT13-Algorithmus) folgen. Eine Erklärung dazu folgt im Anschluss an die vorgegebene System-Architektur.

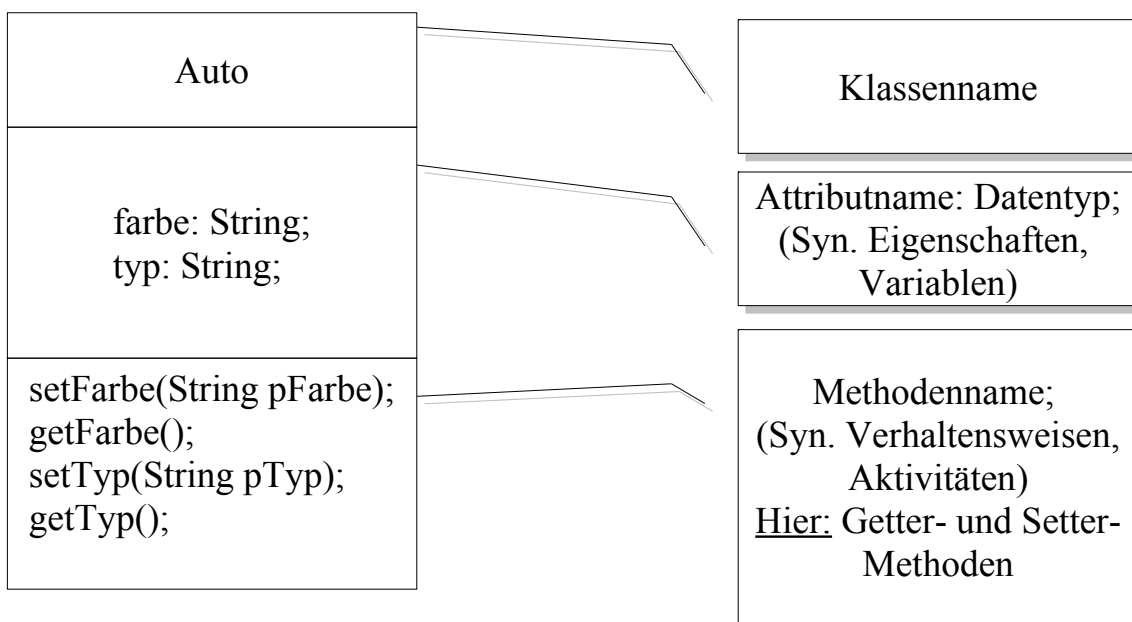
Anwendungsfälle:

<p><i>Bart Simpson</i> ist Schüler des <i>Wirtschaftsgymnasiums</i> der <i>KSW</i>. Er gibt die <i>Nachricht</i> als <i>Eingabe</i> „Hallo Franz ich bin im Computerraum“ ein und erhält die <i>Ausgabe</i> „Unyyb Senam vpu ova vz Pbzchgreenhz“.</p>	<p><i>Red Barklay</i> ist Schüler der <i>Berufsschule</i>. Er gibt als <i>Nachricht</i> „Hallo Moe endlich ist Wochenende“ ein und erhält das <i>Ausgabe</i> „Unyyb Zbr raqyvpu vfg Jbpuraraq“.</p>	<p><i>Moe Szyslak</i> ist Schüler des <i>Berufskollegs</i> der <i>KSW</i>. Er gibt die <i>Nachricht</i> „Unyyb Zbr raqyvpu vfg Jbpuraraq“ ein und erhält die <i>Ausgabe</i> „Hallo Moe endlich ist Wochenende“.</p>

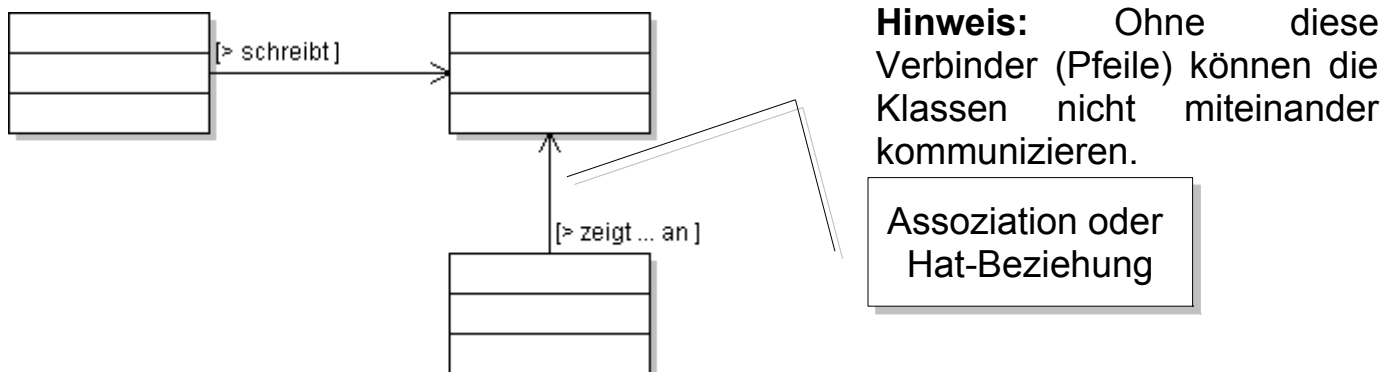


Merke: Sogenannte Getter- und Setter-Methoden¹ (Syn. Aktivität, Verhaltensweise) existieren für jede Eigenschaft. Z.B. Die Methoden `setFarbe(String pFarbe)` und `getFarbe()` für das Attribut (syn. Variable, Eigenschaft) `String farbe` der Klasse `Auto`. Diese Methoden dienen dazu Eigenschaftswerte einzelner Objekte zu modifizieren (bearbeiten, ändern) bzw. erstmalig zu initialisieren (Wert setzen). Es sind quasi Teilhandlungen auf unterster Ebene (Hinweis: kleinschrittig denken).

Zur Erinnerung die UML-Notation einer Klasse:



Vorgabe der System Architektur den Geheimnachrichten-Generator:



¹ „get“ steht für „holen“, „set“ steht für „setzen“



Schema für die Ver- bzw. Entschlüsselung (ROT13-Algorithmus):

Der ROT13-Algorithmus (auch Caesar-Chiffre oder Verschiebechiffre) ist ein Verfahren das symmetrisch verschlüsselt. Wenn jemand den Schlüssel kennt, kann er die Nachricht ver- und entschlüsseln.

%%Merke: Das Verschieben entspricht also dem ver- und entschlüsseln!
Überprüfen Sie diese Aussage!%%

ROT13-Algorithmus: public void verschieben(){...}

```
//Buchstaben verschieben
public void verschieben() {
    String sIn = this.getMessageInput();
    String sOut = new String();
    for (int i = 0; i < sIn.length(); i++) {
        char c = sIn.charAt(i);
        if (c >= 'a' && c <= 'm') c+=13;
        else if (c >= 'n' && c <= 'z') c-=13;
        else if (c >= 'A' && c <= 'M') c+=13;
        else if (c >= 'N' && c <= 'Z') c-=13;
        sOut = sOut + c;
    }
    this.setMessageOutput(sOut);
}
```

Linksverschiebung

Rechtsverschiebung

Erklärung:

Die eingegebene Nachricht wird in sIn vorübergehend gespeichert. Die auszugebende Ausgabe noch leer, aber das lokale Attribut sOut existiert bereits (ist deklariert und initialisiert).

For-Schleife: ist eine Kontrollstruktur die es u.a. erlaubt einen String (Zeichenkette) von Anfang (i = 0) bis hinten (i = 12) zu durchlaufen.

sIn.charAt()-Methode: Holt ein Zeichen an der Stelle *i* (Index) aus der Zeichenkette *sIn* raus und speichert es vorübergehend in einem Attribut *c* vom Typ *char* (char steht für character und enthält ein *einzelnes Zeichen*).



Stellen Sie sich das *Alphabet* so vor: Eine durchnummerierte Liste...

Großbuchstaben:

Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Eingabe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Kleinbuchstaben:

Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Eingabe	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

Beispiel 1: $(c \geq 'a' \ \&\& \ c \leq 'm') \ c+=13$

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Eingabe	a	b	c	d	e	f	g	h	i	j	k	l	m
Ausgabe	n	o	p	q	r	s	t	u	v	w	x	y	z

Man verschiebt im ersten Fall (IF-Zweig) einfach die Ausgabe um 13 Stellen im Alphabet nach rechts ($c+=13$).

Beispiel 2: $(c \geq 'n' \ \&\& \ c \leq 'z') \ c-=13$

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Eingabe	n	o	p	q	r	s	t	u	v	w	x	y	z
Ausgabe	a	b	c	d	e	f	g	h	i	j	k	l	m

Man verschiebt im zweiten Fall (erstes ELSEIF) einfach die Ausgabe um 13 Stellen im Alphabet nach links ($c-=13$).

Für Großbuchstaben gilt das gleiche!



Fach:
Informatik

Klasse:
WG12

Jahrgang:
Jahrgang 1

Autor:
Christine Janischek

Seite 5

Thema:
Systementwicklung einer Webapplikation - Geheimnachrichten-Generator

So könnte die grafische Benutzeroberfläche aussehen:

**Kaufmännische
Schule
Wangen**
Wege zeigen, öffnen,
gehen

Eingabe der Nachricht:
(Strings)

Ausgabe der Nachricht:
(Strings)

Geheime Nachricht erstellen (verschlüsseln)

Geheime Nachricht lesen (entschlüsseln)

Eingaben löschen



Vorlage: Basteln Sie zum besseren Verständnis die Cäsar-Scheibe

