

Objekte und Klassen

Ziel: Systeme entwerfen



Kaufmännische
Schule
Wangen

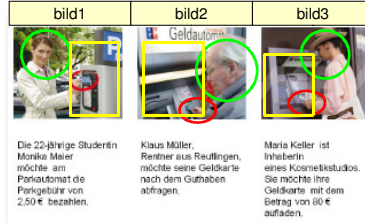
Wege zeigen, öffnen, gehen

Autor: Christine Janischek
Thema: Objekte und Klassen

Objekte der realen Welt

Hinweis: verwendete Bilder stammen aus Vorlagen des LS-BW und werden ausschließlich zu nicht-kommerziellen Bildungszwecken verwendet.

Problemstellung:
Verschiedene Personen bedienen unterschiedliche Automaten, mit dem Sie auf ihre Geldkarte zugreifen können.



Objekte sind *Sachen* oder *Dinge*. Sie **haben** *Eigenschaften* und **können** *Handlungen* **ausführen**.

Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen

Autor: Christine Janischek
Thema: Objekte und Klassen

Quelle: HR- Oberwies Christoph. Informatik an beruflichen Gymnasien - Jahrgangsstufe 1. Landesinstitut für Schulentwicklung, (01), 2008

(Technische) Systeme

Hinweis: verwendete Grafiken stammen aus dem Internet und werden ausschließlich zu nicht-kommerziellen Bildungszwecken verwendet.

In der Technik bezeichnet man alles, was eine geschlossene Funktion erfüllen kann, als technisches System. [...]

Wikimedia Foundation Inc.

Welche technischen Systeme nutzen Sie?



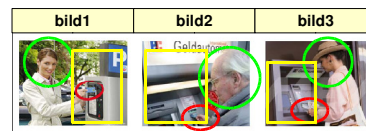
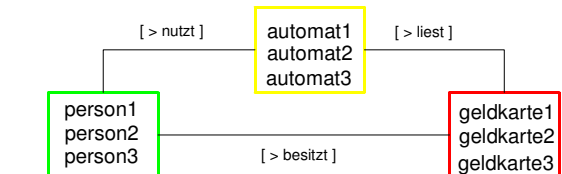
Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen

Autor: Christine Janischek
Thema: Objekte und Klassen

Objekte systematisieren

1. Schritt: Wir stecken zusammen was zusammen gehört!



Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen

Autor: Christine Janischek
Thema: Objekte und Klassen

Modell

Ein Modell ist ein **Abbild der Wirklichkeit**.

Ein Modell im Sinne der Modellbildung ist ein **abstraktes Abbild eines Systems**, welches stellvertretend für das System untersucht wird. [...]

Es geht um das **Erkennen**, **Beschreiben** und **Verstehen** komplexer Systeme [...]

Wikimedia Foundation Inc.

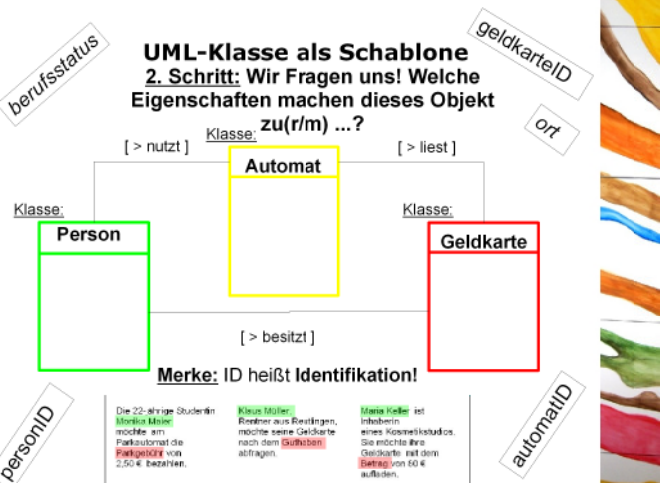
Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen

Autor: Christine Janischek
Thema: Objekte und Klassen

UML-Klasse als Schablone

2. Schritt: Wir Fragen uns! Welche Eigenschaften machen dieses Objekt zu(r/m) ... ?



Eigenschaften einer Klassen 3. Schritt: Zuordnung der Eigenschaften!

Person	Automat	Geldkarte
personID: <?>; nachname: <?>; vorname: <?>; berufsstatus: <?>; geschlecht: <?>;	automatID: <?>; ort: <?>;	geldkarteID: <?>; betrag: <?>; guthaben: <?>;

Merke: Eigenschaften heißen auch Attribute oder Variablen! Jede Eigenschaft hat einen **Datentyp!**

- String = Zeichenkette
- Integer = ganze Zahlen
- long = lange ganze Zahlen
- double = Kommazahlen
- float = lange Kommazahlen

Methoden einer Klasse 6. Schritt: Teilhandlungen zuordnen!

Person	Automat	Geldkarte
personID: String ; nachname: String ; vorname: String ; berufsstatus: String ; geschlecht: boolean ; karteEingeben(); personIDEingeben(); betragAuswaehlen();	automatID: String ; ort: String ; identitaetPruefen(); betragAuszahlen(); auswahlAnzeigen(); betragAnzeigen(); karteAusgeben();	geldkarteID: String ; betrag: double ; guthabenBerechnen();

Merke: Die Parameter-Attribute (pBetrag, pPersonID, pGeldkarteID) wurden aus Platzmangel weglassen!

Die 22-jährige Studentin Moritz Meier möchte am Parkautomat die Parkgebühr von 2,50 € bezahlen.

Klaus Müller, Rentner aus Reutlingen, möchte seine Geldkarte nach dem Guthaben abfragen.

Maria Keller ist Inhaberin eines Kosmetikstudios. Sie möchte ihre Geldkarte mit dem Betrag von 60 € auflösen.

Datentypen einer Eigenschaft 4. Schritt: Zuordnung der Datentypen je Eigenschaft!

Person	Automat	Geldkarte
personID: String ; nachname: String ; vorname: String ; berufsstatus: String ; geschlecht: boolean ;	automatID: String ; ort: String ;	geldkarteID: String ; betrag: double ; guthaben: double ;

Merke: Eigenschaften heißen auch Attribute oder Variablen! Jede Eigenschaft hat einen **Datentyp!**

- String = Zeichenkette
- Integer = ganze Zahlen
- long = lange ganze Zahlen
- double = Kommazahlen
- float = lange Kommazahlen

Alle Zeichen auf der Tastatur sind möglich!

Methoden: Teilhandlungen und Handlungen

```

Handlung: bezahlen() {
Teilhandlungen:
    person1.karteEingeben();
    person1.personIDEingeben();
    automat1.identitaetPruefen();
    automat1.auswahlAnzeigen();
    person1.betragBestaetigen();
    geldkarte1.guthabenBerechnen();
    automat1.betragAnzeigen();
    automat1.karteAusgeben();
}
    
```

Person1 möchte mit der Geldkarte bezahlen!

Merke: Die Parameter-Attribute (pBetrag, pPersonID, pGeldkarteID) wurden aus Platzmangel weglassen!

Die 22-jährige Studentin Moritz Meier möchte am Parkautomat die Parkgebühr von 2,50 € bezahlen.

Klaus Müller, Rentner aus Reutlingen, möchte seine Geldkarte nach dem Guthaben abfragen.

Maria Keller ist Inhaberin eines Kosmetikstudios. Sie möchte ihre Geldkarte mit dem Betrag von 60 € auflösen.

Methoden sind (Teil-)Handlungen einer Klasse 5. Schritt: Handlungen identifizieren!

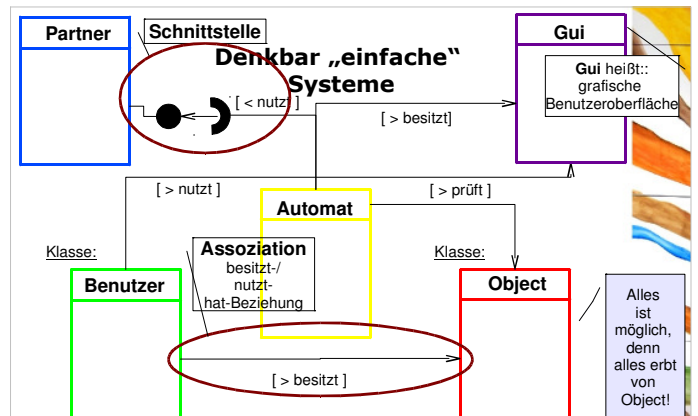
Person	Automat	Geldkarte
personID: String ; nachname: String ; vorname: String ; berufsstatus: String ; geschlecht: boolean ; karteEingeben(); karteAusgeben();	automatID: String ; ort: String ; identitaetPruefen(String pPersonID, String pGeldkarteID); betragAuszahlen(double pBetrag); betragAuswaehlen(double pBetrag); betragBestaetigen();	geldkarteID: String ; betrag: double ; guthabenBerechnen(); auswahlAnzeigen();

Merke: Klassenamen stehen stets im Singular und werden groß geschrieben. Eigenschaftsnamen (Attribute, Variablen) werden klein geschrieben. Methoden werden ebenfalls klein geschrieben.

Die 22-jährige Studentin Moritz Meier möchte am Parkautomat die Parkgebühr von 2,50 € bezahlen.

Klaus Müller, Rentner aus Reutlingen, möchte seine Geldkarte nach dem Guthaben abfragen.

Maria Keller ist Inhaberin eines Kosmetikstudios. Sie möchte ihre Geldkarte mit dem Betrag von 60 € auflösen.



Merke: Klassenamen stehen stets im Singular und werden groß geschrieben. Eigenschaftsnamen (Attribute, Variablen) werden klein geschrieben. Methoden werden ebenfalls klein geschrieben.

Alles ist möglich, denn alles erbt von Object!

