«interface» reiseanbieter::Reiseanbieter

EGAL = 0: int BUS = 1: int BAHN = 2: int

entfernungBus = 288.2: double entfernungBahn = 310: double

buchen(Kunde k, int transport): boolean buchen(Kunde[] gruppe, int transport): boolean buchungBestaetigen(Kunde k, int transport): void buchungAblehnen(Kunde k, int transport): void stornieren(Kunde k, int transport): boolean stornieren(Kunde[] gruppe, int transport): boolean stornoBestaetigen(Kunde k, int transport): void stornoAblehnen(Kunde k, int transport): void preisErfragen(Kunde k, int transport): double dauerErfragen(Kunde k, int transport): double



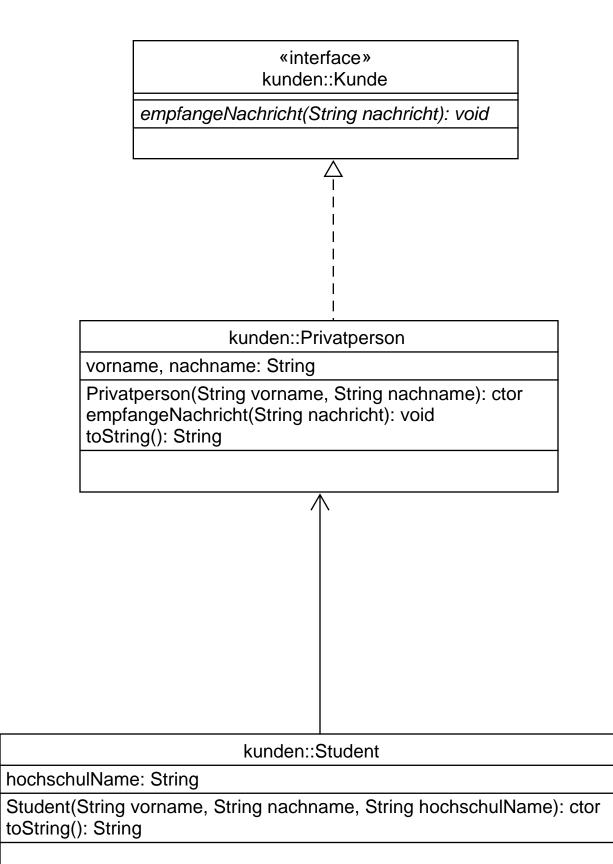
BUSKOSTEN = 25, BAHNKOSTEN = 40: double

bahn: Bahn bus: Bus

STRECKEBAHN = 310, STRECKEBUS = 288.8: double

bahnBuchung: Kunde[] busBuchung: Kunde[]

TuiReisen(Bus bus, Bahn bahn): constructor buchen(Kunde k, int transport): boolean buchen(Kunde[] gruppe, int transport): boolean buchungBestaetigen(Kunde k, int transport): void buchungAblehnen(Kunde k, int transport): void stornieren(Kunde k, int transport): boolean stornieren(Kunde[] gruppe, int transport): boolean stornoBestaetigen(Kunde k, int transport): void stornoAblehnen(Kunde k, int transport): void preisErfragen(Kunde k, int transport): double dauerErfragen(Kunde k, int transport): double



«interface» transportmittel::Transportmittel

getGeschwindigkeit(): double getAnzahlPlaetze(): int

transportmittel::Bus transportmittel::Bahn geschwindigkeit: double anzahlPlaetze: int

geschwindigkeit: double anzahlPlaetze: int

Bus(double geschwindigkeit, int anzahlPlaetze): ctor

getGeschwindigkeit(): double getAnzahlPlaetze(): int

getGeschwindigkeit(): double getAnzahlPlaetze(): int

Bahn(double geschwindigkeit, int anzahlPlaetze): ctor