

Klasse	Methode	Input	Output	Fehlermeldungen	Erfolgsmeldung	Kommentar
NutzerHandler	checkPasswort	String anmeldename, String passwort	String	- "Nutzer nicht gefunden." - "Falsches Passwort."	"Login erfolgreich".	Methode, die die Richtigkeit der Nutzerdaten gewährleistet. Datenbankabfrage auf Nutzernamen und Passwort.
	createNutzer	String vorname, String nachname, String anmeldename, String passwort, String nutzertyp	String	- "Nutzer schon vorhanden." - "Nutzerdaten unvollständig"	"Erfolgreiche Registrierung!"	Methode, die einen Nutzer in der Datenbank anlegt und somit registriert. Das Passwort wird an die Methode als Klartext übertragen.
	getNutzertyp	String anmeldename	String	-	-	Methode, die zu einem Anmeldennamen den Nutzertypen ausgibt. Es sollte vorher die Methode checkPasswort ausgeführt werden.
RelationHandler	createRelation	String startort, String zielort, String flugzeit, int distanz	String	- "Relation schon vorhanden." - "Startort nicht gefunden." - "Zielort nicht gefunden."	"Erfolgreiche Anlage der Relation!"	Methode, die eine Relation anlegt. <b>String flugzeit im Format: "hh:mm:ss". Startort und Zielort werden als Flughafennamen übergeben, aber mit der FlughafenID gespeichert!</b>
	getAllFlughafennamen	-	List<String>	-	-	Methode, die alle Flughafennamen ausgibt.
FlugHandler	createFlug	String abflug, String relationString, double preis	String	"Die Relation ist nicht vorhanden"	"Erfolgreiche Anlage des Fluges"	Methode, die eine Flug anlegt. Anfangs werden die Felder flugzeitid und mahlzeitid auf die default Felder der Tabellen Flugzeug und Mahlzeit belegt (jeweils die ID+1). Die FlugID setzt sich nach dem folgenden Muster zusammen: MH %relationid% / %flugnummer% (Die Flugnummer wird bei Neuanlage um 1 erhöht). Folglich hat der erste Flug der Relation 1 die FlugID MH1/1 der zweite Flug MH1/2 usw. Der String Abflug muss im Format <b>EEE MMM dd HH:mm:ss zzz yyyy</b> (z.B. Tue Apr 17 17:46:00 CEST 2018) sein.
	getAllRelationen	-	List<String>	-	-	gibt alle Relationen aus. Beispiel: "1. Relation: Startort: FRA, Zielort: BOM (1500 km, 10:30:00 Stunden)"
	getAllFluege	-	List<String>	-	-	gibt alle Flüge aus. Beispiel: "MH1/1: Abflug: 2018-26-01 01:26, Ankunft: 2018-26-01 01:26 (Preis: 25.00 €)"
FlugzeugHandler	getFlugstatus	String aktuellString, String flugstring	String	"Der Status kann nicht gesetzt werden."	-	Ausgabe des Status eines Flugs zu einer gegebenen Uhrzeit
	createFlugzeug	String hersteller, String typ, int sitzplaetze	String	"Bitte geben Sie alle Information an!"	"Erfolgreiche Anlage des Flugzeugs!"	Methode, die ein Flugzeug anlegt.
	getAlleFlugzeuge	-	List<String>	-	-	gibt alle Flugzeuge aus. Beispiel: "1. Flugzeug: Airbus A380-800 (853 Sitzplätze)" der Default Wert mit der flugzeitid=1 wird nicht ausgegeben. Der erste angezeigt Eintrag ist der zweite in der Datenbank. (Es wird auch mit 1. Flugzeug begonnen)
MahlzeitHandler	assignFlugzeugToFug	String flugzeugString, String flugString	String	- "Dieses Flugzeug wurde bereits einem Flug zugeordnet." - "Dieser Flugzeug wurde bereits diesem Flug zugeordnet."	"Erfolgreiche Zuordnung des Flugzeugs!"	Methode, die ein Flugzeug einem Flug zuordnet. Zu beachten ist, dass ein Flugzeug nur genau einem Flug zugeordnet werden kann!
	createMahlzeit	String name, String art, Boolean vegetarisch	String	- "Bitte geben Sie alle Information an!" - "Mahlzeit schon vorhanden."	"Erfolgreiche Anlage der Mahlzeit!"	Methode, die eine Mahlzeit anlegt.
	getAllMahlzeiten	-	List<String>	-	-	gibt alle Mahlzeiten aus. Beispiel: "1. Mahlzeit: Pizza Margarita (Teigwaren, vegetarisch: ja)" der Default Wert mit der mahlzeitid=1 wird nicht ausgegeben. Der erste angezeigt Eintrag ist der zweite in der Datenbank. (Es wird auch mit 1. Mahlzeit begonnen)
PassagierHandler	assignMahlzeitToFug	String mahlzeitString, String flugString	String	-	"Erfolgreiche Zuordnung der Mahlzeit!"	Methode, die eine Mahlzeit einem Flug zuordnet.
	createPassagier	String vorname, String nachname, String anschrift, String geburtsdatum, String nationaltaet	String	"Bitte geben Sie alle Information an!"	"Erfolgreiche Anlage des Passagiers"	Methode, die einen Passagier anlegt.
	getAllPassagiere	-	List<String>	-	-	gibt alle Passagier aus. Beispiel: "1. Passagier: Halli Özdoğan (Anschrift: Am Stockhof 2, 31785 Hameln, Geburtsdatum: 08.09.1995, Nationaltaet: deutsch)"
BuchungHandler	createBuchung	String passagierString, String flugString	String	"Bitte geben Sie alle Information an!"	"Erfolgreiche Anlage der Buchung!"	Methode, die eine Buchung anlegt. Das Feld Buchungsdatum wird auf den Zeitpunkt der Buchung gesetzt.
	getBuchungenbyFlug	String flugString	List<String>	-	-	Methode, die alle Buchungen zu einem Flug ausgibt. Beispiel: 1. Buchung am Mon Apr 02 00:47:35 CEST 2018, Flug: MH1/4, Passagier: Halli Özdoğan
	getBuchungenbyPassagier	String passagierString	List<String>	-	-	Methode, die alle Buchungen zu einem Passagier ausgibt. Beispiel: 1. Buchung am Mon Apr 02 00:47:35 CEST 2018, Flug: MH1/4, Passagier: Halli Özdoğan
ModellHandler	calculateCapacity	String flugString	String	Dieser Flug hat noch kein zugeordnetes Flugzeug!	-	gibt die Kapazität eines Flugs aus. Beispiel: 842 von 853 Plätzen noch frei.
	getAlleFlugModels	-	List<FlugModel>	-	-	Gibt eine Liste von Typ FlugModel zurück
ModellHandler	getDepartureSchedulesModels	String time	List<DepartureSchedulesModel>	-	-	Gibt eine Liste von Typ DepartureSchedulesModel zurück