

# RailInsights API

## API and SDK Documentation

Version: v1.0

API for RailInsights

# StationController

## findStationsByName

GET

/api/stations

### Usage and SDK Samples

Curl    Java    Android    Obj-C    JavaScript    C#    PHP    Perl    Python

```
curl -X GET\
-H "Accept: */*" \
"https://railinsights.de/api/stations?name="
```

### Parameters

Query parameters

Name	Description
name*	String Required

### Responses

Status: 200 - OK

Schema

▼ [ {}

▼ { {}

address: ▼ { {}

city: string

zipcode: string

street: string

}

location: ▼ { {}

latitude: number (float)

longitude: number (float)

}

operator: ▼ { {}

type: string

operatorId: string

name: string

}

timeTableOffice: ▼ { {}

email: string

name: string

```

    }
    regionalbereich:
      ▼ { []
        number:    integer (int32)
        name:      string
        shortName: string
      }
    stationManagement:
      ▼ { []
        number: integer (int32)
        name:   string
      }
    szentrale:
      ▼ { []
        number:          integer (int32)
        publicPhoneNumber: string
        name:            string
      }
    stationId:      integer (int64)
    relevance:      number (float)
    score:          number (float)
    weight:         number (float)
    type:           string
    ril100:         string
    nr:             number (float)
    name:           string
    category:       number (float)
    priceCategory:  number (float)
    hasParking:     boolean
    hasBicycleParking: boolean
    hasLocalPublicTransport: boolean
    hasPublicFacilities: boolean
    hasLockerSystem: boolean
    hasTaxiRank:    boolean
    hasTravelNecessities: boolean
    hasSteplessAccess: string
    hasMobilityService: string
    hasWiFi:        boolean
    hasTravelCenter: boolean
    hasRailwayMission: boolean
    hasDBLounge:    boolean
    hasLostAndFound: boolean
    hasCarRental:    boolean
    federalState:    string
    createdAt:       string (date-time)
    ril100Identifiers:
      ▼ [ []
        ▼ { []
          rilIdentifier:    string
          hasSteamPermission: boolean
          geographicCoordinates: ▼ { []
            type:    string
            latitude: number (float)
            longitude: number (float)
          }
          main:    boolean
        }
      ]
    productLine:
      ▼ { []
        productLine: string
        segment:     string

```

```
    }
  }
}
```

# getStationById

GET

/api/stations/{stationId}

## Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X GET\
-H "Accept: */*" \
"https://railinsights.de/api/stations/{stationId}"
```

## Parameters

Path parameters

Name	Description
stationId*	Long (int64) Required

## Responses

Status: 200 - OK

Schema

▼ { []

address:

▼ { []

city: string

zipcode: string

street: string

}

location:

▼ { []

latitude: number (float)

longitude: number (float)

}

operator:

▼ { []

type: string

operatorId: string

name: string

}

timeTableOffice:

▼ { []

email: string

name: string

}

regionalbereich:

▼ { []

number: integer (int32)

name: string

shortName: string

```

    }
stationManagement: ▼ { []
    number: integer (int32)
    name: string
}

szentrale: ▼ { []
    number: integer (int32)
    publicPhoneNumber: string
    name: string
}

stationId: integer (int64)
relevance: number (float)
score: number (float)
weight: number (float)
type: string
ril100: string
nr: number (float)
name: string
category: number (float)
priceCategory: number (float)
hasParking: boolean
hasBicycleParking: boolean
hasLocalPublicTransport: boolean
hasPublicFacilities: boolean
hasLockerSystem: boolean
hasTaxiRank: boolean
hasTravelNecessities: boolean
hasSteplessAccess: string
hasMobilityService: string
hasWiFi: boolean
hasTravelCenter: boolean
hasRailwayMission: boolean
hasDBLounge: boolean
hasLostAndFound: boolean
hasCarRental: boolean
federalState: string
createdAt: string (date-time)
ril100Identifiers: ▼ [ []
    ▼ { []
        rilIdentifier: string
        hasSteamPermission: boolean
        geographicCoordinates: ▼ { []
            type: string
            latitude: number (float)
            longitude: number (float)
        }
        main: boolean
    }
]

productLine: ▼ { []
    productLine: string
    segment: string
}
}

```

# StopController

## findStopsByName

GET

/api/stops

### Usage and SDK Samples

Curl    Java    Android    Obj-C    JavaScript    C#    PHP    Perl    Python

```
curl -X GET\
-H "Accept: */*" \
"https://railinsights.de/api/stops?name="
```

### Parameters

Query parameters

Name	Description
name*	String Required

### Responses

Status: 200 - OK

Schema

▼ [ {}

▼ { {}

type: string

stopId: integer (int64)

name: string

location: ▼ { {}

latitude: number (double)

longitude: number (double)

}

products: ▼ { {}

nationalExpress: boolean

national: boolean

regionalExp: boolean

regional: boolean

suburban: boolean

bus: boolean

ferry: boolean

subway: boolean

tram: boolean

taxi: boolean

}

createdAt: string (date-time)

station: ▼ { {}

address: ▼ { {}

city: string

```

        zipcode: string
        street: string
    }

location: ▼ { []
    latitude: number (float)
    longitude: number (float)
}

operator: ▼ { []
    type: string
    operatorId: string
    name: string
}

timeTableOffice: ▼ { []
    email: string
    name: string
}

regionalbereich: ▼ { []
    number: integer (int32)
    name: string
    shortName: string
}

stationManagement: ▼ { []
    number: integer (int32)
    name: string
}

szentrale: ▼ { []
    number: integer (int32)
    publicPhoneNumber: string
    name: string
}

stationId: integer (int64)
relevance: number (float)
score: number (float)
weight: number (float)
type: string
ril100: string
nr: number (float)
name: string
category: number (float)
priceCategory: number (float)
hasParking: boolean
hasBicycleParking: boolean
hasLocalPublicTransport: boolean
hasPublicFacilities: boolean
hasLockerSystem: boolean
hasTaxiRank: boolean
hasTravelNecessities: boolean
hasSteplessAccess: string
hasMobilityService: string
hasWiFi: boolean
hasTravelCenter: boolean
hasRailwayMission: boolean
hasDBLounge: boolean
hasLostAndFound: boolean
hasCarRental: boolean
federalState: string

```

createdAt:

string (date-time)

ril100Identifiers:

▼ [ {} ]

▼ { {} }

rilIdentifier:

string

hasSteamPermission:

boolean

geographicCoordinates:

▼ { {} }

type:

string

latitude:

number (float)

longitude:

number (float)

}

main:

boolean

}

}

productLine:

▼ { {} }

productLine:

string

segment:

string

}

}

}

## getStopById

GET

/api/stops/{stopId}

### Usage and SDK Samples

Curl

Java

Android

Obj-C

JavaScript

C#

PHP

Perl

Python

```
curl -X GET\  
-H "Accept: */*"\  
"https://railinsights.de/api/stops/{stopId}"
```

### Parameters

Path parameters

Name	Description
stopId*	Long (int64) Required

### Responses

Status: 200 - OK

Schema

▼ { {} }

type:

string

stopId:

integer (int64)

name:

string

location:

▼ { {} }

latitude:

number (double)

longitude:

number (double)

```

    }
products: ▼ { []
    nationalExpress: boolean
    national:         boolean
    regionalExp:      boolean
    regional:         boolean
    suburban:         boolean
    bus:              boolean
    ferry:            boolean
    subway:          boolean
    tram:             boolean
    taxi:             boolean
}
createdAt: string (date-time)
station: ▼ { []
    address: ▼ { []
        city:    string
        zipcode: string
        street:  string
    }
    location: ▼ { []
        latitude: number
        longitude: number
    }
    operator: ▼ { []
        type:      string
        operatorId: string
        name:      string
    }
    timeTableOffice: ▼ { []
        email: string
        name:  string
    }
    regionalbereich: ▼ { []
        number:    integer
        name:      string
        shortName: string
    }
    stationManagement: ▼ { []
        number: integer
        name:   string
    }
    szentrale: ▼ { []
        number:            integer
        publicPhoneNumber: string
        name:              string
    }
    stationId: integer (int64)
    relevance: number (float)
    score:     number (float)
    weight:    number (float)
    type:      string
    ril100:    string
    nr:        number (float)
    name:      string
    category:  number (float)
    priceCategory: number (float)

```



hasParking:	boolean
hasBicycleParking:	boolean
hasLocalPublicTransport:	boolean
hasPublicFacilities:	boolean
hasLockerSystem:	boolean
hasTaxiRank:	boolean
hasTravelNecessities:	boolean
hasSteplessAccess:	string
hasMobilityService:	string
hasWiFi:	boolean
hasTravelCenter:	boolean
hasRailwayMission:	boolean
hasDBLounge:	boolean
hasLostAndFound:	boolean
hasCarRental:	boolean
federalState:	string
createdAt:	string (date-time)
ril100Identifiers:	<div>▼ [ {} ]<div>▼ { {} }<div>rilIdentifier: string</div><div>hasSteamPermission: boolean</div><div>geographicCoordinates: ▼ { {} }<div>type: string</div><div>latitude: number (float)</div><div>longitude: number (float)</div></div></div><div>main: boolean</div></div>

# TripController

## findTripsByLineProductName

GET

/api/trips/line-productName

### Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X GET\  
-H "Accept: */*"\  
"https://railinsights.de/api/trips/line-productName?lineName=&page=&size="
```

## Parameters

## Query parameters

Name	Description
lineName*	String Required
page	Integer (int32)
size	Integer (int32)

## Responses

**Status: 200 - OK**

## Schema

```

▼ { []
  totalPages: integer (int32)
  totalElements: integer (int64)
  pageable: ▼ { []
    pageNumber: integer (int32)
    pageSize: integer (int32)
    paged: boolean
    unpaged: boolean
    offset: integer (int64)
    sort: ▼ { []
      sorted: boolean
      unsorted: boolean
      empty: boolean
    }
  }
  first: boolean
  last: boolean
  size: integer (int32)
  content: ▼ [ []
    ▼ [ []
      ▼ { []
        tripId: string
        stop: ▼ { []
          type: string
          stopId: integer (int64)
          name: string
          location: ▼ { []
            latitude: number (double)
            longitude: number (double)
          }
          products: ▼ { []
            nationalExpress: boolean
            national: boolean
            regionalExp: boolean
            regional: boolean
            suburban: boolean
            bus: boolean
            ferry: boolean
            subway: boolean
          }
        }
      }
    ]
  ]
}

```

```
    tram:          boolean
    taxi:          boolean
  }
createdAt: string (date-time)
station: ▼ { []
  address: ▼ { []
    city:    string
    zipcode: string
    street:  string
  }
  location: ▼ { []
    latitude: number (float)
    longitude: number (float)
  }
  operator: ▼ { []
    type:      string
    operatorId: string
    name:      string
  }
  timeTableOffice: ▼ { []
    email: string
    name: string
  }
  regionalbereich: ▼ { []
    number:    integer (int32)
    name:      string
    shortName: string
  }
  stationManagement: ▼ { []
    number: integer (int32)
    name:   string
  }
  szentrale: ▼ { []
    number:          integer (int32)
    publicPhoneNumber: string
    name:            string
  }
  stationId: integer (int64)
  relevance: number (float)
  score:     number (float)
  weight:    number (float)
  type:      string
  ril100:    string
  nr:        number (float)
  name:      string
  category:  number (float)
  priceCategory: number (float)
  hasParking: boolean
  hasBicycleParking: boolean
  hasLocalPublicTransport: boolean
  hasPublicFacilities: boolean
  hasLockerSystem: boolean
  hasTaxiRank: boolean
  hasTravelNecessities: boolean
  hasSteplessAccess: string
  hasMobilityService: string
  hasWiFi: boolean
```

```
hasTravelCenter:      boolean
hasRailwayMission:    boolean
hasDBLounge:          boolean
hasLostAndFound:      boolean
hasCarRental:         boolean
federalState:         string
createdAt:            string (date-time)
ril100Identifiers:    ▼ [ []
  ▼ { []
    rilIdentifier:      string
    hasSteamPermission: boolean
    geographicCoordinates: ▼ { []
      type:             string
      latitude:          float
      longitude:         float
    }
    main:               boolean
  }
]
productLine:          ▼ { []
  productLine: string
  segment:      string
}
```

```

}
when:      string (date-time)
plannedWhen: string (date-time)
prognosedWhen: string (date-time)
delay:      integer (int32)
platform:   string
plannedPlatform: string
prognosedPlatform: string
prognosisType: string
direction:  string
provenance: string
line:      ▼ { []
  operator: ▼ { []
    type:      string
    operatorId: string
    name:      string
  }
  type:      string
  fahrtNr:   string
  name:      string
  linePublic: boolean
  productName: string
  lineId:    string
  product:   string
  mode:      string
  Enum:      BUS , TAXI , WATERCRAFT , TRAIN
  createdAt: string (date-time)
}
origin: ▼ { []
  type:      string
  stopId:    integer (int64)
```

```
name:      string
location:  ▼ { []
            latitude:  number (double)
            longitude: number (double)
          }
products:  ▼ { []
            nationalExpress: boolean
            national:        boolean
            regionalExp:     boolean
            regional:        boolean
            suburban:        boolean
            bus:              boolean
            ferry:           boolean
            subway:          boolean
            tram:            boolean
            taxi:            boolean
          }
createdAt: string (date-time)
station:   ▼ { []
            address:
              ▼ { []
                city:      string
                zipcode: string
                street:   string
              }
            location:
              ▼ { []
                latitude:  number (float)
                longitude: number (float)
              }
            operator:
              ▼ { []
                type:      string
                operatorId: string
                name:      string
              }
            timeTableOffice:
              ▼ { []
                email: string
                name: string
              }
            regionalbereich:
              ▼ { []
                number:  integer (int32)
                name:    string
                shortName: string
              }
            stationManagement:
              ▼ { []
                number: integer (int32)
                name:   string
              }
            szentrale:
              ▼ { []
                number:          integer (int32)
                publicPhoneNumber: string
                name:            string
              }
            stationId: integer (int64)
            relevance: number (float)
            score:     number (float)
            weight:    number (float)
            type:      string
            ril100:    string
```

	nr:	number (float)
	name:	string
	category:	number (float)
	priceCategory:	number (float)
	hasParking:	boolean
	hasBicycleParking:	boolean
	hasLocalPublicTransport:	boolean
	hasPublicFacilities:	boolean
	hasLockerSystem:	boolean
	hasTaxiRank:	boolean
	hasTravelNecessities:	boolean
	hasSteplessAccess:	string
	hasMobilityService:	string
	hasWiFi:	boolean
	hasTravelCenter:	boolean
	hasRailwayMission:	boolean
	hasDBLounge:	boolean
	hasLostAndFound:	boolean
	hasCarRental:	boolean
	federalState:	string
	createdAt:	string (date-time)
	ril100Identifiers:	<div>▼ [ {} ]</div> <div> <div>▼ { {} }</div> <div> rilIdentifier: string </div> <div> hasSteamPermission: boolean </div> <div> geographicCoordinates: ▼ { {} } <div> type: string </div> <div> latitude: number (float) </div> <div> longitude: number (float) </div> </div> </div>

main: boolean

```
ferry:          boolean
subway:         boolean
tram:          boolean
taxi:          boolean
}

createdAt: string (date-time)
station: ▼ { []
  address: ▼ { []
    city:      string
    zipcode: string
    street:   string
  }

  location: ▼ { []
    latitude:  number (float)
    longitude: number (float)
  }

  operator: ▼ { []
    type:      string
    operatorId: string
    name:      string
  }

  timeTableOffice: ▼ { []
    email: string
    name: string
  }

  regionalbereich: ▼ { []
    number:  integer (int32)
    name:    string
    shortName: string
  }

  stationManagement: ▼ { []
    number: integer (int32)
    name:   string
  }

  szentrale: ▼ { []
    number: integer (int32)
    publicPhoneNumber: string
    name: string
  }

  stationId: integer (int64)
  relevance: number (float)
  score:     number (float)
  weight:    number (float)
  type:      string
  ril100:    string
  nr:        number (float)
  name:      string
  category:  number (float)
  priceCategory: number (float)
  hasParking: boolean
  hasBicycleParking: boolean
  hasLocalPublicTransport: boolean
  hasPublicFacilities: boolean
  hasLockerSystem: boolean
  hasTaxiRank: boolean
  hasTravelNecessities: boolean
  hasSteplessAccess: string
```





# findTripsByTripIdAndStop

GET

/api/trips

## Usage and SDK Samples

Curl    Java    Android    Obj-C    JavaScript    C#    PHP    Perl    Python

```
curl -X GET\
-H "Accept: */*" \
"https://railinsights.de/api/trips?tripId="
```

## Parameters

Query parameters

Name	Description
tripId*	String Required

## Responses

Status: 200 - OK

Schema

▼ [ {}

▼ { {}

tripId: string

stop: ▼ { {}

type: string

stopId: integer (int64)

name: string

location: ▼ { {}

latitude: number (double)

longitude: number (double)

}

products: ▼ { {}

nationalExpress: boolean

national: boolean

regionalExp: boolean

regional: boolean

suburban: boolean

bus: boolean

ferry: boolean

subway: boolean

tram: boolean

taxi: boolean

}

createdAt: string (date-time)

station: ▼ { {}

address: ▼ { {}

city: string

zipcode: string

street: string

}

location: ▼ { {}

```

latitude:  number (float)
longitude:  number (float)
}

operator:  ▼ { []
            type:      string
            operatorId: string
            name:       string
        }

timeTableOffice: ▼ { []
                  email: string
                  name: string
                }

regionalbereich: ▼ { []
                  number:  integer (int32)
                  name:    string
                  shortName: string
                }

stationManagement: ▼ { []
                   number: integer (int32)
                   name:   string
                 }

szentrale: ▼ { []
            number:      integer (int32)
            publicPhoneNumber: string
            name:        string
          }

stationId: integer (int64)
relevance: number (float)
score:     number (float)
weight:    number (float)
type:      string
ril100:    string
nr:        number (float)
name:      string
category:  number (float)
priceCategory: number (float)
hasParking: boolean
hasBicycleParking: boolean
hasLocalPublicTransport: boolean
hasPublicFacilities: boolean
hasLockerSystem: boolean
hasTaxiRank: boolean
hasTravelNecessities: boolean
hasSteplessAccess: string
hasMobilityService: string
hasWiFi: boolean
hasTravelCenter: boolean
hasRailwayMission: boolean
hasDBLounge: boolean
hasLostAndFound: boolean
hasCarRental: boolean
federalState: string
createdAt: string (date-time)
ril100Identifiers: ▼ [ []
                   ▼ { []
                       rilIdentifier: string

```

```

        hasSteamPermission: boolean
        geographicCoordinates: ▼ { []
            type: string
            latitude: number (float)
            longitude: number (float)
        }
        main: boolean
    }
}

productLine: ▼ { []
    productLine: string
    segment: string
}

}

when: string (date-time)
plannedWhen: string (date-time)
prognosedWhen: string (date-time)
delay: integer (int32)
platform: string
plannedPlatform: string
prognosedPlatform: string
prognosisType: string
direction: string
provenance: string
line: ▼ { []
    operator: ▼ { []
        type: string
        operatorId: string
        name: string
    }
    type: string
    fahrtNr: string
    name: string
    linePublic: boolean
    productName: string
    lineId: string
    product: string
    mode: string
    Enum: BUS, TAXI, WATERCRAFT, TRAIN
    createdAt: string (date-time)
}

origin: ▼ { []
    type: string
    stopId: integer (int64)
    name: string
    location: ▼ { []
        latitude: number (double)
        longitude: number (double)
    }
    products: ▼ { []
        nationalExpress: boolean
        national: boolean
        regionalExp: boolean
        regional: boolean
    }
}

```

suburban: boolean  
bus: boolean  
ferry: boolean  
subway: boolean  
tram: boolean  
taxi: boolean

}

createdAt: string (date-time)

station: ▼ { []

address:

▼ { []

city: string

zipcode: string

street: string

}

location:

▼ { []

latitude: number (float)

longitude: number (float)

}

operator:

▼ { []

type: string

operatorId: string

name: string

}

timeTableOffice:

▼ { []

email: string

name: string

}

regionalbereich:

▼ { []

number: integer (int32)

name: string

shortName: string

}

stationManagement:

▼ { []

number: integer (int32)

name: string

}

szentrale:

▼ { []

number: integer (int32)

publicPhoneNumber: string

name: string

}

stationId: integer (int64)

relevance: number (float)

score: number (float)

weight: number (float)

type: string

ril100: string

nr: number (float)

name: string

category: number (float)

priceCategory: number (float)

hasParking: boolean

hasBicycleParking: boolean

hasLocalPublicTransport: boolean

hasPublicFacilities: boolean

hasLockerSystem: boolean

hasTaxiRank: boolean

```

        hasTravelNecessities: boolean
        hasSteplessAccess: string
        hasMobilityService: string
        hasWiFi: boolean
        hasTravelCenter: boolean
        hasRailwayMission: boolean
        hasDBLounge: boolean
        hasLostAndFound: boolean
        hasCarRental: boolean
        federalState: string
        createdAt: string (date-time)
        ril100Identifiers: ▼ [ []
            ▼ { []
                rilIdentifier: string
                hasSteamPermission: boolean
                geographicCoordinates: ▼ { []
                    type: string
                    latitude: number (float)
                    longitude: number (float)
                }
                main: boolean
            }
        ]
        productLine: ▼ { []
            productLine: string
            segment: string
        }
    }
}

destination: ▼ { []
    type: string
    stopId: integer (int64)
    name: string
    location: ▼ { []
        latitude: number (double)
        longitude: number (double)
    }
    products: ▼ { []
        nationalExpress: boolean
        national: boolean
        regionalExp: boolean
        regional: boolean
        suburban: boolean
        bus: boolean
        ferry: boolean
        subway: boolean
        tram: boolean
        taxi: boolean
    }
    createdAt: string (date-time)
    station: ▼ { []
        address: ▼ { []
            city: string
            zipcode: string
            street: string

```

```

    }
location:
  ▼ { []
    latitude: number (float)
    longitude: number (float)
  }
operator:
  ▼ { []
    type: string
    operatorId: string
    name: string
  }
timeTableOffice:
  ▼ { []
    email: string
    name: string
  }
regionalbereich:
  ▼ { []
    number: integer (int32)
    name: string
    shortName: string
  }
stationManagement:
  ▼ { []
    number: integer (int32)
    name: string
  }
szentrale:
  ▼ { []
    number: integer (int32)
    publicPhoneNumber: string
    name: string
  }
stationId: integer (int64)
relevance: number (float)
score: number (float)
weight: number (float)
type: string
ril100: string
nr: number (float)
name: string
category: number (float)
priceCategory: number (float)
hasParking: boolean
hasBicycleParking: boolean
hasLocalPublicTransport: boolean
hasPublicFacilities: boolean
hasLockerSystem: boolean
hasTaxiRank: boolean
hasTravelNecessities: boolean
hasSteplessAccess: string
hasMobilityService: string
hasWiFi: boolean
hasTravelCenter: boolean
hasRailwayMission: boolean
hasDBLounge: boolean
hasLostAndFound: boolean
hasCarRental: boolean
federalState: string
createdAt: string (date-time)
ril100Identifiers: ▼ [ []

```

```
▼ { []
  rilIdentifier:      string
  hasSteamPermission: boolean
  geographicCoordinates: ▼ { []
                          type:      string
                          latitude:  number
                                   (float)
                          longitude: number
                                   (float)
                          }
  main:              boolean
}

{ []
  productLine: string
  segment:     string
```

```

    cancelled: boolean
    loadFactor: string
    remarks: ▼ [ []
      ▼ { []
        type: string
        code: string
        text: string
        createdAt: string (date-time)
      }
    ]
  createdAt: string (date-time)
}
]

```

## findTripsWhenAfter

**GET**

```
/api/trips/stopId/when
```

## Usage and SDK Samples

Curl	Java	Android	Obj-C	JavaScript	C#	PHP	Perl	Python
------	------	---------	-------	------------	----	-----	------	--------

```
curl -X GET\  
-H "Accept: */*"  
"https://railinsights.de/api/trips/stopId/when?stopId=&when="
```

## Parameters

## Query parameters

Name	Description
stopId*	Long (int64) Required
when*	Date (date-time) Required

Responses

Status: 200 - OK

Schema

▼ [ {}

▼ { {}

tripId: string

stop: ▼ { {}

type: string

stopId: integer (int64)

name: string

location: ▼ { {}

latitude: number (double)

longitude: number (double)

}

products: ▼ { {}

nationalExpress: boolean

national: boolean

regionalExp: boolean

regional: boolean

suburban: boolean

bus: boolean

ferry: boolean

subway: boolean

tram: boolean

taxi: boolean

}

createdAt: string (date-time)

station: ▼ { {}

address: ▼ { {}

city: string

zipcode: string

street: string

}

location: ▼ { {}

latitude: number (float)

longitude: number (float)

}

operator: ▼ { {}

type: string

operatorId: string

name: string

}

timeTableOffice: ▼ { {}

email: string

name: string

}

regionalbereich: ▼ { {}

number: integer (int32)

name: string

shortName: string

}

stationManagement: ▼ { {}

number: integer (int32)

name: string

}



	szentrale:	<div><div>▼ { []</div><div>number: integer (int32)</div><div>publicPhoneNumber: string</div><div>name: string</div><div>}</div></div>
	stationId:	integer (int64)
	relevance:	number (float)
	score:	number (float)
	weight:	number (float)
	type:	string
	ril100:	string
	nr:	number (float)
	name:	string
	category:	number (float)
	priceCategory:	number (float)
	hasParking:	boolean
	hasBicycleParking:	boolean
	hasLocalPublicTransport:	boolean
	hasPublicFacilities:	boolean
	hasLockerSystem:	boolean
	hasTaxiRank:	boolean
	hasTravelNecessities:	boolean
	hasSteplessAccess:	string
	hasMobilityService:	string
	hasWiFi:	boolean
	hasTravelCenter:	boolean
	hasRailwayMission:	boolean
	hasDBLounge:	boolean
	hasLostAndFound:	boolean
	hasCarRental:	boolean
	federalState:	string
	createdAt:	string (date-time)
	ril100Identifiers:	<div><div>▼ [ []</div><div><div>▼ { []</div><div>rilIdentifier: string</div><div>hasSteamPermission: boolean</div><div>geographicCoordinates: ▼ { []</div><div><div>type: string</div><div>latitude: number (float)</div><div>longitude: number (float)</div></div><div>}</div><div>main: boolean</div><div>}</div></div></div>
	productLine:	<div><div>▼ { []</div><div>productLine: string</div><div>segment: string</div><div>}</div></div>
		}
when:		string (date-time)
plannedWhen:		string (date-time)
prognosedWhen:		string (date-time)
delay:		integer (int32)
platform:		string

plannedPlatform: string  
prognosedPlatform: string  
prognosisType: string  
direction: string  
provenance: string  
line: **▼ { []**  
    operator: **▼ { []**  
        type: string  
        operatorId: string  
        name: string  
    }  
    type: string  
    fahrtNr: string  
    name: string  
    linePublic: boolean  
    productName: string  
    lineId: string  
    product: string  
    mode: string  
        **Enum:** BUS, TAXI, WATERCRAFT, TRAIN  
    createdAt: string (date-time)  
}  
origin: **▼ { []**  
    type: string  
    stopId: integer (int64)  
    name: string  
    location: **▼ { []**  
        latitude: number (double)  
        longitude: number (double)  
    }  
    products: **▼ { []**  
        nationalExpress: boolean  
        national: boolean  
        regionalExp: boolean  
        regional: boolean  
        suburban: boolean  
        bus: boolean  
        ferry: boolean  
        subway: boolean  
        tram: boolean  
        taxi: boolean  
    }  
    createdAt: string (date-time)  
    station: **▼ { []**  
        address: **▼ { []**  
            city: string  
            zipcode: string  
            street: string  
        }  
        location: **▼ { []**  
            latitude: number (float)  
            longitude: number (float)  
        }  
        operator: **▼ { []**  
            type: string  
            operatorId: string  
            name: string

```

    }
    timeTableOffice: ▼ { []
      email: string
      name: string
    }
    regionalbereich: ▼ { []
      number: integer (int32)
      name: string
      shortName: string
    }
    stationManagement: ▼ { []
      number: integer (int32)
      name: string
    }
    szentrale: ▼ { []
      number: integer (int32)
      publicPhoneNumber: string
      name: string
    }
    stationId: integer (int64)
    relevance: number (float)
    score: number (float)
    weight: number (float)
    type: string
    ril100: string
    nr: number (float)
    name: string
    category: number (float)
    priceCategory: number (float)
    hasParking: boolean
    hasBicycleParking: boolean
    hasLocalPublicTransport: boolean
    hasPublicFacilities: boolean
    hasLockerSystem: boolean
    hasTaxiRank: boolean
    hasTravelNecessities: boolean
    hasSteplessAccess: string
    hasMobilityService: string
    hasWiFi: boolean
    hasTravelCenter: boolean
    hasRailwayMission: boolean
    hasDBLounge: boolean
    hasLostAndFound: boolean
    hasCarRental: boolean
    federalState: string
    createdAt: string (date-time)
    ril100Identifiers: ▼ [ []
      ▼ { []
        rilIdentifier: string
        hasSteamPermission: boolean
        geographicCoordinates: ▼ { []
          type: string
          latitude: number (float)
          longitude: number (float)
        }
      }
    ]
  }

```

```

    main: boolean
  }
]
productLine: ▼ { []
  productLine: string
  segment: string
}
}
}
destination: ▼ { []
  type: string
  stopId: integer (int64)
  name: string
  location: ▼ { []
    latitude: number (double)
    longitude: number (double)
  }
  products: ▼ { []
    nationalExpress: boolean
    national: boolean
    regionalExp: boolean
    regional: boolean
    suburban: boolean
    bus: boolean
    ferry: boolean
    subway: boolean
    tram: boolean
    taxi: boolean
  }
  createdAt: string (date-time)
  station: ▼ { []
    address: ▼ { []
      city: string
      zipcode: string
      street: string
    }
    location: ▼ { []
      latitude: number (float)
      longitude: number (float)
    }
    operator: ▼ { []
      type: string
      operatorId: string
      name: string
    }
    timeTableOffice: ▼ { []
      email: string
      name: string
    }
    regionalbereich: ▼ { []
      number: integer (int32)
      name: string
      shortName: string
    }
    stationManagement: ▼ { []
      number: integer (int32)
      name: string

```

```

    }
    szentrale:
      ▼ { []
        number: integer (int32)
        publicPhoneNumber: string
        name: string
      }
    stationId: integer (int64)
    relevance: number (float)
    score: number (float)
    weight: number (float)
    type: string
    ril100: string
    nr: number (float)
    name: string
    category: number (float)
    priceCategory: number (float)
    hasParking: boolean
    hasBicycleParking: boolean
    hasLocalPublicTransport: boolean
    hasPublicFacilities: boolean
    hasLockerSystem: boolean
    hasTaxiRank: boolean
    hasTravelNecessities: boolean
    hasSteplessAccess: string
    hasMobilityService: string
    hasWiFi: boolean
    hasTravelCenter: boolean
    hasRailwayMission: boolean
    hasDBLounge: boolean
    hasLostAndFound: boolean
    hasCarRental: boolean
    federalState: string
    createdAt: string (date-time)
    ril100Identifiers:
      ▼ [ []
        ▼ { []
          rilIdentifier: string
          hasSteamPermission: boolean
          geographicCoordinates: ▼ { []
            type: string
            latitude: number (float)
            longitude: number (float)
          }
          main: boolean
        }
      ]
    productLine:
      ▼ { []
        productLine: string
        segment: string
      }
  }

cancelled: boolean
loadFactor: string
remarks:
  ▼ [ []
    ▼ { []

```

```
        type:      string
        code:      string
        text:      string
        createdAt: string (date-time)
      }
    ]
  createdAt:      string (date-time)
}
```

---

Suggestions, contact, support and error reporting;

Information URL: <https://railinsights.de> (<https://railinsights.de>)

Contact Info: [hello@helloverb.com](mailto:hello@helloverb.com) ([hello@helloverb.com](mailto:hello@helloverb.com))

License

<https://www.apache.org/licenses/LICENSE-2.0.txt>