

JSON Structure Documentation - FlightRank 2025

Top Level Structure

```
{
  "$id": "string",           // Service ID (optional)
  "metadata": {...},         // Search metadata
  "personalData": {...},     // User data
  "routeData": {...},        // Route information
  "data": {...},             // Flight options
  "ranker_id": "string",     // Unique request ID
  "request_time": "string"   // Request timestamp
}
```

metadata - Search Metadata

```
{
  "$id": "string",           // Service ID
  "searchType": integer      // Search type (0 or 1)
}
```

personalData - User Data

```
{
  "$id": "string",           // Service ID
  "profileId": integer,       // User ID
  "sex": boolean,             // Gender (true/false)
  "bySelf": boolean,          // Self booking
  "yearOfBirth": integer,     // Birth year
  "nationality": integer,     // Nationality code
  "companyID": integer,       // Company ID
  "isVip": boolean,           // VIP status
  "hasAssistant": boolean,    // Has assistant
  "isGlobal": boolean,        // Global status
  "frequentFlyer": string,     // Frequent flyer programs (e.g., "SU/S7")
  "position": null,           // Position (usually null)
  "pointOfSale": null,        // Point of sale (usually null)
  "industryEN": null,         // Industry (usually null)
  "grade": null               // Grade (usually null)
}
```

routeData - Route Information

```
{
  "$id": "string",           // Service ID
  "requestDate": "string",   // Request date and time (ISO format)
  "searchRoute": "string",   // Route in IATA codes format
  "requestDepartureDate": "string", // Desired departure date
  "requestReturnDate": "string" // Desired return date (null for one-way)
}
```

searchRoute Examples:

- "OVBKHV/KHVOVB" - round trip (Novosibirsk ↔ Khabarovsk)
- "MOWALA/ALAMOW" - round trip (Moscow ↔ Almaty)
- "CEKMOW" - one way (Chelyabinsk → Moscow)

data - Flight Options

Data structure:

```
{
  "$id": "string",
  "$values": [...] // Array of flight options
}
```

Flight Option

```
{
  "$id": "string",           // Service ID
  "id": "string",           // Unique option ID
  "validatingCarrier": "string", // Validating carrier code
  "category": integer,      // Ticket category (0, 1)
  "legs": [...],           // Flight legs (outbound/return)
  "pricings": [...]        // Pricing options
}
```

legs[] - Flight Legs

Each leg represents a direction (outbound or return):

```
{
  "$id": "string",
  "duration": "string",      // Total duration ("04:45:00")
  "segments": [...],        // Flight segments
  "departureAt": "string",   // Departure time (ISO format)
  "arrivalAt": "string",     // Arrival time (ISO format)
  "departureFrom": {...},    // Departure airport
}
```

```

    "arrivalTo": {...}           // Arrival airport
  }

```

segments[] - Flight Segments

Each segment is a separate flight:

```

{
  "$id": "string",
  "id": "string",               // Segment ID
  "departureAt": "string",      // Departure time
  "arrivalAt": "string",        // Arrival time
  "duration": "string",         // Segment duration
  "flightNumber": "string",     // Flight number
  "stopCount": integer,         // Number of stops
  "aircraft": {                 // Aircraft type
    "$id": "string",
    "code": "string"           // Aircraft code ("321", "77W", "SU9")
  },
  "marketingCarrier": {         // Marketing carrier
    "$id": "string",
    "code": "string",           // Airline code ("SU", "S7")
    "alliance": {               // Alliance (optional)
      "$id": "string",
      "code": "string"         // Alliance code ("AEROFLOT", "EMIRATES")
    }
  },
  "operatingCarrier": {...},    // Operating carrier (similar structure)
  "departureFrom": {...},       // Departure airport
  "arrivalTo": {...},           // Arrival airport
  "transfer": "string"          // Transfer time (if applicable)
}

```

Airport Structure

```

{
  "$id": "string",
  "airport": {
    "$id": "string",
    "id": integer,              // Airport ID
    "code": "string",           // Airport code
    "iata": "string",           // IATA code ("SVO", "LED")
    "icao": "string",            // ICAO code ("UUUU", "UWGG")
    "city": {                   // City
      "$id": "string",
      "id": integer,            // City ID
      "code": "string",         // City code
      "iata": "string",         // City IATA code
      "country": {              // Country

```

```

    "$id": "string",
    "id": integer,           // Country ID
    "codeA2": "string",     // 2-letter code ("RU", "KZ")
    "codeA3": "string"      // 3-letter code ("RUS", "KAZ")
  }
},
"terminal": "string"       // Terminal (optional)
}

```

pricings[] - Pricing Options

```

{
  "$id": "string",
  "id": "string",           // Pricing option ID
  "totalPrice": number,     // Total price
  "taxes": number,          // Taxes and fees
  "timeLimit": "string",    // Booking time limit
  "supplierId": "string",   // Supplier ID
  "ordinal": integer,       // Order number
  "category": integer,      // Fare category
  "currencyCode": "string", // Currency code ("RUB", "KZT")
  "corporateTariffCode": integer, // Corporate tariff code (can be null)
  "isSegDiscountVariant": boolean, // Discount variant
  "isSegDiscountAsExactValue": boolean, // Discount as exact value
  "labels": [...],          // Option labels
  "miniRules": [...],       // Fare rules
  "pricingInfo": [...],     // Detailed pricing information
  "additionalData": {...}   // Additional data
}

```

labels[] - Option Labels

Array of strings with option characteristics:

- "BestPrice" - best price
- "BestPriceDirect" - best price direct flight
- "BestPriceTravelPolicy" - best price within travel policy
- "BestPriceCorporateTariff" - best price corporate tariff
- "Convenience" - convenient option
- "MinTime" - minimum travel time

miniRules[] - Fare Rules

```

{
  "$id": "string",
  "category": integer,       // Rule category (31, 33)
  "monetaryAmount": integer, // Penalty amount
}

```

```
"statusInfos": boolean,           // Rule status
"currencyCode": "string"         // Penalty currency
}
```

pricingInfo[] - Detailed Pricing Information

```
{
  "$id": "string",
  "passengerType": integer,       // Passenger type (0 - adult)
  "price": number,               // Price
  "taxes": number,               // Taxes
  "passengerCount": integer,     // Number of passengers
  "selfPaid": boolean,           // Self-paid
  "isAccessTP": boolean,         // Travel policy compliance
  "faresInfo": [...]             // Fare information
}
```

faresInfo[] - Fare Information

```
{
  "$id": "string",
  "ticketDesignator": "string",  // Fare designator (can be null)
  "class": "string",             // Booking class ("Y", "C", "F")
  "cabinClass": integer,         // Service class (1=economy, 2=business,
4=premium)
  "seatsAvailable": integer,     // Available seats
  "fareFamilyKey": "string",     // Fare family key
  "baggageAllowance": {         // Baggage allowance
    "$id": "string",
    "quantity": integer,         // Baggage quantity/weight
    "type": integer,             // Measurement type
    "weightMeasurementType": integer
  },
  "applyToSegmentIds": [...]     // Segment IDs where fare applies
}
```