

# AutoCycle API User Guide

---

version 1.12.1

## Authentication Instructions

---

When given access to use the **AutoCycle API** (<https://api.economy.com/autocycle/v1>), you will receive an *access key* and an *encryption key*. Both authentication methods require the keys.

**HMAC signature** is created by using the encryption key to create a SHA256 hash of the concatenation of access key and *timestamp*. The timestamp must be formatted as *yyyy-MM-ddTHH:mm:ssZ* with the time converted to the Universal Coordinated Time (UTC). For example, *July 30th, 2012 5:03:28pm EST* would be represented as *2012-07-30T21:03:28Z*.

**Note:** The timestamp must be current and be retrieved just prior to the creation of the signature. If the timestamp is not within a 5 minute window of the current time, access will be denied and the service will return a 401 error.

The authentication process requires the access key, timestamp, and the computed signature to be passed in the header of the request. Here is an example of what the header should look like when using an access key of *DB73FDF0-043C-4018-A7EB-CFB57356BA22*, an encryption key of *7C7C2FEA-6D18-49A1-BEC9-193B67EAE87D* and a timestamp of *2012-08-02T14:25:20Z*.

AccessKeyId: DB73FDF0-043C-4018-A7EB-CFB57356BA22

Signature: A7808C5A67C422054364F195B16175308317930848232C6A08A77224F1017E83

Timestamp: 2012-08-02T14:25:20Z

For samples on how to create a signature, [click here](#) (note: the linked samples are for our Data Buffet API; however, the methods for creating HMAC signature are still applicable for AutoCycle API).

**oAuth Token** can be generated by calling an API endpoint, using API access key as *client\_id* and API encryption key as *client\_secret* and it will remain valid for 1 hour.

The *oauth2/token* endpoint is used to generate oAuth Token using your *access key as client\_id*, *encryption key as client\_secret* and *grant\_type as client\_credentials*. Following cURL request can be used to obtain an OAuth token.

```
curl -X POST \  
  https://api.economy.com/autocycle/v1/oauth2/token \  
  -H 'Content-Type: application/x-www-form-urlencoded' \  
  -d 'client_id=DB73FDF0-043C-4018-A7EB-CFB57356BA22' \  
  -d 'client_secret=47C7C2FEA-6D18-49A1-BEC9-193B67EAE87D' \  
  -d 'grant_type=client_credentials'
```

The response to the above request will have a new access token.

```
{  
  "token_type": "bearer",  
  "access_token": "SrZ5UkbzPn432zqMLgV3Ja",  
  "expires_in": 3600  
}
```

## Making an API Call

---

The current version of the *Autocycle API* (<https://api.economy.com/autocycle/v1>) has two endpoints

1. *POST /forecast*
2. *GET /vins*

### 1. POST /forecast

*/forecast* accepts a POST request. The *Content-Type* of this request needs to be *application/json*.

This endpoint has an optional *modelVersion* query parameter which follows a *v[YYYYMM]* format (e.g. v201711). If a value is not passed in, it will use the latest model version in our system.

The json payload needs to be in *raw* format (as opposed to *form-data*, *x-www-form-urlencoded*, etc.) and will need to have the following properties:

1. A **scenario** string (See "Model Input Values" section for enumeration).
2. A **vehicles** array (you can include up to 100 vehicles in this array). A vehicle object includes the following properties:

Property	Description	Required?
end_month	Closing month in YYYY-MM-01 format. Defaults to <i>start_month</i> + 59 months.	No
exterior_color	See "Model Input Values" section for enumeration.	Yes
interior_color	See "Model Input Values" section for enumeration.	Yes
mileage	Mileage at forecast horizon start. See the table below for input assumptions.	No
mileage_per_year	Mileage accrual per year. Set to 0 if total mileage is constant over the forecast horizon.	Yes
moodys_region	See "Model Input Values" section for enumeration. Defaults to "South".	No
state	US abbreviated states. Maps to moodys_region. If both moodys_region and state are present in an input, moodys_region will take priority. See "Model Input Values" for enumeration.	No
msrp	Overrides the MSRP found in the VIN dataset.	No
percentile	Percentile value in increments of 10 from 10 to 90.	No
sale_type	See "Model Input Values" section for enumeration.	Yes
start_month	Origination date in YYYY-MM-01 format. See the table below for input assumptions.	No

Property	Description	Required?
vin	Only the first 8 and 10th digit are used by the model, e.g. 137FA833_3. If the user sends in a full vin, the api returns this value back in this property. A short_vin property will contain the part of the vin used by the model	Yes

### *mileage and start\_month input assumptions*

mileage	start_month	Mileage accumulation start	Forecast period start
>0	Not null	start_month	start_month
>0	Null	current month	current month
=0	Not null	start_month	start_month
=0	Null	October before the vehicle model year	current month

### *Example Request*

Here is an example cURL request for two vehicles (same *vin* with two different time periods and *mileage\_per\_year* assumptions) run for a *baseline* scenario:

```
curl -X POST \
  https://api.economy.com/autocycle/v1/forecast/ \
  -H 'Authorization: Bearer SrZ5UkbzPn432zqMLgV3Ja' \
  -H 'content-type: application/json' \
  -d '{
    "scenario": "bl",
    "vehicles": [
      {
        "vin": "WA1DKAFP8CA107201",
        "mileage_per_year":10000,
        "sale_type":"as is",
        "moodys_region":"",
        "state": "NJ",
```

```
    "interior_color": "black",
    "exterior_color": "white",
    "start_month": "2018-01-01",
    "end_month": "2018-12-01"
  },
  {
    "vin": "WA1DKAFP8CA107201",
    "mileage_per_year": 10000,
    "sale_type": "as is",
    "moodys_region": "midwest",
    "state": "OH",
    "interior_color": "black",
    "exterior_color": "white",
    "start_month": "2018-01-01",
    "end_month": "2018-12-01"
  }
]
}'
```

## Response

---

The response for the api is a json array that has the same number of elements as the request *vehicles* array. The input properties are merged with the following if the vin was found in our database:

- Properties from the vin dataset
- Error code and message (0 and empty string if no error occurred)
- Start month of forecast
- End month of forecast
- A *forecastPrice* array of price values
- A *forecastMileage* array of mileage value for each forecast interval

If the vin was not found in our database, a non-zero error code with the accompanying message is returned.

Here is an example response:

```
[
  {
    "vin": "WA1DKAFP8CA107201",
    "mileage_per_year": 10000,
    "sale_type": "as is",
    "moodys_region": "Northeast",
    "state": "NJ",
    "interior_color": "black",
    "exterior_color": "white",
    "start_month": "2018-01-31",
    "end_month": "2018-12-31",
    "short_vin": "wa1dkafp_c",
    "make": "audi",
    "model": "q5",
    "model_year": 2012,
    "msrp": 43750,
    "med_msrp": 41825,
    "nada_sub_segment": "luxury compact utility",
    "fuel_type": "gas",
    "doors": "4",
    "liters": "3",
    "cylinders": "6",
    "drive_type": "all wheel drive",
    "body_type": "utility",
    "induction_type": "standard",
    "transmission": "n/a",
    "truck": 1,
    "car": 0,
    "luxury": 1,
    "trim_level": 1.046025,
    "vqi_adj": 0,
    "error_code": 0,
    "error_message": "",
    "forecastPrice": [
      20655,
      20725,
```

```
    20972,  
    20650,  
    20206,  
    19683,  
    19046,  
    18450,  
    17604,  
    16567,  
    15925,  
    15689  
  ],  
  "forecastMileage": [  
    0,  
    833,  
    1667,  
    2500,  
    3333,  
    4167,  
    5000,  
    5833,  
    6667,  
    7500,  
    8333,  
    9167  
  ]  
},  
{  
  "vin": "WA1DKAFP8CA107201",  
  "mileage_per_year": 10000,  
  "sale_type": "as is",  
  "moodys_region": "midwest",  
  "state": "OH",  
  "interior_color": "black",  
  "exterior_color": "white",  
  "start_month": "2018-01-31",  
  "end_month": "2018-12-31",  
  "short_vin": "wa1dkafp_c",  
  "make": "audi",
```

```
"model": "q5",
"model_year": 2012,
"msrp": 43750,
"med_msrp": 41825,
"nada_sub_segment": "luxury compact utility",
"fuel_type": "gas",
"doors": "4",
"liters": "3",
"cylinders": "6",
"drive_type": "all wheel drive",
"body_type": "utility",
"induction_type": "standard",
"transmission": "n/a",
"truck": 1,
"car": 0,
"luxury": 1,
"trim_level": 1.046025,
"vqi_adj": 0,
"error_code": 0,
"error_message": "",
"forecastPrice": [
    21453,
    21605,
    21799,
    21477,
    20884,
    20425,
    19881,
    19478,
    18477,
    17379,
    16687,
    16432
],
"forecastMileage": [
    0,
    833,
    1667,
```

```
2500,  
3333,  
4167,  
5000,  
5833,  
6667,  
7500,  
8333,  
9167  
]  
}  
]
```

## Model Input Values

---

### scenario

- bl - Baseline
- s1 - Stronger Near-Term Rebound
- s2 - Slower Near-Term Recovery
- s3 - Moderate Recession
- s4 - Protracted Slump
- s5 - Below-Trend Long-Term Growth
- s6 - Stagflation
- s7 - Next-Cycle Recession
- s8 - Low Oil Price

### moodys\_region

- northeast
- south

- midwest
- west

## **sale\_type**

- as is
- dealer
- fleet/lease
- manufacturer/factory
- repossession
- salvage
- unknown

## **interior\_color**

- . (a period for missing or unknown)
- black
- blue
- gray
- red
- tan
- white

## **exterior\_color**

- . (a period for missing or unknown)
- black
- blue
- brown

- gold
- gray
- green
- orange
- purple
- red
- white

### **state**

- AK
- AL
- AR
- AZ
- CA
- CO
- CT
- DC
- DE
- FL
- GA
- HI
- IA
- ID
- IL
- IN
- KS

- KY
- LA
- MA
- MD
- ME
- MI
- MN
- MO
- MS
- MT
- NC
- ND
- NE
- NH
- NJ
- NM
- NV
- NY
- OH
- OK
- OR
- PA
- PR
- RI
- SC
- SD

- TN
- TX
- UT
- VA
- VT
- WA
- WI
- WV
- WY

## State to region mapping

- IA, IL, IN, KS, MI, MN, MO, ND, NE, OH, SD, WI => Midwest
- CT, MA, ME, NH, NJ, NY, PA, PR, RI, VT, => Northeast
- AL, AR, DC, DE, FL, GA, KY, LA, MD, MS, NC, OK, SC, TN, TX, VA, WV => South
- AK, AZ, CA, CO, HI, ID, MT, NM, NV, OR, UT, WA, WY => West

## 2. GET /vins

This endpoint is helpful in finding out the list of vins and descriptive properties of vehicles with known make, model and model year.

/vins accept a GET request. This endpoint has four query parameters:

1. **vehicleMake** - *(Required parameter)* a string for make of the vehicle
2. **vehicleModel** - *(Required parameter)* a string for model of the vehicle
3. **vehicleModelYear** - *(Required parameter)* an integer for model year of the vehicle
4. **modelVersion** - *(Optional parameter)* Model Version is the version of the statistical/economic model, which is published every month and follows a v[YYYYMM] format (e.g. v201711). If a value is not passed in, it will use the latest model

version in our system.

### Example Request

```
curl -X GET \  
  'https://api.economy.com/autocycle/v1/vins?vehicleMake=Audi&vehicleModel=Q5&vehicleModelYear=2017&modelVersion=v201807' \  
  -H 'Authorization: Bearer SrZ5UkbzPn432zqMLgV3Ja' \  
  -H 'Content-Type: application/json'
```

with parameters as:

vehicleMake = "Audi"

vehicleModel = "Q5"

vehicleModelYear = 2017

modelVersion = "v201807"

## Response

---

The response for the api is a json array with all the possible vin combinations and descriptive properties of the vehicle for the input provided.

Here is an example response for the above example request:

```
[  
  {  
    "vin": "wa1c2afp",  
    "make": "audi",  
    "model": "q5",  
    "model_year": 2017,  
    "msrp": 40900,
```

```
    "nada_sub_segment": "luxury compact utility",
    "body_type": "utility",
    "fuel_type": "gas"
  },
  {
    "vin": "wa1d7afp",
    "make": "audi",
    "model": "q5",
    "model_year": 2017,
    "msrp": 46400,
    "nada_sub_segment": "luxury compact utility",
    "body_type": "utility",
    "fuel_type": "gas"
  },
  {
    "vin": "wa1m2afp",
    "make": "audi",
    "model": "q5",
    "model_year": 2017,
    "msrp": 43150,
    "nada_sub_segment": "luxury compact utility",
    "body_type": "utility",
    "fuel_type": "gas"
  },
  {
    "vin": "wa1w7afp",
    "make": "audi",
    "model": "q5",
    "model_year": 2017,
    "msrp": 54300,
    "nada_sub_segment": "luxury compact utility",
    "body_type": "utility",
    "fuel_type": "gas"
  },
  {
    "vin": "wa1v7afp",
    "make": "audi",
    "model": "q5",
```

```
    "model_year": 2017,  
    "msrp": 53200,  
    "nada_sub_segment": "luxury compact utility",  
    "body_type": "utility",  
    "fuel_type": "gas"  
  }  
]
```