

# API do Mercado Livre

---

- API do Mercado Livre
  - Resumo
  - Como a API pode ajudar a Led.Tools
    - 1) Controle e auditoria de estoque (incluindo Full)
    - 2) Vendas e performance comercial
    - 3) Logística e nível de serviço
    - 4) Reputação, tráfego e prova social
  - Base URL e autenticação
  - 1) OAuth — Autenticação
    - 1.1 Trocar authorization code por access\_token
    - 1.2 Renovar access\_token via refresh\_token
  - 2) Itens e buscas (catálogo do seller)
    - 2.1 Listar itens por vendedor (busca no site)
    - 2.2 Listar itens publicados por um usuário (privado)
    - 2.3 Detalhes de um item
  - 3) Estoque Full (Fulfillment)
    - 3.1 Estoque por user\_product (onde aparece Full)
  - 4) Categorias
    - 4.1 Detalhes de uma categoria (hierarquia)
  - 5) Pedidos (vendas)
    - 5.1 Buscar pedidos do seller
  - 6) Envios (shipments)
    - 6.1 Consultar um envio
  - 7) Perguntas e respostas
    - 7.1 Buscar perguntas por item
  - 8) Visitas (tráfego)
    - 8.1 Consultar visitas de um item (VIP)
  - 9) Opiniões de produto (reviews / rating)
    - 9.1 Consultar rating/opiniões de um produto
  - Referência oficial: “todos os endpoints”

## Resumo

A API do Mercado Livre é um conjunto de endpoints REST que permite integrar sistemas externos com as principais funcionalidades do marketplace, incluindo:

- **Autenticação e autorização** (OAuth2: `access_token` + `refresh_token`)
- **Catálogo / anúncios** (itens, variações, preço, frete, status)
- **Vendas** (pedidos, itens vendidos, valores, status)
- **Logística** (envios, status, rastreio)
- **Reputação e qualidade** (métricas, visitas, reviews/avaliações)

Na prática, ela permite que a Led.Tools automatize rotinas e construa relatórios sem depender de telas manuais do painel.

# Como a API pode ajudar a Led.Tools

## 1) Controle e auditoria de estoque (incluindo Full)

- Automatizar a extração diária de estoque por SKU/variação
- Criar alertas (ex.: Full zerado para itens top sellers)

O caminho “mais confiável” para Full é por `user_product_id` e estoque por `locations.type`, onde `Full` aparece como `meli_facility`. ([Mercado Livre Developers](#))

## 2) Vendas e performance comercial

- Consolidar pedidos por período
- Curva ABC por produto/categoria
- Receita estimada, ticket médio, itens mais vendidos

## 3) Logística e nível de serviço

- Acompanhar status de envios e atrasos
- Medir tempo médio até “postagem / pronto para enviar”

## 4) Reputação, tráfego e prova social

- Medir visitas por anúncio (VIP) e correlação com vendas
- Monitorar nota média e volume de avaliações (reviews) para produtos-chave

# Base URL e autenticação

**Base URL:** <https://api.mercadolivre.com>

**Header padrão (quase tudo):**

```
Authorization: Bearer ACCESS_TOKEN  
Content-Type: application/json
```

## 1) OAuth — Autenticação

### 1.1 Trocar authorization code por access\_token

**Endpoint**

- `POST /oauth/token`

**O que faz** Troca o `code` do OAuth por `access_token` e `refresh_token`. ([Mercado Livre Developers](#))

**Exemplo de request (form urlencoded)**

```
{  
  "grant_type": "authorization_code",
```

```

"client_id": "APP_ID",
"client_secret": "APP_SECRET",
"code": "CODE RECEBIDO",
"redirect_uri": "https://sua-url/callback"
}

```

## Exemplo de resposta (JSON)

```
{
  "access_token": "APP_USR-...",
  "token_type": "bearer",
  "expires_in": 21600,
  "scope": "offline_access read write",
  "user_id": 570565928,
  "refresh_token": "TG-..."
}
```

## Tipos

Campo	Tipo	Descrição
access_token	str	Token Bearer para chamadas autenticadas
token_type	str	Normalmente "bearer"
expires_in	int	Validade em segundos
scope	str	Escopos aprovados
user_id	int	ID do usuário (seller) autorizado
refresh_token	str	Token para renovar o access_token

## 1.2 Renovar access\_token via refresh\_token

### Endpoint

- POST /oauth/token

O que faz Renova o `access_token` com `refresh_token`. ([Mercado Livre Developers](#))

### Request

```
{
  "grant_type": "refresh_token",
  "client_id": "APP_ID",
  "client_secret": "APP_SECRET",
  "refresh_token": "TG-..."
}
```

**Resposta:** semelhante à anterior.

## 2) Itens e buscas (catálogo do seller)

A doc diferencia **recurso público** (busca por site) e **recurso privado** (itens do vendedor via conta), e explica isso explicitamente. ([Mercado Livre Developers](#))

### 2.1 Listar itens por vendedor (busca no site)

#### Endpoint

- GET /sites/{SITE\_ID}/search?seller\_id={SELLER\_ID}&limit=50&offset=0

**O que faz** Lista anúncios (tipicamente ativos) do vendedor no marketplace. ([Mercado Livre Developers](#))

#### Exemplo de resposta (resumido)

```
{  
    "paging": { "total": 1234, "offset": 0, "limit": 50 },  
    "results": [  
        {  
            "id": "MLB123",  
            "title": "Produto X",  
            "price": 199.9,  
            "currency_id": "BRL",  
            "available_quantity": 12,  
            "sold_quantity": 340,  
            "permalink": "https://produto.mercadolivre.com.br/MLB123",  
            "thumbnail": "https://http2.mlstatic.com/....jpg"  
        }  
    ]  
}
```

#### Tipos (principais)

Campo	Tipo	Descrição
paging	dict	Info de paginação
paging.total	int	Total de resultados
paging.offset	int	Offset atual
paging.limit	int	Tamanho da página
results	list[dict]	Lista de itens
results[].id	str	ID do item (MLB...)
results[].title	str	Título
results[].price	float	Preço atual

Campo	Tipo	Descrição
results[].available_quantity	int	Estoque “disponível”
results[].sold_quantity	int	Vendidos acumulados
results[].permalink	str	URL do anúncio
results[].thumbnail	str	URL de miniatura

## 2.2 Listar itens publicados por um usuário (privado)

### Endpoint

- GET /users/{USER\_ID}/items/search?limit=50&offset=0

**O que faz** Lista itens publicados pelo seller “pela conta” (recurso privado), geralmente mais útil quando a API “fecha” buscas públicas. ([Mercado Livre Developers](#))

### Exemplo de resposta

```
{
  "seller_id": 570565928,
  "results": ["MLB123", "MLB456"],
  "paging": { "total": 2, "offset": 0, "limit": 50 }
}
```

### Tipos

Campo	Tipo	Descrição
seller_id	int	ID do vendedor
results	list[str]	IDs dos itens
paginação	dict	Paginação

## 2.3 Detalhes de um item

### Endpoint

- GET /items/{ITEM\_ID}

**O que faz** Retorna o JSON completo do anúncio: preço, estoque, vendidos, categoria, shipping, fotos, variações, etc. (é a base de quase tudo). É também onde você obtém `user_product_id` para consultar estoque Full. ([Mercado Livre Developers](#))

### Exemplo de resposta (resumido)

```
{
  "id": "MLB123",
```

```

"title": "Produto X",
"category_id": "MLB1744",
"price": 199.9,
"available_quantity": 12,
"sold_quantity": 340,
"permalink": "https://produto.mercadolivre.com.br/MLB123",
"pictures": [{ "id": "1", "url": "https://http2.mlstatic.com/....jpg" }],
"shipping": { "free_shipping": true, "logistic_type": "fulfillment" },
"variations": [
  {
    "id": 10001,
    "user_product_id": "MLBU123456",
    "available_quantity": 3,
    "attribute_combinations": [{"name": "Cor", "value_name": "Preto"}]
  }
],
"user_product_id": "MLBU000000"
}

```

## Tipos (principais)

Campo	Tipo	Descrição
id	str	ID do item
title	str	Título
category_id	str	ID da categoria
price	float	Preço
available_quantity	int	Estoque disponível
sold_quantity	int	Vendidos acumulados
permalink	str	Link público
pictures	list[dict]	Lista de imagens
pictures[].url	str	URL da imagem
shipping	dict	Infos de frete
shipping.free_shipping	bool	Frete grátis
shipping.logistic_type	str	"fulfillment", "xd_drop_off", etc.
variations	list[dict]	Variações (se houver)
variations[].id	int	ID da variação
variations[].user_product_id	str	Identificador para estoque “por produto físico”
user_product_id	str	Pode existir no item (sem variações)

## 3) Estoque Full (Fulfillment)

A doc oficial orienta: para checar estoque (Full/Flex), você precisa obter `user_product_id` via `/items` (ou dentro de `variations`) e então chamar o endpoint de stock. ([Mercado Livre Developers](#))

### 3.1 Estoque por user\_product (onde aparece Full)

#### Endpoint

- `GET /user-products/{USER_PRODUCT_ID}/stock`

**O que faz** Retorna estoque por “locations”. Para **Full**, o `type` costuma ser `meli_facility`. Para estoque do seller, `selling_address`. A doc mostra essa abordagem e menciona rate limit (100 RPM). ([Mercado Livre Developers](#))

#### Exemplo de resposta

```
{  
    "user_product_id": "MLBU123456",  
    "locations": [  
        { "type": "meli_facility", "quantity": 5 },  
        { "type": "selling_address", "quantity": 9 }  
    ]  
}
```

#### Tipos

Campo	Tipo	Descrição
<code>user_product_id</code>	<code>str</code>	ID do produto físico
<code>locations</code>	<code>list[dict]</code>	Estoque por local
<code>locations[].type</code>	<code>str</code>	Tipo do local ( <code>meli_facility</code> = Full)
<code>locations[].quantity</code>	<code>int</code>	Quantidade naquele local

## 4) Categorias

### 4.1 Detalhes de uma categoria (hierarquia)

#### Endpoint

- `GET /categories/{CATEGORY_ID}`

**O que faz** Retorna metadados da categoria e hierarquia (`path_from_root`). A doc de categorização fala sobre dump/hierarquia e uso de IDs/nome. ([Mercado Livre Developers](#))

#### Exemplo de resposta (resumido)

```
{  
    "id": "MLB1744",  
    "name": "Ferramentas",
```

```

    "path_from_root": [
        { "id": "MLB1", "name": "Categorias" },
        { "id": "MLB1744", "name": "Ferramentas" }
    ]
}

```

## Tipos

Campo	Tipo	Descrição
id	str	ID da categoria
name	str	Nome
path_from_root	list[dict]	Caminho hierárquico
path_from_root[].id	str	ID do nó
path_from_root[].name	str	Nome do nó

## 5) Pedidos (vendas)

A doc de “Pedidos e opiniões / gerenciamento de vendas” lista `/orders/search?seller=...` como rota para gerenciar pedidos. ([Mercado Livre Developers](#))

### 5.1 Buscar pedidos do seller

#### Endpoint

- GET `/orders/search?seller={SELLER_ID}&order.date_created.from=YYYY-MM-DD&order.date_created.to=YYYY-MM-DD`

**O que faz** Busca pedidos do seller, filtrando por período/status etc. ([Mercado Livre Developers](#))

#### Exemplo de resposta (resumido)

```
{
    "paging": { "total": 2, "offset": 0, "limit": 50 },
    "results": [
        {
            "id": 999000111,
            "status": "paid",
            "date_created": "2026-02-20T12:00:00.000-03:00",
            "buyer": { "id": 12345 },
            "order_items": [
                {
                    "item": { "id": "MLB123", "title": "Produto X" },
                    "quantity": 2,
                    "unit_price": 199.9
                }
            ],
            "total_amount": 399.8
        }
    ]
}
```

```

    }
]
}

```

## Tipos

Campo	Tipo	Descrição
paging	dict	Paginação
results	list[dict]	Pedidos
results[].id	int	ID do pedido
results[].status	str	Status (ex.: paid, cancelled)
results[].date_created	str	Data/hora ISO
results[].buyer	dict	Comprador
results[].buyer.id	int	ID do comprador
results[].order_items	list[dict]	Itens do pedido
results[].order_items[].item	dict	Dados do item
results[].order_items[].quantity	int	Quantidade
results[].order_items[].unit_price	float	Preço unitário
results[].total_amount	float	Total do pedido

## 6) Envios (shipments)

A doc de Shipments explica que `/shipments/{shipment_id}` concentra os dados do envio e menciona a header `x-format-new: true` para JSON atualizado. ([Mercado Livre Developers](#))

### 6.1 Consultar um envio

#### Endpoint

- `GET /shipments/{SHIPMENT_ID}`
- Header recomendado (em alguns cenários): `x-format-new: true` ([Mercado Livre Developers](#))

**O que faz** Retorna dados logísticos do envio (status, endereço, etiquetas, etc.). ([Mercado Livre Developers](#))

#### Exemplo (resumido)

```
{
  "id": 888777666,
  "status": "ready_to_ship",
  "tracking_number": "BR123...",
  "sender_id": 570565928,
```

```
    "receiver_id": 12345
}
```

## Tipos

Campo	Tipo	Descrição
id	int	ID do envio
status	str	Status logístico
tracking_number	str	Código de rastreio (quando disponível)
sender_id	int	ID do remetente (seller)
receiver_id	int	ID do destinatário

## 7) Perguntas e respostas

A doc de Perguntas e Respostas lista [/questions/search?item=\\$ITEM\\_ID](#) para buscar perguntas. ([Mercado Livre Developers](#))

### 7.1 Buscar perguntas por item

#### Endpoint

- [GET /questions/search?item={ITEM\\_ID}](#)

**O que faz** Lista perguntas feitas em um anúncio. ([Mercado Livre Developers](#))

#### Exemplo (resumido)

```
{
  "total": 1,
  "questions": [
    {
      "id": 111222333,
      "item_id": "MLB123",
      "status": "ANSWERED",
      "text": "Tem na cor preta?",
      "date_created": "2026-02-20T10:00:00.000-03:00"
    }
  ]
}
```

## Tipos

Campo	Tipo	Descrição
total	int	Total de perguntas

Campo	Tipo	Descrição
questions	list[dict]	Lista de perguntas
questions[].id	int	ID da pergunta
questions[].item_id	str	Item relacionado
questions[].status	str	Status (ex.: ANSWERED)
questions[].text	str	Texto da pergunta
questions[].date_created	str	Data ISO

## 8) Visitas (tráfego)

A doc de Visits explica que é possível obter visitas por janelas de tempo e sites. ([Mercado Livre Developers](#))

### 8.1 Consultar visitas de um item (VIP)

**Endpoint (varia por recurso/guia, mas o módulo é “Visits”)**

- `GET /visits/items?ids={ITEM_ID}`

**Com janela de tempo**

- `GET /visits/items?ids={ITEM_ID}&date_from={DATE_FROM}&date_to={DATE_TO}`  
([Mercado Livre Developers](#))
- Recurso de visitas (VIP) por item / janela de tempo ([Mercado Livre Developers](#))

**O que faz** Mede tráfego do anúncio para analytics (ex.: correlação visita → conversão). ([Mercado Livre Developers](#))

**Exemplo (genérico/resumido)**

```
{
  "item_id": "MLB123",
  "visits": 1500,
  "date_from": "2026-02-01",
  "date_to": "2026-02-20"
}
```

### Tipos

Campo	Tipo	Descrição
item_id	str	ID do item
visits	int	Quantidade de visitas
date_from	str	Início (YYYY-MM-DD)

Campo	Tipo	Descrição
date_to	str	Fim (YYYY-MM-DD)

## 9) Opiniões de produto (reviews / rating)

A doc “Opiniões de produtos” menciona campo como `rating_average` e endpoints para consultar estrelas/opiniões. ([Mercado Livre Developers](#))

### 9.1 Consultar rating/opiniões de um produto

#### Endpoint (módulo Opiniões de produtos)

- `GET /reviews/item/{ITEM_ID}`

#### Opcional (quando existir vínculo com catálogo)

- `GET /reviews/item/{ITEM_ID}?catalog_product_id={CATALOG_PRODUCT_ID}` ([Mercado Livre Global Selling](#))
- Recurso “Opiniões de produtos” ([Mercado Livre Developers](#))

**O que faz** Coleta reputação/estrelas para analytics e qualidade de catálogo. ([Mercado Livre Developers](#))

#### Exemplo (genérico/resumido)

```
{
  "product_id": "MLB-PROD-XYZ",
  "rating_average": 4.6,
  "total_reviews": 120
}
```

#### Tipos

Campo	Tipo	Descrição
product_id	str	ID do produto
rating_average	float	Média de estrelas
total_reviews	int	Total de avaliações

## Referência oficial: “todos os endpoints”

Para anexar na apresentação, use o hub oficial:

- **API Docs (por unidade de negócio):** lista as áreas e endpoints disponíveis. ([Mercado Livre Developers](#))
- **Playlist explicando a API do Mercado Livre:** ([Consumindo a API do Mercado Livre](#))