DEVELOPER DOCS

F.A.C.T AI

■■■ ■■

API ■■■■, SDK ■■■, ■■ ■■■ ■ ■■■ ■■■■■.

■■ ■■	v2.0.0 (2026■ 3■)
■■■ ■■	F.A.C.T AI Platform v2.0
■■ ■■	Python 3.9+, Node.js 18+, REST (■■ ■■)
■■ ■■	ceo@4dvision.co.kr | TEL. 031-901-4823
■■ ■■	MOB. 010-9039-0329 | www.4dvision.co.kr

■ ■

1. ■■■■ ■■■ ■■, ■■■■, ■■, ■■■■
2. API ■■■■ ■■■■·■■·■■■·■■ API
3. SDK & ■■■■■ Python SDK, Node.js SDK, REST
4. ■■ ■■■ SAP ERP, ■■, Snowflake, Excel/CSV
5. ■■ & ■■■■ ■■, ■■, ■■ ■■, ■■■■■■
6. ■■ ■■ & ■■■■■ ■■ ■■ ■■, ■■■ ■■■
7. ■■■ ■■ ■■■ ■■■■

1. ■■■■

1.1 ■■■ ■■ ■ ■■■■

F.A.C.T AI(FourD AI Convergence Transformer)■ ■■ ■■·■■·■■■■ ■■■ ■■ AI Agent ■■ ■■ ■■■ ■■ ■■■■■■. ERP·MES·SCM·CRM ■ ■■■ ■■ ■■■■ ■■ ■■■■ ■■■■ ■■■ ■■, AI ■■, ■■■ ■■■■■ ■■ ■■ ■■■ ■■■ ■■■■■.

■■ ■■ ■■

Frontend	Next.js 14 + TypeScript — ■■■■, ■■, ■■■, ■■ ■■
Backend	Python FastAPI + Celery + Redis — REST API, ■■■ ■■ ■■
Database	PostgreSQL (■ ■■■■) + Snowflake (■■ ■■■)
AI Engine	Snowflake Cortex Analyst / Cortex Search / Cortex Agents
Auth	JWT Bearer Token + API Key (RSA ■■■ ■■)

■■■■ ■■■

```
■■■ ■■ (■■■ ■■■)
■→ F.A.C.T AI Agent Router
■→ Cortex Analyst (structured data: SQL ■■ ■■)
■→ Cortex Search (document RAG: ■■■ ■■ ■■)
■→ Cortex Agents (multi-step: ■■ ■■ ■■■■■■■)
■→ ■■ ■■ → ■■■ ■■ ■■ → ■■/■■■ ■■
```

1.2 5■ ■■■■

Step 1 — ■■ ■■ ■ API ■ ■■

```
# 1. https://app.factai.kr ■■ ■ ■■ ■■ ■■
# 2. Settings > API Keys > Create New Key ■■
# 3. ■ ■■·■■ ■■ ■■ ■ ■■

export FACT_API_KEY='fak_live_xxxxxxxxxxxxxxxxxxxx'
export FACT_BASE_URL='https://api.factai.kr/v2'
```

Step 2 — Python SDK ■■ ■ ■ ■■ ■■

```
pip install fact-ai-sdk

from fact_ai import FactClient

client = FactClient(api_key='fak_live_xxx')

# ■■■ ■■
response = client.agent.query(
question='■■ ■ 2■■ ■■■ ■■■ ■■■■',
context={'department': 'quality', 'line': 2}
)
print(response.answer)
print(response.data) # pandas DataFrame
```

Step 3 — ■■ ■■

```
# Response ■■
{
'answer': '2■■ ■■■■ ■■ ■■ 12.4% ■■...',
'data': DataFrame(...),
'chart': 'bar|line|pie',
'confidence': 0.94,
'sources': ['MES_DEFECT_LOG', 'ERP_PROD_ORDER']
}
```

1.3 API ■ ■■ ■ ■■

■■ API ■■■ Authorization ■■■ Bearer ■■■ ■■■■ ■■■.

```
# HTTP ■■ ■■
Authorization: Bearer fak_live_xxxxxxxxxxxxxxxxxxxx
Content-Type: application/json
X-Tenant-ID: your-company-id

# cURL ■■
curl -X POST https://api.factai.kr/v2/agent/query \
-H 'Authorization: Bearer $FACT_API_KEY' \
-H 'Content-Type: application/json' \
-d '{"question": "■■ ■ ■■ ■■■?"}'
```

API Key ■■	fak_live_* (■■■■) | fak_test_* (■■■■)
■ ■■	■■ 90■ — Settings■■ ■■ ■■
■■ ■■	read / write / admin — ■■ ■■ ■■ ■■
Rate Limit	1,000 req/min (Standard) | 5,000 req/min (Enterprise)

1.4 ■■■■ ■■

```
# ■■■■ Base URL
FACT_BASE_URL='https://sandbox.api.factai.kr/v2'

# ■■■■ ■■ ■ (fak_test_■ ■■)
FACT_API_KEY='fak_test_xxxxxxxxxxxxxxxxxxxx'

# ■■■■■ ■■ DB■ ■■ ■■ — ■ ■■■ ■■ ■■
```

2. API ■■■■

2.1 ■■■■ API [v2.0]

AI ■■■■ ■■, ■■ ■■, ■■ ■■ ■■■■■■■■.

POST /v2/agent/query	■■■ ■■ ■■ — ■■ ■■
POST /v2/agent/task	■■■ ■■ ■■ ■■ — task_id ■■
GET /v2/agent/task/{id}	■■ ■■ ■ ■■ ■■
GET /v2/agent/history	■■ ■■ ■■ ■■
DELETE /v2/agent/task/{id}	■■ ■■

POST /v2/agent/query — ■■ ■■

```
{
"question": "string (required) — ■■■ ■■■ ■■",
"context": {
"department": "quality|sales|purchase|production",
"date_range": {"from": "2026-01-01", "to": "2026-03-31"},
"filters": {"line": 2, "plant": "A"}
},
"output_format": "json|chart|report",
"language": "ko"
```

```
}
```

2.2 ■■ API [v2.0]

■■■ ■■, KPI ■■, ■■■ ■■ ■■ ■■■■■■■■■.

GET /v2/analytics/kpi	KPI ■■ ■■ ■ ■■■
GET /v2/analytics/kpi/{id}	■■ KPI ■■ ■ ■■■■
POST /v2/analytics/query	SQL ■■ ■■■ ■■
GET /v2/analytics/trend	■■■ ■■■ ■■
GET /v2/analytics/alert	■■■ ■■ ■■ ■■

```
# KPI ■■ ■■
GET /v2/analytics/kpi?department=quality.=monthly

# ■■
{
"kpis": [
{"id": "ppm", "name": "■■■(PPM)", "value": 12.4, "unit": "ppm"},
{"id": "defect_cnt","name": "■■ ■■", "value": 3, "unit": "■"}
],
"period": "2026-03",
"updated_at": "2026-03-28T09:00:00Z"
}
```

2.3 ■■■ API [v2.0]

■■■ ■■, ■■■ ■■, ■■■■ ■■■■■■■■■.

POST /v2/reports/generate	■■■ ■■ ■■ (PDF/PPT/Excel)
GET /v2/reports	■■■ ■■■ ■■
GET /v2/reports/{id}/download	■■■ ■■ ■■■■
POST /v2/reports/schedule	■■ ■■ ■■■ ■■
GET /v2/reports/templates	■■■ ■■■ ■■

```
# ■■■ ■■ ■■
POST /v2/reports/generate
{
"template_id": "weekly_quality",
"format": "pdf",
"period": {"from": "2026-03-01", "to": "2026-03-28"},
"recipients": ["manager@company.com"],
"auto_send": true
}
```

2.4 ■■ API [v1.5]

■■■ ■■ ■ ■■ ■■■■■ ■■■■■.

POST /v2/webhooks	■■ ■■■■■ ■■

GET /v2/webhooks	■■■ ■■ ■■
DELETE /v2/webhooks/{id}	■■ ■■

```
# ■■ ■■■ ■■
alert.triggered — ■■■ ■■ ■■
report.completed — ■■■ ■■ ■■
task.finished — ■■■ ■■ ■■
approval.requested — ■■ ■■ ■■

# ■■ ■■■■ ■■
{
"event": "alert.triggered",
"timestamp": "2026-03-28T09:15:00Z",
"data": {"kpi": "ppm", "value": 45.2, "threshold": 30}
}
```

3. SDK & ■■■■■

3.1 Python SDK [v1.2.0]

```
pip install fact-ai-sdk

from fact_ai import FactClient
from fact_ai.models import QueryContext, ReportRequest

client = FactClient(
api_key='fak_live_xxx',
base_url='https://api.factai.kr/v2', # ■■■
timeout=30,
)

# ■■■■ ■■
resp = client.agent.query('■■ ■ ■■ TOP5■?')

# KPI ■■
kpis = client.analytics.get_kpis(department='quality')

# ■■■ ■■
report = client.reports.generate(
template_id='weekly_quality',
format='pdf'
)
report.download('/tmp/report.pdf')
```

3.2 Node.js / TypeScript SDK [v1.1.0]

```
npm install @fact-ai/sdk

import { FactClient } from '@fact-ai/sdk';

const client = new FactClient({
apiKey: process.env.FACT_API_KEY!,
});

// ■■■■ ■■
const resp = await client.agent.query({
```

```
question: '■■ ■ ■■ ■■■?',
context: { department: 'sales' },
});
console.log(resp.answer);

// TypeScript ■■ ■■ ■■
const kpis: KpiList = await client.analytics.getKpis();
```

3.3 REST API ■■ ■■

```
# Python requests ■■
import requests

BASE = 'https://api.factai.kr/v2'
HEADERS = {
'Authorization': f'Bearer {API_KEY}',
'Content-Type': 'application/json',
}

r = requests.post(f'{BASE}/agent/query',
headers=HEADERS,
json={'question': '■■ ■■ ■■ ■■■?'})
print(r.json()['answer'])
```

4. ■■ ■■■

4.1 SAP ERP ■■

SAP RFC/BAPI■ ■■ ■■■ ■■■ ■■■■■■. SAP NetWeaver RFC SDK ■■ pyrfc ■■■■■■ ■■■■■.

```
pip install fact-ai-sdk[sap]

from fact_ai.connectors import SAPConnector

sap = SAPConnector(
host='sap-server.company.com',
sysnr='00',
client='100',
user='RFC_USER',
password='***',
)

# ■■ ■■ ■■■ ■■■
sap.sync_table('AUFK', # ■■ ■■ ■■
fields=['AUFNR','AUART','WERKS','GAMNG'],
target='fact_sap_prod_orders',
schedule='*/15 * * * *') # 15■■■
```

■■ BAPI	BAPI_PRODORD_GET_LIST, BAPI_GOODSMVT_GETDETAIL, BAPI_MATERIAL_GETLIST
■■■ ■■	■(Pull) ■■■ / Change Pointer / IDoc ■■■
■■ SAP ■■	RFC ■■, ■■■ ■■ (S_RFC, S_TABU_DIS)

4.2 ■■ WEHAGO ■■

```
from fact_ai.connectors import WEHAGOConnector

wehago = WEHAGOConnector(
client_id='YOUR_CLIENT_ID',
client_secret='YOUR_CLIENT_SECRET',
company_code='12345',
)

# ■■ ■■ ■■■
wehago.sync_accounting(
period='2026-03',
tables=['SLIP', 'ACCOUNT', 'VENDOR'],
)
```

4.3 Snowflake ■■ ■■

```
# .env ■■
SNOWFLAKE_ACCOUNT=your-account.ap-northeast-1
SNOWFLAKE_USER=FACT_AI_USER
SNOWFLAKE_PRIVATE_KEY_PATH=/secrets/snowflake_rsa.pem
SNOWFLAKE_WAREHOUSE=FACT_WH
SNOWFLAKE_DATABASE=FACT_DB

# Cortex Analyst ■■
CORTEX_ANALYST_SEMANTIC_FILE=@FACT_DB.SCHEMA.SEMANTIC_MODEL.yaml
```

4.4 Excel / CSV ■■■ ■■■

```
# API■ ■■ ■■ ■■■
curl -X POST https://api.factai.kr/v2/data/upload \
-H 'Authorization: Bearer $FACT_API_KEY' \
-F 'file=@production_data.xlsx' \
-F 'table_name=fact_excel_upload' \
-F 'auto_schema=true'

# Python SDK
client.data.upload('production_data.xlsx',
auto_schema=True,
date_columns=['date', 'created_at'])
```

5. ■■ & ■■■■

5.1 ■■ ■■

API Key	Bearer Token — ■■■ ■■ ■■
OAuth 2.0	■■■ ■■ — ■ ■·■■■ ■
SSO/SAML	■■■■■■ SSO ■■ — SAML 2.0 ■■
RSA ■■■	Snowflake ■■ — ■■■■ ■ ■■ ■■

5.2 ■■ ■■ ■■ ■■ (RBAC)

admin	■■ ■■■ ■■ — ■■■·■■·■■ ■■
manager	■■■ ■■·■■, ■■ ■■, ■■■ ■■
analyst	■■■ ■■, ■■, ■■■■ ■■
viewer	■■■■·■■■ ■■ ■■

5.3 ■■■ ■■

■■ ■■■	TLS 1.3 (■■ API ■■)
■■ ■■■	AES-256 (Snowflake ■■■■ ■■■)
■■	ISO 27001 · ISMS-P
■■ ■■	■■ API ■■ ■■ — 90■ ■■
■■■ ■■	AWS ap-northeast-2 (■■ ■■)

6. ■■ ■■ & ■■■■■■

6.1 HTTP ■■ ■■

200 OK	■■ ■■
201 Created	■■■ ■■ ■■
400 Bad Request	■■ ■■■■ ■■ — ■■ ■■ ■■
401 Unauthorized	API ■ ■■ ■■ ■■ — ■ ■■■
403 Forbidden	■■ ■■ — ■■ ■■
404 Not Found	■■■ ■■
429 Too Many Requests	Rate Limit ■■ — Retry-After ■■ ■■
500 Internal Server Error	■■ ■■ — ■■ ■■ ■■

6.2 ■■ ■■ ■■

```
{
"error": {
"code": "RATE_LIMIT_EXCEEDED",
"message": "API ■■ ■■■ ■■■■■■.",
"retry_after": 60,
"request_id": "req_abc123"
}
}
```

6.3 ■■ ■■ ■■

INVALID_API_KEY	API ■ ■■ ■■ ■■ ■■■■ ■■ ■

KEY_EXPIRED	API ■ ■■ — Settings■■ ■■
QUOTA_EXCEEDED	■■ ■■ ■■ ■■ — ■■ ■■■■■
CONTEXT_TOO_LARGE	■■ ■■■■ ■■ ■■ (■■ 8K ■■)
DATA_SOURCE_OFFLINE	■■ ■■■ ■■ ■■ ■■
MODEL_UNAVAILABLE	AI ■■ ■■ ■■ — ■■ ■ ■■■

7. ■■■ ■■

v2.0.0 (2026-03-01)

■■	Cortex Agents ■■ ■■■■ ■■■■■■■
■■	SAP ERP ■■■ Change Pointer ■■■
■■	■■■ API PPT ■■ ■■
■■	■■■■ ■■ ■■ 40% ■■
■■	Rate Limit 1,000 → 2,000 req/min (Standard)
■■	■■ ■■ ■■ ■■ (date_range ■■■■)

v1.5.0 (2025-12-01)

■■	■■ API v1.5 ■■
■■	Node.js / TypeScript SDK v1.1.0
■■	■■ WEHAGO ■■■ ■■■ ■■

■■ ■■ ■■

ceo@4dvision.co.kr | TEL. 031-901-4823 | MOB. 010-9039-0329 ■■
09:00~18:00 (KST) — Enterprise ■■: 24/7 ■■ ■■ ■■