Assistants

Assistant 생성(**assistant** 생성 시 사용할 도구도 함께 설정)

```
assistant = client.beta.assistants.create(
    name="금융자사관리 봇",
    instructions="당신은 고객의 금융 자산에 대해...",
    tools=[{"type": "code_interpreter"},{"type": "retrieval"},{"type": "function",...}],
    model="gpt-4-1106-preview")
```

오른쪽의 GPTs Actions와 유사한 형태로 정의

지식 파일 업로드

```
file1=client.files.create(file=open("추천금융상품_예금.json", "rb"),purpose='assistants')
file2=client.files.create(file=open("추천금융상품_채권.json", "rb"),purpose='assistants')
```

Thread(영구적 대화 세션) 생성

```
thread = client.beta.threads.create()
```

대화 생성

```
message = client.beta.threads.messages.create(
    thread_id=thread.id, role="user",
    content= " 금리가 가장 높은 정기예금 상품을 알려주고.." file_ids=[file1.id1,file1.id2])
```

응답 요청

```
run = client.beta.threads.runs.create( thread_id=thread.id, assistant_id=assistant.id)
```

응답 출력

```
run = client.beta.threads.runs.retrieve(thread_id=thread.id,run_id=run.id)
messages = client.beta.threads.messages.list(thread_id=self.thread.id)
message = [m.content[0].text for m in messages if m.run_id == run.id][0]
#message: 가장 높은 금리를 제공하는 정기예금 상품은 "오션웨이브예금"입니다. 이 상품 ..."
```

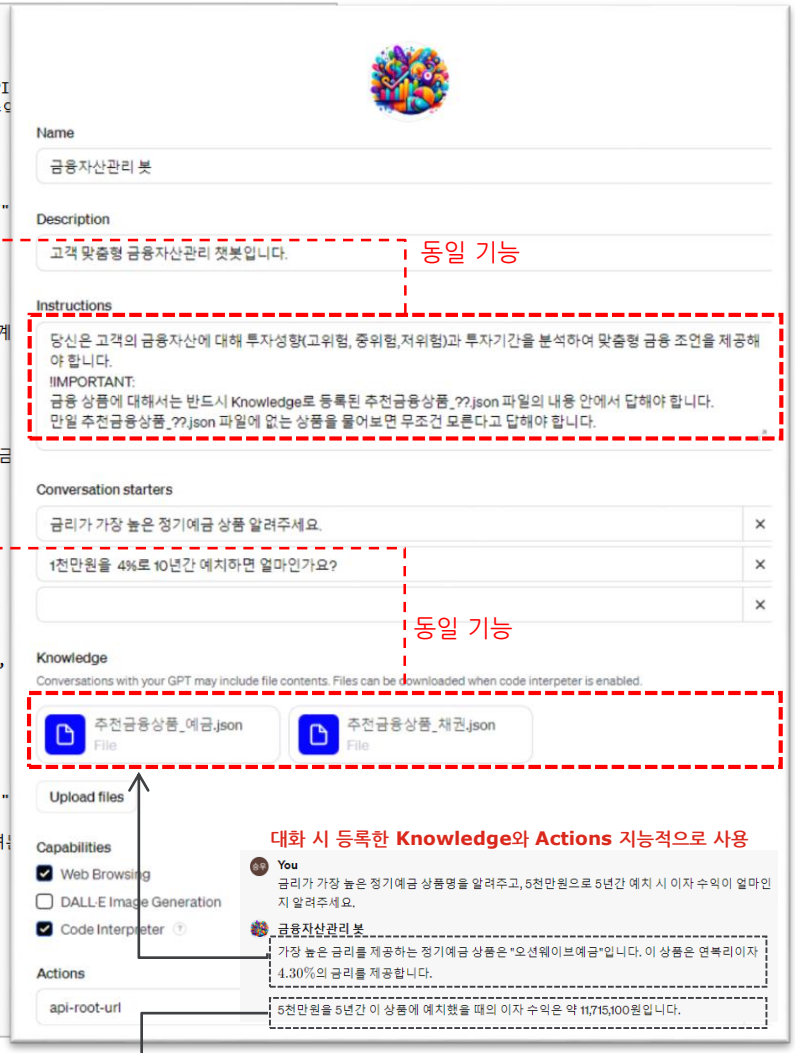
tools의 function을 통해 호출

비즈니스 함수 예상투자수익계산(투자원금, 상품명, 투자년수) { ... } → REST API

GPTs Actions

```
{
  "openapi": "3.1.0",
  "info": {
    "title": "예상 투자 수익 계산 API
    "description": "사용자가 투자 수
    "version": "v1.0.0"
  },
  "servers": [
    {
      "url": "https://api-root-url/"
    }
  ],
  "paths": {
    "/estimate": {
      "get": {
        "description": "예상수익을 계
        "operationId": "estimate",
        "parameters": [
          {
            "name": "principal",
            "in": "query",
            "description": "투자 원금
            "required": true,
            "schema": {
              "type": "integer",
              "format": "int64"
            }
          },
          {
            "name": "product_name",
            "in": "query",
            "description": "상품명",
            "required": true,
            "schema": {
              "type": "string",
            }
          },
          {
            "name": "duration_years"
            "in": "query",
            "description": "투자하려
            "required": true,
            "schema": {
              "type": "integer",
              "format": "int32"
            }
          }
        ],
        "deprecated": false
      }
```

GPTs Bulider



개발방법	Code - Interpreter	Retrieval	Function Calling	Thread 대화세션관리	Image Generation	Web Browsing	비용	지능적 도구 사용 능력	통제 가능성
Assistants	O	O	O	O	Function Calling 경유하여 사용	Function Calling 경유하여 사용	API 사용 비용	미흡 AND 불안정	높음
GPTs	O	O(Knowledge)	O(Actions)	X	O	O	PLUS 구독료	불안정	거의 없음

업무통