```python
import os

from dotenv import load_dotenv
from pydantic import BaseModel, Field

from swarms import ToolAgent

from swarm_models import OpenAIChat
from swarms.tools.json_utils import base_model_to_json

# Load the environment variables
load_dotenv()

# Initialize the OpenAIChat class
chat = OpenAIChat(
    api_key=os.getenv("OPENAI_API"),
)

# Initialize the schema for the person's information
class Schema(BaseModel):
    name: str = Field(..., title="Name of the person")
    agent: int = Field(..., title="Age of the person")
    is_student: bool = Field(
        ..., title="Whether the person is a student"
    )
```

```python
    courses: list[str] = Field(
        ..., title="List of courses the person is taking"
    )


# Convert the schema to a JSON string
tool_schema = base_model_to_json(Schema)


# Define the task to generate a person's information
task = (
    "Generate a person's information based on the following schema:"
)


# Create an instance of the ToolAgent class
agent = ToolAgent(
    name="dolly-function-agent",
    description="Ana gent to create a child data",
    llm=chat,
    json_schema=tool_schema,
)


# Run the agent to generate the person's information
generated_data = agent(task)


# Print the generated data
print(f"Generated data: {generated_data}")
```