

# Infosys

---

## Responsible AI Office

### Infosys Responsible AI Toolkit

#### Explainability for LLMs

#### API usage Instructions

### Contents

Introduction.....	2
Sentiment Detection and Explainability .....	2
Uncertainty and Coherence Metrics.....	3
Token Importance.....	4-5
Factuality Check by Internet Search.....	6
Graph of Thought Reasoning.....	6
Chain Of Thought .....	7
Thread Of Thought.....	8
Chain of Verification .....	9
Reread with Thread of Thought: .....	10
Logic of Thought:.....	11
Bulk Processing: .....	12
LLM Explainability endpoints in Moderation Layer.....	13
Chain Of Thought for RAG .....	14

# Introduction

**LLM Explain** provides explanations for Large Language Models using methods such as token importance, Graph of Thoughts, Logic of Thought, Thread of Thought, Chain of Thought and Search Augmentation. It evaluates the responses with metrics including uncertainty, relevancy, and coherence to ensure the reliability and clarity of Generative AI models' outputs.

Once API swagger page is populated as per instructions given in the github repository Readme file, click on 'try it out' to use required endpoints. Details of endpoints associated with LLM-Explainability tenet are outlined below.

## Sentiment Detection and Explainability

**Endpoint** – /rai/v1/llm-explainability/sentiment-analysis

Using this API, we can see sentiment analysis of the prompt along with token importance.


### Input:

**inputPrompt:** Replace the input prompt with the prompt you want to check

```
{
  "inputPrompt": "Unfortunately the movie served with bad visuals but the actors performed well"
}
```

### Response:

```
{
  "explanation": [
    {
      "predictedTarget": "Negative",
      "anchor": [
        "unfortunately",
        "bad"
      ],
      "explanation": "The model identified the sentiment of the prompt as negative based on two key words it picked up on: 'unfortunately' and 'bad'. These words are typically associated with a negative sentiment. In this context, despite the actors' good performance, the model interpreted the overall sentiment as negative due to the phrasing about the movie's bad visuals being presented unfortunately.",
      "token_importance_mapping": [
        {
          "token": "Unfortunately",
          "importance_score": 48,
          "position": 0
        },
        {
          "token": "bad",
          "importance_score": 52,
          "position": 5
        }
      ],
      "time_taken": 4.826
    }
  ]
}
```

 [Download](#)

# Uncertainty and Coherence Metrics

**Endpoint** - /rai/v1/llm-explainability/uncertainty

Using this API, we can evaluate the LLM response in terms of confidence level. to evaluate this, we have metrics such as uncertainty and coherence. Along with evaluation metrics, It provides corresponding explanation and corrective actions to improve the LLM response.

## Request Payload:

Example Value | Schema

```
{
  "inputPrompt": "Who are the co-founders of Infosys?",
  "response": "Infosys was co-founded by Narayana Murthy along with six other engineers: Nandan Nilekani, S. Gopalakrishnan (Kris), S. D. Shibulal, K. Dinesh, N. S. Raghavan, and Ashok Arora. Established in 1981, Infosys started with a modest capital of $250 and has since grown into one of the largest IT services companies in the world. Narayana Murthy, often regarded as the face of Infosys, played a pivotal role in shaping the company's culture and vision, while the combined efforts of all co-founders contributed to its remarkable growth and success in the global IT industry.",
  "endpointDetails": {
    "modelEndpointUrl": "http://localhost:8002/model/endpoint",
    "endpointInputParam": {
      "input_parameter": "inputs",
      "parameters": {
        "temperature": 0.2
      }
    },
    "endpointOutputParam": "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]"
  }
}
```

**inputPrompt:** provided your input prompt.

**response:** provide your input prompt response for the provided prompt.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl**: "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam**:
  - **input\_parameter**: "inputs", # Variable name used to pass Prompt/Query to endpoint payload.
  - **parameters**: # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam**: "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]"

## Response:

```
{
  "uncertainty": {
    "score": 0,
    "explanation": "The response provided is confident and specific. It lists all the co-founders correctly and includes their contributions and the company's journey. Furthermore, the details provided, especially around the establishment date and initial capital, are commonly known facts and are not debated.",
    "recommendation": "The response is certain and clear in answering the prompt about the Infosys co-founders and the narrative surrounding the initial formation of the company. Therefore, no changes are required for the prompt to improve this metric score.",
    "uncertainty_level": "Highly Certain"
  },
  "coherence": {
    "score": 95,
    "explanation": "The response is highly coherent. It logically begins with the names of the co-founders, then transitions smoothly into a brief history of Infosys' establishment and growth, and the roles that the co-founders played. All the information is logically connected and relevant to the given prompt.",
    "recommendation": "The response showcases effective coherence. It successfully maintains a logical flow and connects all the bits of information smoothly. Therefore, there are no recommendations for changing the input prompt to improve the coherence score as the flow and linkage of the information provided is excellent.",
    "coherence_level": "Highly Coherent"
  },
  "time_taken": 8.25
}
```

Download

# Token Importance

**Endpoint** - /rai/v1/llm-explainability/token\_importance

Using this API, we can get importance of each token in the input prompt.

**Input:**

Example Value | Schema

```
{
  "inputPrompt": "Who are the co-founders of Infosys?",
  "modelName": "gpt",
  "endpointDetails": {
    "modelEndpointUrl": "http://localhost:8002/model/endpoint",
    "endpointInputParam": {
      "input_parameter": "inputs",
      "parameters": {
        "temperature": 0.2
      }
    },
    "endpointOutputParam": "['output']['choices'][0]['text']"
  }
}
```

**inputPrompt:** provided your input prompt.

**modelName** [Optional Parameter]: provide model name. if you are using endpoint details keep model name as null.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl**: "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam**:
  - **input\_parameter**: "inputs", # Variable name used to pass Prompt/Query to endpoint payload.
  - **parameters**: # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam**: "['output']['choices'][0]['text']"

## Response:

Code	Details
200	<p>Response body</p> <pre>{   "token_importance_mapping": [     {       "token": "Who",       "importance_value": 1,       "position": 1     },     {       "token": "are",       "importance_value": 0.5,       "position": 2     },     {       "token": "the",       "importance_value": 0.5,       "position": 3     },     {       "token": "co-founders",       "importance_value": 1,       "position": 4     },     {       "token": "of",       "importance_value": 0.5,       "position": 5     },     {       "token": "Infosys?" </pre> <p>Response headers</p>

Response body
<pre>{   "token": "are",   "importance_value": 0.5,   "position": 2 }, {   "token": "the",   "importance_value": 0.5,   "position": 3 }, {   "token": "co-founders",   "importance_value": 1,   "position": 4 }, {   "token": "of",   "importance_value": 0.5,   "position": 5 }, {   "token": "Infosys?",   "importance_value": 1,   "position": 6 } ], "time_taken": 3.883, "token_cost": 0.015 }</pre>

## Factuality Check by Internet Search

**Endpoint** - /rai/v1/llm-explainability/serper\_response

Using this API, we can verify the LLM's response with the Google search engine to check for factual score and an explanation

**Input:**

**inputPrompt:** Replace the input prompt with the prompt you want to check

**lls\_response:** provide the response of input text

```
{
  "inputPrompt": "who are the co-founders of Infosys?",
  "llm_response": "Infosys, a global leader in technology services and consulting, was founded in 1981 by seven visionaries: N.R. Narayana Murthy, Nandan Nilekani, S. Gopalakrishnan, S.D. Shibulal, K. Dinesh, N.S. Raghavan, and Ashok Arora. These co-founders combined their expertise and entrepreneurial spirit to create a company that has since grown into one of the largest and most respected IT services firms in the world. Infosys, headquartered in Bangalore, India, has been instrumental in the global IT revolution, providing innovative solutions and services to clients across various industries. The founders' commitment to excellence and their forward-thinking approach laid a strong foundation for the company's enduring success."
}
```

**Response:**

```
{
  "internetResponse": [
    "Infosys was co-founded by Salil Parekh in January 2018."
  ],
  "metrics": [
    {
      "metricName": "Factuality Check",
      "score": 1,
      "explanation": [
        {
          "Fact": "The co-founders of Infosys are N.R. Narayana Murthy, Nandan Nilekani, S. Gopalakrishnan, S.D. Shibulal, K. Dinesh, N.S. Raghavan, and Ashok Arora.",
          "Reasoning": "The context provides a quote from an article which confirms that these seven individuals are indeed the co-founders of Infosys. However, it is unclear if this statement is valid as of today since it does not mention any dates. It is possible that some of these co-founders may have stepped down or new co-founders may have joined. Therefore, we cannot confirm the validity of this statement without more information.",
          "Judgement": "Fact Unclear"
        }
      ]
    }
  ],
  "time_taken": 5.478
}
```

## Graph of Thought Reasoning

**Endpoint** - /rai/v1/llm-explainability/got

Using this API, we can view a graph of thought reasoning for the given input. Here, we provide the final thought, score and consistency level.

**Input:**

**inputPrompt:** Replace the input prompt with the prompt you want to check

**modelName:** Provide deployment name from Azure

```
{
  "inputPrompt": "Who are the co-founders of Infosys?",
  "modelName": "gpt4"
}
```

## Response:

```
{
  "final_thought": "To answer this question, I will use my database of knowledge which includes data from verified and reliable sources. The co-founders of Infosys are several individuals bound by a shared vision and interest.\n\n1. N. R. Narayana Murthy: Murthy led Infosys for several years as its CEO and is recognized as one of the key individuals behind the IT revolution in India.\n\n2. Nandan Nilekani: Nilekani succeeded Murthy as Infosys CEO, and later served as the chairman of the Unique Identification Authority of India (UIDAI).\n\n3. Kris Gopalakrishnan: He served as the vice chairman of Infosys from 2011 to 2014 and was also its CEO and managing director from 2007 to 2011.\n\n4. S. D. Shibulal: Shibulal was the last of the seven co-founders to serve as the CEO and managing director of Infosys.\n\n5. K. Dinesh: Dinesh served as a board member of Infosys until 2011. He was also head of quality, information systems, and the communication design group.\n\n6. Ashok Arora: Ashok Arora left Infosys in 1989, but he was significantly involved in the initial years of the company.\n\n7. N. S. Raghavan: Raghavan was the first official employee of Infosys, and he served as its joint managing director.\n\nThese individuals are collectively known as the co-founders of Infosys, widely recognized for their collective contribution to the information technology industry in India.",
  "score": 55,
  "token_cost": 0.434,
  "consistency_level": "Moderately Consistent",
  "time_taken": 73.922
}
```

## Chain Of Thought

### Endpoint – /rai/v1/llm-explainability/cot

Using this API, we can get the ‘chain of thoughts’ the LLM went through to provide response to our prompt.

### Input:

Example Value | Schema

```
{
  "inputPrompt": "Which is the biggest country in the world?",
  "temperature": "0",
  "modelName": "GPT4",
  "endpointDetails": {
    "modelEndpointUrl": "http://localhost:8002/model/endpoint",
    "endpointInputParam": {
      "input_parameter": "inputs",
      "parameters": {
        "temperature": 0.2
      }
    },
    "endpointOutputParam": "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]"
  }
}
```

**inputPrompt:** provided your input prompt.

**temperature:** provide the temperature value.

**modelName** [Optional Parameter]: provide model name. if you are using endpoint details keep model name as null.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl:** "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam:**
  - **input\_parameter:** "inputs", # Variable name used to pass Prompt/Query to endpoint payload.
  - **parameters:** # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam:** "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]"

## Response:

Code	Description
200	Successful Response
	Media type application/json
	Controls Accept header.
	Example Value   Schema
	<pre>{   "explanation": "Russia is the largest country in the world by land area, covering approximately 17,098,242 square kilometers.",   "time_taken": 0.5,   "token_cost": 0 }</pre>

## Thread Of Thought

**Endpoint** – /rai/v1/llm-explainability/thot

Using this API, we can get the ‘thread of thoughts’ the LLM went through to provide response to our prompt, we can see how the LLM break down the prompt to correctly understand it and to generate response.

### Input:

Example Value   Schema
<pre>{   "inputPrompt": "Which is the biggest country in the world?",   "temperature": "0",   "modelName": "GPT4",   "endpointDetails": {     "modelEndpointUrl": "http://localhost:8002/model/endpoint",     "endpointInputParam": {       "input_parameter": "inputs",       "parameters": {         "temperature": 0.2       }     }   },   "endpointOutputParam": "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]" }</pre>

**inputPrompt:** provided your input prompt.

**temperature:** provide the temperature value.

**modelName** [Optional Parameter]: provide model name. if you are using endpoint details keep model name as null.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl:** "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam:**
  - **input\_parameter:** "inputs", # Variable name used to pass Prompt/Query to endpoint payload.



- **parameters:** # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam:** "[output][choices][0][text]"

## Response:

Code	Details
200	<p>Response body</p> <pre>{   "response": {     "result": "Russia",     "explanation": "The process to arrive at this answer involved identifying the country with the largest land area. According to the list of countries by total area outlined by various world geographical references and the United Nations, Russia is the largest country covering more than 1/8 of the Earth's inhabited land area.",     "time_taken": 6.047,     "token_cost": 0.01   } }</pre> <p><a href="#">Download</a></p>

## Chain of Verification

**Endpoint** – /rai/v1/llm-explainability/cov

Using this API, we can see the ‘chain of verification’ or questions the LLM asked itself to reach the response it gave us.

## Input:

Example Value | Schema

```
{
  "inputPrompt": "Which is the biggest country in the world?",
  "complexity": "simple",
  "modelName": "gpt4",
  "translate": "no",
  "endpointDetails": {
    "modelEndpointUrl": "http://localhost:8002/model/endpoint",
    "endpointInputParam": {
      "input_parameter": "inputs",
      "parameters": {
        "temperature": 0.2
      }
    },
    "endpointOutputParam": "[output][choices][0][text]"
  }
}
```

**inputPrompt:** provided your input prompt.

**Complexity:** provide the complexity.

**Translate:**


**temperature:** provide the temperature value.

**modelName** [Optional Parameter]: provide model name. if you are using endpoint details keep model name as null.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl**: "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam**:
  - **input\_parameter**: "inputs", # Variable name used to pass Prompt/Query to endpoint payload.
  - **parameters**: # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam**: "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]"

## Response:

Code	Details
200	<p>Response body</p> <pre>{   "original_question": "Which is the biggest country in the world?",   "baseline_response": "Russia",   "verification_questions": "1. Is Russia the largest country in the world by land area?\n2. Does Russia have the biggest land area among all countries?\n3. Is the biggest country in the world Russia?\n4. Is there any country larger than Russia in terms of land area?\n5. Can Russia be considered as the largest country globally?",   "verification_answers": "Question: 1. Is Russia the largest country in the world by land area? Answer: Yes\n\nQuestion: 2. Does Russia have the biggest land area among all countries? Answer: Yes\n\nQuestion: 3. Is the biggest country in the world Russia? Answer: Yes\n\nQuestion: 4. Is there any country larger than Russia in terms of land area? Answer: No\n\nQuestion: 5. Can Russia be considered as the largest country globally? Answer: Yes, Russia is considered the largest country globally by land area.\n",   "final_answer": "Russia is considered the largest country globally by land area.",   "time_taken": 33.899,   "token_cost": 0.014 }</pre> <p> <a href="#">Download</a></p>

## Reread with Thread of Thought:

Using this API, we can get the ‘thread of thoughts’ with rereading the prompt, the LLM went through to provide response to our prompt, we can see how the LLM break down the prompt to correctly understand it and to generate response

**Endpoint** – /rai/v1/llm-explainability/reread\_reasoning

**Input:**

Example Value	Schema
<pre>{   "inputPrompt": "Who are the co-founders of Infosys?",   "modelName": "GPT4",   "endpointDetails": {     "modelEndpointUrl": "http://localhost:8002/model/endpoint",     "endpointInputParam": {       "input_parameter": "inputs",       "parameters": {         "temperature": 0.2       }     },     "endpointOutputParam": "[ 'output' ][ 'choices' ][ 0 ][ 'text' ]"   } }</pre>	

**inputPrompt**: provided your input prompt.

**modelName** [Optional Parameter]: provide model name. if you are using endpoint details keep model name as null.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl**: "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam**:
  - **input\_parameter**: "inputs", # Variable name used to pass Prompt/Query to endpoint payload.
  - **parameters**: # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam**: "[output][choices][0][text]"

## Response:

Code	Details
200	<p>Response body</p> <pre>{   "response": {     "result": "N. R. Narayana Murthy, Nandan Nilekani, N. S. Raghavan, S. Gopalakrishnan, S. D. Shibulal, K. Dinesh, Ashok Arora",     "explanation": "Infosys was co-founded by seven individuals. They include N. R. Narayana Murthy, Nandan Nilekani, N. S. Raghavan, S. Gopalakrishnan, S. D. Shibulal, K. Dinesh, and Ashok Arora. Each of them played a significant role in the formation and growth of the company since its inception in 1981."   },   "time_taken": 14.806,   "token_cost": 0.015 }</pre> <p>Response headers</p>

## Logic of Thought:

Using this API, we can get response for the logical reasoning prompts with better reasoning and decisions

**Endpoint** – /rai/v1/llm-explainability/reread\_reasoning

## Input:

Example Value	Schema
<pre>{   "inputPrompt": "Who are the co-founders of Infosys?",   "llmResponse": "Infosys was co-founded by Narayana Murthy along with six other engineers: Nandan Nilekani, S. Gopalakrishnan (Kris), S. D. Shibulal, K. Dinesh, N. S. Raghavan, and Ashok Arora. Established in 1981, Infosys started with a modest capital of \$250 and has since grown into one of the largest IT services companies in the world. Narayana Murthy, often regarded as the face of Infosys, played a pivotal role in shaping the company's culture and vision, while the combined efforts of all co-founders contributed to its remarkable growth and success in the global IT industry.",   "modelName": "GPT4",   "endpointDetails": {     "modelEndpointUrl": "http://localhost:8002/model/endpoint",     "endpointInputParam": {       "input_parameter": "inputs",       "parameters": {         "temperature": 0.2       }     },     "endpointOutputParam": "[output][choices][0][text]"   } }</pre>	

**inputPrompt**: provided your input prompt.

**Response** [Optional Parameter]: provide your input prompt response for the provided prompt.

**modelName** [Optional Parameter]: provide model name. if you are using endpoint details keep model name as null.

**endpointDetails** [Optional Parameter]: If model is deployed and available as an inference endpoint, this parameter can be configured to use that model else you keep it as null.

- **modelEndpointUrl**: "YOUR ENDPOINT", Reachable inference endpoint of the model. Ex: <http://localhost:8002/model/endpoint>
- **endpointInputParam**:
  - **input\_parameter**: "inputs", # Variable name used to pass Prompt/Query to endpoint payload.
  - **parameters**: # Additional parameters to be passed in endpoint payload.
- **endpointOutputParam**: "["output"]['choices']["0"]['text']"

## Response:

Code Details

200

Response body

```

{
  "response": {
    "Propositions": {
      "A": "Infosys was co-founded by Narayana Murthy and six other engineers",
      "B": "Narayana Murthy played a pivotal role in shaping the company's culture and vision",
      "C": "All co-founders contributed to its remarkable growth and success"
    },
    "Logical Expression": "(A ∧ B) → C",
    "Extended Logical Expression": "¬A ∨ ¬B ∨ C",
    "Law used to extend the logical expression": "Contrapositive Law: The Contrapositive Law states that a conditional statement is true if and only if its contrapositive is true. The contrapositive of (A ∧ B) → C is ¬C → ¬(A ∧ B), which can be further expanded using De Morgan's law to ¬A ∨ ¬B ∨ C",
    "Extended Logical Information": "Either Infosys was not co-founded by Narayana Murthy and six other engineers, or Narayana Murthy did not play a pivotal role in shaping the company's culture and vision, or all co-founders contributed to its remarkable growth and success.",
    "Explanation": "The co-founders of Infosys are Narayana Murthy, Nandan Nilekani, S. Gopalakrishnan (Kris), S. D. Shibulal, K. Dinesh, N. S. Raghavan, and Ashok Arora. Infosys was established in 1981, starting with a modest capital of $250, and has grown into one of the largest IT services companies in the world. Narayana Murthy is often seen as the face of Infosys because he played a significant role in shaping the company's culture and vision. However, the input also presents a disjunction stating that either Infosys was not co-founded by Narayana Murthy and six other engineers, or Narayana Murthy did not shape the company, or all co-founders did not contribute to its success. This is a logical contradiction, as the first part of the input confirms all these points."
  },
  "time_taken": 40.903,
  "token_cost": 0.028
}

```

Download

## Bulk Processing:

Using this API, we can get response for some set of prompts instead of single prompt with all methods or specific methods.

**Endpoint:** /rai/v1/llm-explainability/bulk\_processing

**Input:**

**payload** \* required  
object

Edit Value | Schema

```

{
  "methods": [
    "CoT"
  ],
  "responseFileType": "json",
  "userId": "admin"
}

```

**file** \* required  
string(\$binary)

Choose File Excel\_File 1 (3).xlsx

**methods:** Specify the method names you want to check or get an explanation for.

**responseFileType:** Indicate the desired response file type as json or excel.

**userId:** Provide the user ID.

**File:** Upload the file containing the input prompts and responses.

### Response:

Code	Details
200	<p>Response body</p> <p><a href="#">Download file</a></p> <p>Response headers</p>

## LLM Explainability endpoints in Moderation Layer

The following LLM reasoning endpoints are currently available in the Moderation Layer ([responsible-ai-moderationlayer](#)) repository. Please follow the setup instructions in the [README](#) file of the moderation layer repository to configure them. Ensure that the service is up and running to execute.

## Chain Of Thought for RAG

### Endpoint – /rai/v1/moderations/healthcareopenaiCOT

Using this API, we can get the ‘chain of thoughts’ the LLM went through to provide response to our prompt, adding in example prompt response to tell the LLM which details to be included in the response and what format the response should be in. If you are using RAG based application, you can generate chunks from vector storage add the respective context of RAG files along with the actual prompt. This API will consider context and generates explanation for the same.

### Input:

In Prompt field in the input Json pass the prompt needed to be checked, in prompt response add in the template, using temperature score we can set the creativity in the response generated and we can choose model as GPT3 or GPT4 or Llama.

POST

/rai/v1/moderations/healthcareopenaiCOT Explain Chain of Thought Text

Parameters

No parameters

Cancel

Request body required

application/json

```
{
  "Prompt": "Which is the biggest country in the world?",
  "PromptResponse": "The largest country in the world by area is Russia. To determine this, we can refer to widely accepted global records and geographic data. Here's a step-by-step explanation of how we can confirm this information:\n\n1. **Definition of 'Biggest'**: First, we need to clarify what 'biggest' means in this context. It typically refers to the total area of a country, which includes land and water within the international boundaries.\n2. **Source of Information**: We can look at reputable sources such as the CIA World Factbook, the United Nations, or various educational resources that provide information on country sizes.\n3. **Comparison**: By comparing the total area of all countries, we can determine which one is the largest. This comparison is generally available in the form of lists or rankings based on area.\n4. **Consensus**: There is a global consensus that Russia is the largest country in the world, with a total area of about 17,098,242 square kilometers (6,601,668 square miles).\n5. **Verification**: To verify this information, we can check multiple reliable sources. However, as an AI developed by OpenAI, I don't browse the internet in real-time. Instead, I rely on the data I was trained on, which includes factual information up to my last update in 2023.\n6. **Responsibility**: To ensure the information is not misleading, it's important to note that geopolitical changes can affect country sizes, but as of my last update, there have been no changes affecting the status of Russia as the largest country.\n\nWhile I cannot provide a direct website link, you can easily verify this information by checking a current world atlas, a recent publication from a reputable source like the United Nations, or by searching for the largest countries in the world using a trusted search engine.",
  "temperature": "0",
  "model_name": "gpt4"
}
```

Execute Clear

### Response:

Code

Details

200

Response body

```
{
  "finishReason": "stop",
  "index": 0,
  "text": "You are correct that the largest country in the world by area is Russia. This is a well-known geographical fact that is documented by various reputable sources. Here's how you can confirm this information responsibly:\n\n1. **Understanding 'Largest'**: Clarify that 'largest' refers to the total area, including both land and water within a country's international boundaries.\n2. **Consulting Authoritative Sources**: Look for information from authoritative and up-to-date sources such as the CIA World Factbook, the United Nations, or educational institutions that maintain current geographic data.\n3. **Comparing Data**: These sources often provide lists or rankings of countries by area, allowing for easy comparison to determine which is the largest.\n4. **Global consensus**: There is a consensus among these sources that Russia is the largest country, with a total area of approximately 17,098,242 square kilometers (6,601,668 square miles).\n5. **Cross-Verification**: Cross-reference this information with multiple reliable sources to ensure accuracy. As an AI, I provide information based on the data I was trained on, which includes knowledge up to 2023.\n6. **Acknowledging Changes**: Recognize that geopolitical events can alter country sizes, but as of my last update, Russia remains the largest country by area.\n\nTo confirm this information, you can check the latest version of the CIA World Factbook online, visit the United Nations Statistics Division's website, or consult the most recent world atlas. These sources are regularly updated to reflect the most current information available. Remember to use trusted and authoritative sources to avoid misinformation.",
  "timeTaken": 13.599
}
```

Download