boto3==1.9.216\n   requests&gt;=2.20.0\n\n### 10.\n\nCreate an OpenAI API key on the \u301068\u2020API keys page\u2020platform.openai.com\u3011 by clicking &quot;+ Create new secret key&quot;.\n\n### 11.\n\nReplace your lambda_functions.py file with the \u301071\u2020provided lambda_function.py\u3011.\n\n   from ask_sdk_core.dispatch_components import AbstractExceptionHandler\n   from ask_sdk_core.dispatch_components import AbstractRequestHandler\n   from ask_sdk_core.skill_builder import SkillBuilder\n   from ask_sdk_core.handler_input import HandlerInput\n   from ask_sdk_model import Response\n   import ask_sdk_core.utils as ask_utils\n   import requests\n   import logging\n   import json\n\n   # Set your OpenAI API key\n   api_key = &quot;YOUR_API_KEY&quot;\n\n   logger = logging.getLogger(__name__)\n   logger.setLevel(logging.INFO)\n\n   class LaunchRequestHandler(AbstractRequestHandler):\n   &quot;&quot;&quot;Handler for Skill Launch.&quot;&quot;&quot;\n   def can_handle(self, handler_input):\n   # type: (HandlerInput) -&gt; bool\n\n   return ask_utils.is_request_type(&quot;LaunchRequest&quot;)(handler_input)\n\n   def handle(self, handler_input):\n   # type: (HandlerInput) -&gt; Response\n   speak_output = &quot;Chat G.P.T. mode activated&quot;\n\n   session_attr = handler_input.attributes_manager.session_attributes\n   session_attr[&quot;chat_history&quot;] = []\n\n   return (\n   handler_input.response_builder\n   .speak(speak_output)\n   .ask(speak_output)\n   .response\n   )\n\n   class GptQueryIntentHandler(AbstractRequestHandler):\n   &quot;&quot;&quot;Handler for Gpt Query Intent.&quot;&quot;&quot;\n   def can_handle(self, handler_input):\n   # type: (HandlerInput) -&gt; bool\n   return ask_utils.is_intent_name(&quot;GptQueryIntent&quot;)(handler_input)\n\n   def handle(self, handler_input):\n   # type: (HandlerInput) -&gt; Response\n   query = handler_input.request_envelope.request.intent.slots[&quot;query&quot;].value\n\n   session_attr = handler_input.attributes_manager.session_attributes\n   if &quot;chat_history&quot; not in session_attr:\n   session_attr[&quot;chat_history&quot;] = []\n   response = generate_gpt_response(session_attr[&quot;chat_history&quot;], query)\n   session_attr[&quot;chat_history&quot;].append((query, response))\n\n   return (\n   handler_input.response_builder\n   .speak(response)\n   .ask(&quot;Any other questions?&quot;)\n   .response\n   )\n\n   class CatchAllExceptionHandler(AbstractExceptionHandler):\n   &quot;&quot;&quot;Generic error handling to capture any syntax or routing errors.&quot;&quot;&quot;\n   def can_handle(self, handler_input, exception):\n   # type: (HandlerInput, Exception) -&gt; bool\n   return True\n\n   def handle(self, handler_input,", "pub_date": null, "extra": {"cited_message_idx": 90, "search_result_idx": null, "evidence_text": "source", "cloud_doc_url": null}}}], "content_references": [{"matched_text": "\u301089\u2020source\u3011", "start_idx": 549, "end_idx": 560, "alt": null, "type": "webpage_extended", "title": "GitHub - Joao208/alexa-chatgpt: This is an Alexa bot developed with the help of OpenAI&#x27;s ChatGPT language model. With it, you can converse naturally with Alexa, answering questions and performing tasks based on its language training.", "url": "https://github.com/Joao208/alexa-chatgpt", "pub_date": null, "snippet": "   *

```
\u301012\u2020 GitHub Skills \u2020skills.github.com\u3011\n    * \u301013\u2020 Blog
\u2020github.blog\u3011\n\n  * Solutions \n\nBy size\n    * \u301014\u2020 Enterprise
\u3011\n    * \u301015\u2020 Teams \u3011\n    * \u301016\u2020 Startups \u3011\n\nBy
industry\n    * \u301017\u2020 Healthcare \u3011\n    * \u301018\u2020 Financial
services \u3011\n    * \u301019\u2020 Manufacturing \u3011\n\nBy use case\n    *
\u301020\u2020 CI/CD &amp; Automation \u3011\n    * \u301021\u2020 DevOps \u3011\n
* \u301022\u2020 DevSecOps \u3011\n\n  * Resources \n\nTopics\n    * \u301023\u2020 AI
\u3011\n    * \u301024\u2020 DevOps \u3011\n    * \u301025\u2020 Security \u3011\n
* \u301026\u2020 Software Development \u3011\n    * \u301027\u2020 View all
\u3011\n\nExplore\n    * \u301028\u2020 Learning Pathways
\u2020resources.github.com\u3011\n    * \u301029\u2020 White papers, Ebooks, Webinars
\u2020resources.github.com\u3011\n    * \u301030\u2020 Customer Stories \u3011\n    *
\u301031\u2020 Partners \u2020partner.github.com\u3011\n\n  * Open Source \n\n    *
\u301032\u2020 GitHub Sponsors Fund open source developers \u3011\n\n    *
\u301033\u2020 The ReadME Project GitHub community articles \u3011\n\nRepositories\n
* \u301034\u2020 Topics \u3011\n    * \u301035\u2020 Trending \u3011\n    *
\u301036\u2020 Collections \u3011\n\n  * Enterprise \n\n    * \u301014\u2020
Enterprise platform AI-powered developer platform \u3011\n\nAvailable add-ons\n    *
\u301037\u2020 Advanced Security Enterprise-grade security features \u3011\n    *
\u301038\u2020 GitHub Copilot Enterprise-grade AI features \u3011\n    *
\u301039\u2020 Premium Support Enterprise-grade 24/7 support \u3011\n\n  *
\u301040\u2020Pricing\u3011 \n\nSearch or jump to...\n\n# Search code, repositories,
users, issues, pull requests...\n\n Search \n\nClear  \n\n\u301041\u2020 Search syntax
tips \u2020docs.github.com\u3011 \n\n#  Provide feedback \n\nWe read every piece of
feedback, and take your input very seriously.\n\nInclude my email address so I can be
contacted\n\nCancel  Submit feedback \n\n#  Saved searches \n\n## Use saved searches
to filter your results more quickly\n\nName\n\nQuery\n\nTo see all available
qualifiers, see our \u301041\u2020documentation\u2020docs.github.com\u3011. \n\nCancel
Create saved search \n\n\u30101\u2020 Sign in \u3011 \n\n\u301042\u2020 Sign up \u3011
Reseting focus\n\nYou signed in with another tab or window. Reload to refresh your
session. You signed out in another tab or window. Reload to refresh your session. You
switched accounts on another tab or window. Reload to refresh your session.   Dismiss
alert\n\n{{ message }}\n\n\u301043\u2020 Joao208 \u3011  / \u301044\u2020alexa-
chatgpt\u3011  Public\n\n  * \u301045\u2020 Notifications \u3011 You must be signed in
to change notification settings\n  * \u301045\u2020 Fork 48 \u3011 \n  *
\u301045\u2020 Star 190 \u3011\n\nThis is an Alexa bot developed with the help of
OpenAI&#x27;s ChatGPT language model. With it, you can converse naturally with Alexa,
answering questions and performing tasks based on its language training.
\n\n\u301046\u2020bit.ly/3wukdiz\u2020bit.ly\u3011 \n\n\u301047\u2020 190 stars \u3011
\u301048\u2020 48 forks \u3011 \u301049\u2020 Branches \u3011 \u301050\u2020 Tags
\u3011 \u301051\u2020 Activity \u3011 \n\n\u301045\u2020 Star \u3011\n\n\u301045\u2020
Notifications \u3011 You must be signed in to change notification settings\n\n  *
\u301044\u2020 Code \u3011\n  * \u301052\u2020 Issues 4 \u3011\n  * \u301053\u2020
Pull requests 0 \u3011\n  * \u301054\u2020 Actions \u3011\n  * \u301055\u2020 Projects
0 \u3011\n  * \u301056\u2020 Security \u3011\n  * \u301057\u2020 Insights \u3011\n\n
Additional navigation options  \n\n  * \u301044\u2020 Code \u3011 \n  * \u301052\u2020
Issues \u3011 \n  * \u301053\u2020 Pull requests \u3011 \n  * \u301054\u2020 Actions
\u3011 \n  * \u301055\u2020 Projects \u3011 \n  * \u301056\u2020 Security \u3011 \n  *
\u301057\u2020 Insights \u3011 \n\n# Joao208/alexa-chatgpt\n\nThis commit does not
```

belong to any branch on this repository, and may belong to a fork outside of the
repository.\n\n\u00a0main\n\n\u301049\u2020Branches\u3011\u301050\u2020Tags\u3011\n\nG
o to file \n\nCode\n\n## Folders and files \n\nName Name \n\nLast commit message
\n\nLast commit date   \n\n## Latest commit\n\n## History\n\n\u301058\u202010
Commits\u3011
\n\n\u301059\u2020.vscode\u3011\n\n\u301059\u2020.vscode\u3011\n\n\u301060\u2020.gitig
nore\u3011\n\n\u301060\u2020.gitignore\u3011\n\n\u301061\u2020README.md\u3011\n\n\u301
061\u2020README.md\u3011\n\n\u301062\u2020index.js\u3011\n\n\u301062\u2020index.js\u30
11\n\n\u301063\u2020local-debugger.js\u3011\n\n\u301063\u2020local-
debugger.js\u3011\n\n\u301064\u2020package.json\u3011\n\n\u301064\u2020package.json\u3
011\n\n\u301065\u2020util.js\u3011\n\n\u301065\u2020util.js\u3011\n\n\u301066\u2020yar
n.lock\u3011\n\n\u301066\u2020yarn.lock\u3011\n\nView all files   \n\n## Repository
files navigation \n\n * README \n\n# Alexa Skill with ChatGPT\n\nThis repository
contains an example of how to use OpenAI&#x27;s ChatGPT language model to create an
Alexa Skill.\n\n## How it works\n\nThis code connects to the OpenAI ChatGPT API and
sends Alexa&#x27;s questions to the model, which processes them and returns the
answers. Then, these responses are transmitted back to Alexa and presented to the
user.\n\n## Configuration\n\nYou will need to obtain an API key from OpenAI to use the
ChatGPT model. More information on how to do this can be found in the OpenAI API
documentation.\n\nOnce you have your API key, simply insert it into the index.js file
in the designated location.\n\n## Running the example\n\nThis example was developed
using Amazon&#x27;s ASK CLI. To run it, you will need to install the ASK CLI and
configure your Amazon Developer account.\n\nOnce configured, simply run the following
commands in the terminal:\n\n   ask init\n   ask deploy\n\nThis will deploy your
Skill to your Amazon Developer account and make it available for use with Alexa.\n\n##
Final considerations\n\nThis is just a basic example of how to use ChatGPT with Alexa.
You can expand this implementation to create more complex and personalized skills.
Enjoy!\n\n## About\n\nThis is an Alexa bot developed with the help of OpenAI&#x27;s
ChatGPT language model. With it, you can converse naturally with Alexa, answering
questions and performing tasks based on its language training.
\n\n\u301046\u2020bit.ly/3wUKdIz\u2020bit.ly\u3011 \n\n### Topics\n\n\u301067\u2020
javascript \u3011 \u301068\u2020 chat \u3011 \u301069\u2020 alexa \u3011
\u301070\u2020 axios \u3011 \u301071\u2020 chat-gpt \u3011 \n\n### Resources\n\nReadme
\n\n\u301051\u2020 Activity \u3011 \n\n### Stars\n\n\u301047\u2020 190 stars \u3011
\n\n### Watchers\n\n\u301072\u2020 7 watching \u3011 \n\n### Forks\n\n\u301048\u2020
48 forks \u3011 \n\n\u301073\u2020 Report repository \u3011 \n\n## \u301074\u2020
Releases \u3011\n\nNo releases published\n\n## \u301075\u2020 Packages 0 \u3011\n\nNo
packages published   \n\n## \u301076\u2020 Contributors 2 \u3011\n\n *
\u301043\u2020 \u3011 \u30104"}, {"matched_text": "\u301090\u2020source\u3011",
"start_idx": 969, "end_idx": 980, "alt": null, "type": "webpage_extended", "title":
"GitHub - bsab/Alexa-AI-Skill", "url": "https://github.com/bsab/Alexa-AI-Skill",
"pub_date": null, "snippet": " * \u301053\u2020 Pull requests 0 \u3011\n *
\u301054\u2020 Actions \u3011\n * \u301055\u2020 Projects 0 \u3011\n *
\u301056\u2020 Security \u3011\n * \u301057\u2020 Insights \u3011\n\n Additional
navigation options \n\n * \u301044\u2020 Code \u3011 \n * \u301052\u2020 Issues
\u3011 \n * \u301053\u2020 Pull requests \u3011 \n * \u301054\u2020 Actions \u3011
\n * \u301055\u2020 Projects \u3011 \n * \u301056\u2020 Security \u3011 \n *
\u301057\u2020 Insights \u3011 \n\n# bsab/Alexa-AI-Skill\n\nThis commit does not
belong to any branch on this repository, and may belong to a fork outside of the

repository.\n\n\u00a0main\n\n\u301049\u2020Branches\u3011\u301050\u2020Tags\u3011\n\nGo to file \n\nCode\n\n## Folders and files \n\nName Name \n\nLast commit message \n\nLast commit date \n\n## Latest commit\n\n## History\n\n\u301058\u20205 Commits\u3011 \n\n\u301059\u2020images\u3011\n\n\u301059\u2020images\u3011\n\n\u301060\u2020lambda\u3011\n\n\u301060\u2020lambda\u3011\n\n\u301061\u2020models\u3011\n\n\u301061\u2020models\u3011\n\n\u301062\u2020.gitignore\u3011\n\n\u301062\u2020.gitignore\u3011\n\n\u301046\u2020LICENSE\u3011\n\n\u301046\u2020LICENSE\u3011\n\n\u301063\u2020README.md\u3011\n\n\u301063\u2020README.md\u3011\n\n\u301064\u2020ask-resources.json\u3011\n\n\u301064\u2020ask-resources.json\u3011\n\n\u301065\u2020json_editor.json\u3011\n\n\u301065\u2020json_editor.json\u3011\n\n\u301066\u2020skill.json\u3011\n\n\u301066\u2020skill.json\u3011\n\nView all files \n\n## Repository files navigation \n\n * README \n * MIT license \n\n# Alexa AI\n\n[Image 0: License: MIT]\n\nAzure OpenAI GPT-4.0 will be integrated, and the project is Alexa AI: Boost your Alexa by enhancing its responses. This repository contains a tutorial on how to create a simple Alexa skill that uses the OpenAI API to generate responses from the model.\n\n[Image 1]\n\n## Prerequisites\n\n * An \u301067\u2020Amazon Developer account\u2020developer.amazon.com\u3011\n * An \u301068\u2020OpenAI API key\u2020platform.openai.com\u3011\n\n## Step-by-step tutorial\n\n### 1. \n\nLog in to your Amazon Developer account and navigate to the \u301069\u2020Alexa Developer Console\u2020developer.amazon.com\u3011.\n\n### 2.\n\nClick on &quot;Create Skill&quot; and name the skill &quot;Chat&quot;. Choose the primary locale according to your language.\n\n[Image 2: name your skill]\n\n### 3.\n\nChoose &quot;Other&quot; and &quot;Custom&quot; for the model.\n\n[Image 3: type of experience]\n\n[Image 4: choose a model]\n\n### 4.\n\nChoose &quot;Alexa-hosted (Python)&quot; for the backend resources.\n\n[Image 5: hosting services]\n\n### 5.\n\nYou now have two options:\n\n * Click on &quot;Import Skill&quot;, paste the link of this repository (\u301070\u2020https://github.com/k4l1sh/alexa-gpt.git\u3011) and click on &quot;Import&quot;. [Image 6: template]\n\nOr if you want to create the skill manually\n\n * Select &quot;Start from Scratch&quot; and click on &quot;Create Skill&quot;\n\n[Image 7: template]\n\n### 6.\n\nIn the &quot;Build&quot; section, navigate to the &quot;JSON Editor&quot; tab.\n\n### 7.\n\nIf you have directly imported the skill from this repository, just change the &quot;invocationName&quot; to &quot;chat&quot; or another preferred word for activation and proceed to step 12.\n\nHowever, if you chose to manually create the skill, replace the existing JSON content with the \u301065\u2020provided JSON content\u3011:\n\n {\n &quot;interactionModel&quot;: {\n &quot;languageModel&quot;: {\n &quot;invocationName&quot;: &quot;chat&quot;,\n &quot;intents&quot;: [\n {\n &quot;name&quot;: &quot;GptQueryIntent&quot;,\n &quot;slots&quot;: [\n {\n &quot;name&quot;: &quot;query&quot;,\n &quot;type&quot;: &quot;AMAZON.Person&quot;\n }\n],\n &quot;samples&quot;: [\n &quot;{query}&quot;\n]\n },\n {\n &quot;name&quot;: &quot;AMAZON.CancelIntent&quot;,\n &quot;samples&quot;: []\n },\n {\n &quot;name&quot;: &quot;AMAZON.HelpIntent&quot;,\n &quot;samples&quot;: []\n },\n {\n

&quot;name&quot;: &quot;AMAZON.StopIntent&quot;,\n
&quot;samples&quot;: []\n   },\n   {\n
&quot;name&quot;: &quot;AMAZON.NavigateHomeIntent&quot;,\n
&quot;samples&quot;: []\n   }\n   ],\n
&quot;types&quot;: []\n   }\n   }\n   }\n\n[Image 8:
json_editor]\n\n### 8.\n\nSave the model and click on &quot;Build Model&quot;.\n\n###
9.\n\nGo to &quot;Code&quot; section and add &quot;openai&quot; to requirements.txt.
Your requirements.txt should look like this:\n\n   ask-sdk-core==1.11.0\n
boto3==1.9.216\n   requests&gt;=2.20.0\n\n### 10.\n\nCreate an OpenAI API key on the
\u301068\u2020API keys page\u2020platform.openai.com\u3011 by clicking &quot;+ Create
new secret key&quot;.\n\n### 11.\n\nReplace your lambda_functions.py file with the
\u301071\u2020provided lambda_function.py\u3011.\n\n   from
ask_sdk_core.dispatch_components import AbstractExceptionHandler\n   from
ask_sdk_core.dispatch_components import AbstractRequestHandler\n   from
ask_sdk_core.skill_builder import SkillBuilder\n   from ask_sdk_core.handler_input
import HandlerInput\n   from ask_sdk_model import Response\n   import
ask_sdk_core.utils as ask_utils\n   import requests\n   import logging\n   import
json\n\n   # Set your OpenAI API key\n   api_key = &quot;YOUR_API_KEY&quot;\n\n
logger = logging.getLogger(__name__)\n   logger.setLevel(logging.INFO)\n\n   class
LaunchRequestHandler(AbstractRequestHandler):\n   &quot;&quot;&quot;Handler for
Skill Launch.&quot;&quot;&quot;\n   def can_handle(self, handler_input):\n
# type: (HandlerInput) -&gt; bool\n\n   return
ask_utils.is_request_type(&quot;LaunchRequest&quot;)(handler_input)\n\n   def
handle(self, handler_input):\n   # type: (HandlerInput) -&gt; Response\n
speak_output = &quot;Chat G.P.T. mode activated&quot;\n\n   session_attr =
handler_input.attributes_manager.session_attributes\n
session_attr[&quot;chat_history&quot;] = []\n\n   return (\n
handler_input.response_builder\n   .speak(speak_output)\n
        .ask(speak_output)\n   .response\n   )\n\n   class
GptQueryIntentHandler(AbstractRequestHandler):\n   &quot;&quot;&quot;Handler for
Gpt Query Intent.&quot;&quot;&quot;\n   def can_handle(self, handler_input):\n
# type: (HandlerInput) -&gt; bool\n   return
ask_utils.is_intent_name(&quot;GptQueryIntent&quot;)(handler_input)\n\n   def
handle(self, handler_input):\n   # type: (HandlerInput) -&gt; Response\n
query =
handler_input.request_envelope.request.intent.slots[&quot;query&quot;].value\n\n
session_attr = handler_input.attributes_manager.session_attributes\n   if
&quot;chat_history&quot; not in session_attr:\n
session_attr[&quot;chat_history&quot;] = []\n   response =
generate_gpt_response(session_attr[&quot;chat_history&quot;], query)\n
session_attr[&quot;chat_history&quot;].append((query, response))\n\n   return
(\n
handler_input.response_builder\n   .speak(response)\n
            .ask(&quot;Any other
questions?&quot;)\n   .response\n   )\n\n   class
CatchAllExceptionHandler(AbstractExceptionHandler):\n   &quot;&quot;&quot;Generic
error handling to capture any syntax or routing errors.&quot;&quot;&quot;\n   def
can_handle(self, handler_input, exception):\n   # type: (HandlerInput,
Exception) -&gt; bool\n   return True\n\n   def handle(self,

```
handler_input,"}], "gizmo_id": null, "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true, "message_type": null, "model_slug": "gpt-4o",
"default_model_slug": "gpt-4o", "parent_id": "f4dfc757-99ea-4881-bd4b-bc2bdae243a8",
"request_id": "8d1fb9c0887abadf-MXP", "timestamp_": "absolute"}, "recipient": "all",
"channel": null}, "parent": "f4dfc757-99ea-4881-bd4b-bc2bdae243a8", "children":
["aaa227ec-dd61-4e1d-94f8-77c0b3e15a55"]}, "aaa227ec-dd61-4e1d-94f8-77c0b3e15a55":
{"id": "aaa227ec-dd61-4e1d-94f8-77c0b3e15a55", "message": {"id": "aaa227ec-dd61-4e1d-
94f8-77c0b3e15a55", "author": {"role": "user", "name": null, "metadata": {}},
"create_time": 1728826727.347502, "update_time": null, "content": {"content_type":
"text", "parts": ["which one is best for lazy fringe coders like me? "]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d1fbb634ea6badf-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "cc71f4b3-541b-
40e9-9a3f-e386a87c95d6", "children": ["423f25ca-d543-4f1b-8c55-c04eda2a2c35"]},
"423f25ca-d543-4f1b-8c55-c04eda2a2c35": {"id": "423f25ca-d543-4f1b-8c55-c04eda2a2c35",
"message": {"id": "423f25ca-d543-4f1b-8c55-c04eda2a2c35", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728826739.015246,
"update_time": null, "content": {"content_type": "text", "parts": ["For &quot;lazy
fringe coders&quot; who want a quick setup with minimal effort, the **Joao208/alexa-
chatgpt** repository is likely the best option. Here&#x27;s why:\n\n1. **Ease of Use**:
The repository is designed to be simple. You just need to clone the repository, add
your OpenAI API key, and configure a few settings.\n   \n2. **Step-by-Step Setup**:
The project provides a clear step-by-step guide to setting up an Alexa skill and
integrating it with OpenAI. This reduces the need for digging into complex code or
spending a lot of time on troubleshooting.\n\n3. **Minimal Coding Involvement**: You
won\u2019t need to write much code beyond configuring a few environment variables,
making it ideal for those who want quick results without deep diving into
coding.\n\nIf you\u2019re looking for the most straightforward solution,
**Joao208/alexa-chatgpt** is a good fit for your needs.\n\nYou can find the repository
[here on GitHub](https://github.com/Joao208/alexa-
chatgpt)\u301089\u2020source\u3011.\n\n"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [{"start_ix": 944, "end_ix": 955,
"citation_format_type": "tether_og", "metadata": {"type": "webpage", "title": "GitHub
- Joao208/alexa-chatgpt: This is an Alexa bot developed with the help of OpenAI&#x27;s
ChatGPT language model. With it, you can converse naturally with Alexa, answering
questions and performing tasks based on its language training.", "url":
"https://github.com/Joao208/alexa-chatgpt", "text": "    * \u301012\u2020 GitHub
Skills \u2020skills.github.com\u3011\n    * \u301013\u2020 Blog
\u2020github.blog\u3011\n\n  * Solutions \n\nBy size\n    * \u301014\u2020 Enterprise
\u3011\n    * \u301015\u2020 Teams \u3011\n    * \u301016\u2020 Startups \u3011\n\nBy
industry\n    * \u301017\u2020 Healthcare \u3011\n    * \u301018\u2020 Financial
services \u3011\n    * \u301019\u2020 Manufacturing \u3011\n\nBy use case\n    *
\u301020\u2020 CI/CD &amp; Automation \u3011\n    * \u301021\u2020 DevOps \u3011\n
* \u301022\u2020 DevSecOps \u3011\n\n  * Resources \n\nTopics\n    * \u301023\u2020 AI
\u3011\n    * \u301024\u2020 DevOps \u3011\n    * \u301025\u2020 Security \u3011\n
* \u301026\u2020 Software Development \u3011\n    * \u301027\u2020 View all
\u3011\n\nExplore\n    * \u301028\u2020 Learning Pathways
```

```
\u2020resources.github.com\u3011\n    * \u301029\u2020 White papers, Ebooks, Webinars
\u2020resources.github.com\u3011\n    * \u301030\u2020 Customer Stories \u3011\n    *
\u301031\u2020 Partners \u2020partner.github.com\u3011\n\n  * Open Source \n\n    *
\u301032\u2020 GitHub Sponsors Fund open source developers \u3011\n\n    *
\u301033\u2020 The ReadME Project GitHub community articles \u3011\n\nRepositories\n
* \u301034\u2020 Topics \u3011\n    * \u301035\u2020 Trending \u3011\n    *
\u301036\u2020 Collections \u3011\n\n  * Enterprise \n\n    * \u301014\u2020
Enterprise platform AI-powered developer platform \u3011\n\nAvailable add-ons\n    *
\u301037\u2020 Advanced Security Enterprise-grade security features \u3011\n    *
\u301038\u2020 GitHub Copilot Enterprise-grade AI features \u3011\n    *
\u301039\u2020 Premium Support Enterprise-grade 24/7 support \u3011\n\n  *
\u301040\u2020Pricing\u3011 \n\nSearch or jump to...\n\n# Search code, repositories,
users, issues, pull requests...\n\n Search \n\nClear  \n\n\u301041\u2020 Search syntax
tips \u2020docs.github.com\u3011 \n\n#  Provide feedback \n\nWe read every piece of
feedback, and take your input very seriously.\n\nInclude my email address so I can be
contacted\n\nCancel  Submit feedback \n\n#  Saved searches \n\n## Use saved searches
to filter your results more quickly\n\nName\n\nQuery\n\nTo see all available
qualifiers, see our \u301041\u2020documentation\u2020docs.github.com\u3011. \n\nCancel
Create saved search \n\n\u30101\u2020 Sign in \u3011 \n\n\u301042\u2020 Sign up \u3011
Reseting focus\n\nYou signed in with another tab or window. Reload to refresh your
session. You signed out in another tab or window. Reload to refresh your session. You
switched accounts on another tab or window. Reload to refresh your session.   Dismiss
alert\n\n{{ message }}\n\n\u301043\u2020 Joao208 \u3011  / \u301044\u2020alexa-
chatgpt\u3011  Public\n\n  * \u301045\u2020 Notifications \u3011 You must be signed in
to change notification settings\n  * \u301045\u2020 Fork 48 \u3011 \n  *
\u301045\u2020 Star 190 \u3011\n\nThis is an Alexa bot developed with the help of
OpenAI&#x27;s ChatGPT language model. With it, you can converse naturally with Alexa,
answering questions and performing tasks based on its language training.
\n\n\u301046\u2020bit.ly/3wukdiz\u2020bit.ly\u3011 \n\n\u301047\u2020 190 stars \u3011
\u301048\u2020 48 forks \u3011 \u301049\u2020 Branches \u3011 \u301050\u2020 Tags
\u3011 \u301051\u2020 Activity \u3011 \n\n\u301045\u2020 Star \u3011\n\n\u301045\u2020
Notifications \u3011 You must be signed in to change notification settings\n\n  *
\u301044\u2020 Code \u3011\n  * \u301052\u2020 Issues 4 \u3011\n  * \u301053\u2020
Pull requests 0 \u3011\n  * \u301054\u2020 Actions \u3011\n  * \u301055\u2020 Projects
0 \u3011\n  * \u301056\u2020 Security \u3011\n  * \u301057\u2020 Insights \u3011\n\n
Additional navigation options  \n\n  * \u301044\u2020 Code \u3011 \n  * \u301052\u2020
Issues \u3011 \n  * \u301053\u2020 Pull requests \u3011 \n  * \u301054\u2020 Actions
\u3011 \n  * \u301055\u2020 Projects \u3011 \n  * \u301056\u2020 Security \u3011 \n  *
\u301057\u2020 Insights \u3011 \n\n# Joao208/alexa-chatgpt\n\nThis commit does not
belong to any branch on this repository, and may belong to a fork outside of the
repository.\n\n\u00a0main\n\n\u301049\u2020Branches\u3011\u301050\u2020Tags\u3011\n\nGo to file \n\nCode\n\n## Folders and files \n\nName Name \n\nLast commit message
\n\nLast commit date   \n\n## Latest commit\n\n## History\n\n\u301058\u202010
Commits\u3011
\n\n\u301059\u2020.vscode\u3011\n\n\u301059\u2020.vscode\u3011\n\n\u301060\u2020.gitignore\u3011\n\n\u301060\u2020.gitignore\u3011\n\n\u301061\u2020README.md\u3011\n\n\u301061\u2020README.md\u3011\n\n\u301062\u2020index.js\u3011\n\n\u301062\u2020index.js\u3011\n\n\u301063\u2020local-
debugger.js\u3011\n\n\u301063\u2020local-
debugger.js\u3011\n\n\u301064\u2020package.json\u3011\n\n\u301064\u2020package.json\u3
```

011\n\n\u301065\u2020util.js\u3011\n\n\u301065\u2020util.js\u3011\n\n\u301066\u2020yar
n.lock\u3011\n\n\u301066\u2020yarn.lock\u3011\n\nView all files   \n\n## Repository
files navigation \n\n * README \n\n# Alexa Skill with ChatGPT\n\nThis repository
contains an example of how to use OpenAI&#x27;s ChatGPT language model to create an
Alexa Skill.\n\n## How it works\n\nThis code connects to the OpenAI ChatGPT API and
sends Alexa&#x27;s questions to the model, which processes them and returns the
answers. Then, these responses are transmitted back to Alexa and presented to the
user.\n\n## Configuration\n\nYou will need to obtain an API key from OpenAI to use the
ChatGPT model. More information on how to do this can be found in the OpenAI API
documentation.\n\nOnce you have your API key, simply insert it into the index.js file
in the designated location.\n\n## Running the example\n\nThis example was developed
using Amazon&#x27;s ASK CLI. To run it, you will need to install the ASK CLI and
configure your Amazon Developer account.\n\nOnce configured, simply run the following
commands in the terminal:\n\n   ask init\n   ask deploy\n\nThis will deploy your
Skill to your Amazon Developer account and make it available for use with Alexa.\n\n##
Final considerations\n\nThis is just a basic example of how to use ChatGPT with Alexa.
You can expand this implementation to create more complex and personalized skills.
Enjoy!\n\n## About\n\nThis is an Alexa bot developed with the help of OpenAI&#x27;s
ChatGPT language model. With it, you can converse naturally with Alexa, answering
questions and performing tasks based on its language training.
\n\n\u301046\u2020bit.ly/3wUKdIz\u2020bit.ly\u3011 \n\n### Topics\n\n\u301067\u2020
javascript \u3011 \u301068\u2020 chat \u3011 \u301069\u2020 alexa \u3011
\u301070\u2020 axios \u3011 \u301071\u2020 chat-gpt \u3011 \n\n### Resources\n\nReadme
\n\n\u301051\u2020 Activity \u3011 \n\n### Stars\n\n\u301047\u2020 190 stars \u3011
\n\n### Watchers\n\n\u301072\u2020 7 watching \u3011 \n\n### Forks\n\n\u301048\u2020
48 forks \u3011 \n\n\u301073\u2020 Report repository \u3011 \n\n## \u301074\u2020
Releases \u3011\n\nNo releases published\n\n## \u301075\u2020 Packages 0 \u3011\n\nNo
packages published   \n\n## \u301076\u2020 Contributors 2 \u3011\n\n *
\u301043\u2020 \u3011 \u30104", "pub_date": null, "extra": {"cited_message_idx": 89,
"search_result_idx": null, "evidence_text": "source", "cloud_doc_url": null}}}],
"content_references": [{"matched_text": "\u301089\u2020source\u3011", "start_idx": 944,
"end_idx": 955, "alt": null, "type": "webpage_extended", "title": "GitHub -
Joao208/alexa-chatgpt: This is an Alexa bot developed with the help of OpenAI&#x27;s
ChatGPT language model. With it, you can converse naturally with Alexa, answering
questions and performing tasks based on its language training.", "url":
"https://github.com/Joao208/alexa-chatgpt", "pub_date": null, "snippet": "   *
\u301012\u2020 GitHub Skills \u2020skills.github.com\u3011\n   * \u301013\u2020 Blog
\u2020github.blog\u3011\n\n * Solutions \n\nBy size\n   * \u301014\u2020 Enterprise
\u3011\n   * \u301015\u2020 Teams \u3011\n   * \u301016\u2020 Startups \u3011\n\nBy
industry\n   * \u301017\u2020 Healthcare \u3011\n   * \u301018\u2020 Financial
services \u3011\n   * \u301019\u2020 Manufacturing \u3011\n\nBy use case\n   *
\u301020\u2020 CI/CD &amp; Automation \u3011\n   * \u301021\u2020 DevOps \u3011\n
* \u301022\u2020 DevSecOps \u3011\n\n * Resources \n\nTopics\n   * \u301023\u2020 AI
\u3011\n   * \u301024\u2020 DevOps \u3011\n   * \u301025\u2020 Security \u3011\n
* \u301026\u2020 Software Development \u3011\n   * \u301027\u2020 View all
\u3011\n\nExplore\n   * \u301028\u2020 Learning Pathways
\u2020resources.github.com\u3011\n   * \u301029\u2020 White papers, Ebooks, Webinars
\u2020resources.github.com\u3011\n   * \u301030\u2020 Customer Stories \u3011\n   *
\u301031\u2020 Partners \u2020partner.github.com\u3011\n\n * Open Source \n\n   *

```
\u301032\u2020 GitHub Sponsors Fund open source developers \u3011\n\n    *
\u301033\u2020 The ReadME Project GitHub community articles \u3011\n\nRepositories\n
* \u301034\u2020 Topics \u3011\n    * \u301035\u2020 Trending \u3011\n    *
\u301036\u2020 Collections \u3011\n\n  * Enterprise \n\n    * \u301014\u2020
Enterprise platform AI-powered developer platform \u3011\n\nAvailable add-ons\n    *
\u301037\u2020 Advanced Security Enterprise-grade security features \u3011\n    *
\u301038\u2020 GitHub Copilot Enterprise-grade AI features \u3011\n    *
\u301039\u2020 Premium Support Enterprise-grade 24/7 support \u3011\n\n  *
\u301040\u2020Pricing\u3011 \n\nSearch or jump to...\n\n# Search code, repositories,
users, issues, pull requests...\n\n Search \n\nClear  \n\n\u301041\u2020 Search syntax
tips \u2020docs.github.com\u3011 \n\n#  Provide feedback \n\nWe read every piece of
feedback, and take your input very seriously.\n\nInclude my email address so I can be
contacted\n\nCancel  Submit feedback \n\n#  Saved searches \n\n## Use saved searches
to filter your results more quickly\n\nName\n\nQuery\n\nTo see all available
qualifiers, see our \u301041\u2020documentation\u2020docs.github.com\u3011. \n\nCancel
Create saved search \n\n\u30101\u2020 Sign in \u3011 \n\n\u301042\u2020 Sign up \u3011
Reseting focus\n\nYou signed in with another tab or window. Reload to refresh your
session. You signed out in another tab or window. Reload to refresh your session. You
switched accounts on another tab or window. Reload to refresh your session.   Dismiss
alert\n\n{{ message }}\n\n\u301043\u2020 Joao208 \u3011  / \u301044\u2020alexa-
chatgpt\u3011  Public\n\n  * \u301045\u2020 Notifications \u3011 You must be signed in
to change notification settings\n  * \u301045\u2020 Fork 48 \u3011 \n  *
\u301045\u2020 Star 190 \u3011\n\nThis is an Alexa bot developed with the help of
OpenAI&#x27;s ChatGPT language model. With it, you can converse naturally with Alexa,
answering questions and performing tasks based on its language training.
\n\n\u301046\u2020bit.ly/3wukdiz\u2020bit.ly\u3011 \n\n\u301047\u2020 190 stars \u3011
\u301048\u2020 48 forks \u3011 \u301049\u2020 Branches \u3011 \u301050\u2020 Tags
\u3011 \u301051\u2020 Activity \u3011 \n\n\u301045\u2020 Star \u3011\n\n\u301045\u2020
Notifications \u3011 You must be signed in to change notification settings\n\n  *
\u301044\u2020 Code \u3011\n  * \u301052\u2020 Issues 4 \u3011\n  * \u301053\u2020
Pull requests 0 \u3011\n  * \u301054\u2020 Actions \u3011\n  * \u301055\u2020 Projects
0 \u3011\n  * \u301056\u2020 Security \u3011\n  * \u301057\u2020 Insights \u3011\n\n
Additional navigation options  \n\n  * \u301044\u2020 Code \u3011 \n  * \u301052\u2020
Issues \u3011 \n  * \u301053\u2020 Pull requests \u3011 \n  * \u301054\u2020 Actions
\u3011 \n  * \u301055\u2020 Projects \u3011 \n  * \u301056\u2020 Security \u3011 \n  *
\u301057\u2020 Insights \u3011 \n\n# Joao208/alexa-chatgpt\n\nThis commit does not
belong to any branch on this repository, and may belong to a fork outside of the
repository.\n\n\u00a0main\n\n\u301049\u2020Branches\u3011\u301050\u2020Tags\u3011\n\nG
o to file \n\nCode\n\n## Folders and files \n\nName Name \n\nLast commit message
\n\nLast commit date   \n\n## Latest commit\n\n## History\n\n\u301058\u202010
Commits\u3011
\n\n\u301059\u2020.vscode\u3011\n\n\u301059\u2020.vscode\u3011\n\n\u301060\u2020.gitig
nore\u3011\n\n\u301060\u2020.gitignore\u3011\n\n\u301061\u2020README.md\u3011\n\n\u301
061\u2020README.md\u3011\n\n\u301062\u2020index.js\u3011\n\n\u301062\u2020index.js\u30
11\n\n\u301063\u2020local-debugger.js\u3011\n\n\u301063\u2020local-
debugger.js\u3011\n\n\u301064\u2020package.json\u3011\n\n\u301064\u2020package.json\u3
011\n\n\u301065\u2020util.js\u3011\n\n\u301065\u2020util.js\u3011\n\n\u301066\u2020yar
n.lock\u3011\n\n\u301066\u2020yarn.lock\u3011\n\nView all files   \n\n## Repository
files navigation \n\n  * README \n\n# Alexa Skill with ChatGPT\n\nThis repository
```

```
contains an example of how to use OpenAI&#x27;s ChatGPT language model to create an
Alexa Skill.\n\n## How it works\n\nThis code connects to the OpenAI ChatGPT API and
sends Alexa&#x27;s questions to the model, which processes them and returns the
answers. Then, these responses are transmitted back to Alexa and presented to the
user.\n\n## Configuration\n\nYou will need to obtain an API key from OpenAI to use the
ChatGPT model. More information on how to do this can be found in the OpenAI API
documentation.\n\nOnce you have your API key, simply insert it into the index.js file
in the designated location.\n\n## Running the example\n\nThis example was developed
using Amazon&#x27;s ASK CLI. To run it, you will need to install the ASK CLI and
configure your Amazon Developer account.\n\nOnce configured, simply run the following
commands in the terminal:\n\n    ask init\n    ask deploy\n\nThis will deploy your
Skill to your Amazon Developer account and make it available for use with Alexa.\n\n##
Final considerations\n\nThis is just a basic example of how to use ChatGPT with Alexa.
You can expand this implementation to create more complex and personalized skills.
Enjoy!\n\n## About\n\nThis is an Alexa bot developed with the help of OpenAI&#x27;s
ChatGPT language model. With it, you can converse naturally with Alexa, answering
questions and performing tasks based on its language training.
\n\n\u301046\u2020bit.ly/3wUKdIz\u2020bit.ly\u3011 \n\n### Topics\n\n\u301067\u2020
javascript \u3011 \u301068\u2020 chat \u3011 \u301069\u2020 alexa \u3011
\u301070\u2020 axios \u3011 \u301071\u2020 chat-gpt \u3011 \n\n### Resources\n\nReadme
\n\n\u301051\u2020 Activity \u3011 \n\n### Stars\n\n\u301047\u2020 190 stars \u3011
\n\n### Watchers\n\n\u301072\u2020 7 watching \u3011 \n\n### Forks\n\n\u301048\u2020
48 forks \u3011 \n\n\u301073\u2020 Report repository \u3011 \n\n##  \u301074\u2020
Releases \u3011\n\nNo releases published\n\n##  \u301075\u2020 Packages 0 \u3011\n\nNo
packages published   \n\n##  \u301076\u2020 Contributors 2 \u3011\n\n  *
\u301043\u2020 \u3011  \u30104"}], "gizmo_id": null, "message_type": null,
"model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa227ec-dd61-
4e1d-94f8-77c0b3e15a55", "finish_details": {"type": "stop", "stop_tokens": [200002]},
"is_complete": true, "request_id": "8d1fbb634ea6badf-MXP", "timestamp_": "absolute"},
"recipient": "all", "channel": null}, "parent": "aaa227ec-dd61-4e1d-94f8-77c0b3e15a55",
"children": ["aaa28fe7-5d05-40f3-a682-73834e23fbb7"]}, "aaa28fe7-5d05-40f3-a682-
73834e23fbb7": {"id": "aaa28fe7-5d05-40f3-a682-73834e23fbb7", "message": {"id":
"aaa28fe7-5d05-40f3-a682-73834e23fbb7", "author": {"role": "user", "name": null,
"metadata": {}}, "create_time": 1728826815.604684, "update_time": null, "content":
{"content_type": "text", "parts": ["alex offerrs the ability to monetize a skill will
i be acting dishonestly, disreputabled or just kind of shady if I monetize the skill i
create using these instructions? "]}, "status": "finished_successfully", "end_turn":
null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets":
[]}, "request_id": "8d1fbd8a598dbadf-MXP", "message_source": null, "timestamp_":
"absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent":
"423f25ca-d543-4f1b-8c55-c04eda2a2c35", "children": ["cd4f0326-ccdd-48c7-9b98-
c80f59e4fabc"]}, "cd4f0326-ccdd-48c7-9b98-c80f59e4fabc": {"id": "cd4f0326-ccdd-48c7-
9b98-c80f59e4fabc", "message": {"id": "cd4f0326-ccdd-48c7-9b98-c80f59e4fabc", "author":
{"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728826840.278847,
"update_time": null, "content": {"content_type": "text", "parts": ["Monetizing an
Alexa skill that you create using open-source code like the one from **Joao208/alexa-
chatgpt** raises several important considerations, especially from ethical and
reputational perspectives:\n\n### 1. **License Compliance**:  \nMany open-source
projects, including those on GitHub, are released under specific licenses (e.g., MIT,
GPL). These licenses often allow modification and even commercialization, but with
certain conditions. For example:\n   - **MIT License**: This is very permissive and
typically allows for commercial use, provided that you include the original copyright
notice and license in your project.\n   - **GPL**: This requires that if you modify
and distribute the software, you must also open-source your version under the same
license.\n\n   **What to do**: Check the repository&#x27;s license (which is usually
included in the project) to ensure you comply with its terms before monetizing
anything based on it.\n\n### 2. **Attribution**:  \nIn some cases, even if
monetization is allowed, you are required to give proper credit to the original
creator. Failing to do so can be seen as dishonest or shady. Be transparent about the
use of open-source code and clearly state your modifications or contributions if you
are selling or monetizing the skill.\n\n### 3. **Ethical Considerations**:  \nUsing
someone else\u2019s open-source work for commercial gain without adding substantial
value or giving credit can be perceived as taking advantage of the community.
It&#x27;s important to ask yourself:\n   - Are you adding unique value to the product,
or is it just a repackaged version of someone else&#x27;s work?\n   - Would the
original author approve of your monetization efforts?\n\n   If you&#x27;re mostly
using the open-source work without significant changes, it\u2019s more ethical to
either refrain from monetizing or to contribute back to the project (e.g., through
donations, code contributions, or support).\n\n### 4. **User Expectations**:  \nIf you
charge users for a skill, they will expect high reliability, support, and ongoing
updates. If the core functionality relies on a third-party open-source project,
you\u2019ll need to ensure that it\u2019s stable and properly supported. Otherwise,
users may see this as shady or a ripoff if issues arise, especially if they find out
it\u2019s built on open-source software.\n\n### 5. **Reputational Risk**:  \nWhile not
necessarily dishonest, monetizing something that heavily relies on open-source
contributions can give the impression of being opportunistic if not handled
transparently. However, if you build upon the open-source code, add significant
improvements, and provide value beyond what\u2019s already freely available, this risk
is mitigated.\n\n### Final Thought:\nMonetizing your Alexa skill can be done ethically
as long as you:\n   - Follow the license terms.\n   - Give credit where it\u2019s
due.\n   - Provide value beyond what the open-source project offers.\n   - Ensure
users know what they\u2019re paying for.\n\nIf you handle it transparently and add
value, there\u2019s no reason why it would be considered shady. However, failing to
follow these guidelines could hurt your reputation."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa28fe7-5d05-40f3-a682-73834e23fbb7", "request_id": "8d1fbd8a598dbadf-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa28fe7-5d05-40f3-a682-73834e23fbb7", "children": ["aaa218bd-e19c-445f-81f9-
a63bcb1bd4f3"]}, "aaa218bd-e19c-445f-81f9-a63bcb1bd4f3": {"id": "aaa218bd-e19c-445f-
81f9-a63bcb1bd4f3", "message": {"id": "aaa218bd-e19c-445f-81f9-a63bcb1bd4f3", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728827098.563885,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
MAqCAj8zYpzHjqlMrtG9YOfI", "size_bytes": 457587, "width": 1361, "height": 2006,
"fovea": null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image":
```

null, "emu_patches_override": null, "sanitized": true}}, "okay I went to the page but
can you give me a step by step detailed walk through on what to click/do next? "]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"attachments": [{"id": "file-MAqCAj8zYpzHjqlMrtG9YOfI", "size": 457587, "name":
"github.png", "mime_type": "image/png", "width": 1361, "height": 2006}],
"serialization_metadata": {"custom_symbol_offsets": []}, "targeted_reply": "You can
find the repository [here on GitHub](https://github.com/Joao208/alexa-chatgpt)",
"request_id": "8d1fc46d9ebb0e0b-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "cd4f0326-ccdd-
48c7-9b98-c80f59e4fabc", "children": ["c5363c82-8279-467c-9339-629de470092c"]},
"c5363c82-8279-467c-9339-629de470092c": {"id": "c5363c82-8279-467c-9339-629de470092c",
"message": {"id": "c5363c82-8279-467c-9339-629de470092c", "author": {"role": "system",
"name": null, "metadata": {}}, "create_time": 1728827098.564288, "update_time": null,
"content": {"content_type": "text", "parts": [""]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"request_id": "8d1fc46d9ebb0e0b-MXP",
"message_source": null, "timestamp_": "absolute", "message_type": null,
"is_visually_hidden_from_conversation": true}, "recipient": "all", "channel": null},
"parent": "aaa218bd-e19c-445f-81f9-a63bcb1bd4f3", "children": ["5dc3c5fd-70f7-42f2-
9d54-af1c37276f2c"]}, "5dc3c5fd-70f7-42f2-9d54-af1c37276f2c": {"id": "5dc3c5fd-70f7-
42f2-9d54-af1c37276f2c", "message": {"id": "5dc3c5fd-70f7-42f2-9d54-af1c37276f2c",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728827139.176281, "update_time": null, "content": {"content_type": "text", "parts":
["Here\u2019s a detailed walkthrough of how to use the GitHub repository for
&quot;Alexa Skill with ChatGPT&quot; and integrate it with your Alexa skill.\n\n1.
Install ASK CLI (Alexa Skills Kit Command Line Interface):\n   - You\u2019ll need
to install ASK CLI if you haven\u2019t already. The ASK CLI lets you manage your Alexa
skills. \n   - To install ASK CLI, run the following command in your terminal:\n
```bash\n   npm install -g ask-cli\n   ```\n\n2. **Clone the Repository**:\n   -
Go to the GitHub repository page [Joao208/alexa-
chatgpt](https://github.com/Joao208/alexa-chatgpt) and click on the **Code** button.\n
- Copy the URL from the HTTPS or SSH section.\n   - In your terminal or command prompt,
navigate to the directory where you want to store the project, then run the following
command:\n   ```bash\n   git clone https://github.com/Joao208/alexa-chatgpt.git\n
```\n   - This will download the repository onto your local machine.\n\n3. **Open the
Project Folder**:\n   - Navigate to the downloaded project directory:\n     ```bash\n
cd alexa-chatgpt\n     ```\n\n4. **Open the `index.js` File**:\n   - In this project
folder, find the `index.js` file, which contains the core logic for the Alexa skill.\n
- You\u2019ll need to modify this file to include your OpenAI API key.\n   - Open the
file in any text editor (VSCode, Sublime Text, etc.) and look for the section where
you should input your OpenAI API key (usually something like `const openAiApiKey =
&#x27;YOUR_API_KEY&#x27;;`).\n   - Replace `&#x27;YOUR_API_KEY&#x27;` with your actual
OpenAI API key.\n\n5. **Update Dependencies**:\n   - Install the required dependencies
for the project by running:\n     ```bash\n     npm install\n     ```\n\n6.
**Configure ASK CLI**:\n   - Run the following command to initialize ASK CLI:\n
```bash\n     ask init\n     ```\n   - Follow the on-screen prompts to authenticate
and configure your Amazon developer account.\n\n7. **Deploy the Skill**:\n   - Once
everything is set up, you can deploy the skill to your Amazon Developer account by
running:\n     ```bash\n     ask deploy\n     ```\n\n8. **Test the Skill**:\n   -
After deploying, you can test your skill using the Alexa developer console or any

~~~
Alexa-enabled device.\n   - You can also test it locally by using the `local-
debugger.js` file provided in the repository. You\u2019ll need to set up a local
debugging environment using Node.js.\n\n9. **Modify and Expand the Skill
(Optional)**:\n   - The project you\u2019ve cloned is a basic template for Alexa to
interact with ChatGPT. You can modify or expand the skill by adding new features,
personalizing the responses, or even monetizing the skill through Alexa\u2019s in-
skill purchasing.\n\n10. **Monitoring and Logs**:\n    - To ensure that your skill is
working properly, monitor the logs using CloudWatch, which will give you insights into
how the skill is performing or if there are any errors during execution.\n\nOnce
you\u2019ve completed these steps, your Alexa skill integrated with OpenAI\u2019s
ChatGPT will be live and ready to use!\n\nLet me know if you need clarification on any
specific step!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "c5363c82-8279-467c-9339-629de470092c", "request_id": "8d1fc46d9ebb0e0b-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"c5363c82-8279-467c-9339-629de470092c", "children": ["aaa21f71-eb27-4763-bb9c-
8f3683e86e32"]}, "aaa21f71-eb27-4763-bb9c-8f3683e86e32": {"id": "aaa21f71-eb27-4763-
bb9c-8f3683e86e32", "message": {"id": "aaa21f71-eb27-4763-bb9c-8f3683e86e32", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728827468.441283,
"update_time": null, "content": {"content_type": "text", "parts": ["powershell said
this: \nnpm : The term &#x27;npm&#x27; is not recognized as the name of a cmdlet,
function, script file, or operable program. Check the\r\nspelling of the name, or if a
path was included, verify that the path is correct and try again.\r\nAt line:1
char:1\r\n+ npm install -g ask-cli\r\n+ ~~~\r\n    + CategoryInfo          :
ObjectNotFound: (npm:String) [], CommandNotFoundException\r\n    +
FullyQualifiedErrorId : CommandNotFoundException\n\nwhat i&#x27;m doing wrong? "]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d1fcd7b3a4b0e66-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "5dc3c5fd-70f7-
42f2-9d54-af1c37276f2c", "children": ["86aa5c03-83e4-4a71-943c-d5ee7f8c1a99"]},
"86aa5c03-83e4-4a71-943c-d5ee7f8c1a99": {"id": "86aa5c03-83e4-4a71-943c-d5ee7f8c1a99",
"message": {"id": "86aa5c03-83e4-4a71-943c-d5ee7f8c1a99", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728827487.346909,
"update_time": null, "content": {"content_type": "text", "parts": ["The error
you&#x27;re seeing means that PowerShell does not recognize the `npm` command, which
is likely because Node.js is not installed on your system. \n\nHere are the steps to
resolve the issue:\n\n### 1. **Install Node.js and npm:**\n   - Go to the official
Node.js website: [https://nodejs.org](https://nodejs.org).\n   - Download the **LTS
(Long-Term Support)** version for Windows.\n   - Follow the installation instructions
to install Node.js. This will also install **npm** (Node Package Manager)
automatically.\n\n### 2. **Verify the Installation:**\n   - After installation, open
**PowerShell** or **Command Prompt** and run the following commands to verify that
Node.js and npm were installed correctly:\n     ```bash\n     node -v\n     npm -v\n
```\n   - This should return the installed versions of Node.js and npm. If it does,
you&#x27;re ready to proceed.\n\n### 3. **Run the npm Command Again:**\n   - Now that
Node.js and npm are installed, you can proceed with the command that caused the
~~~

error:\n   ```bash\n   npm install -g ask-cli\n   ```\n\nAfter following these steps, your system should recognize the `npm` command, and you should be able to install the ASK CLI successfully."]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa21f71-eb27-4763-bb9c-8f3683e86e32", "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true, "request_id": "8d1fcd7b3a4b0e66-MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa21f71-eb27-4763-bb9c-8f3683e86e32", "children": ["aaa2697d-2405-4bfe-9b8f-1078e45ed046"]}, "aaa2697d-2405-4bfe-9b8f-1078e45ed046": {"id": "aaa2697d-2405-4bfe-9b8f-1078e45ed046", "message": {"id": "aaa2697d-2405-4bfe-9b8f-1078e45ed046", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728827590.10812, "update_time": null, "content": {"content_type": "text", "parts": ["but whats chocolatey? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id": "8d1fd0720bf2bb13-MXP", "message_source": null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent": "86aa5c03-83e4-4a71-943c-d5ee7f8c1a99", "children": ["d104266b-90ef-451a-9bfd-96d0274db5de"]}, "d104266b-90ef-451a-9bfd-96d0274db5de": {"id": "d104266b-90ef-451a-9bfd-96d0274db5de", "message": {"id": "d104266b-90ef-451a-9bfd-96d0274db5de", "author": {"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728827603.583597, "update_time": null, "content": {"content_type": "text", "parts": ["**Chocolatey** is a package manager for Windows that simplifies the installation, updating, and management of software. It\u2019s similar to package managers like `apt` on Linux or `Homebrew` on macOS. Chocolatey automates the process of downloading, installing, and configuring software on Windows systems by using simple commands.\n\n### Why Use Chocolatey?\n- **Automated Installations:** Chocolatey allows you to install software with a single command, saving you time from manually downloading and running installers.\n- **Easily Manage Packages:** You can update or remove software in bulk.\n- **Wide Range of Software:** Chocolatey supports many popular programs, libraries, and tools, including development environments like Node.js, Docker, and Visual Studio Code.\n\n### Example of Installing Node.js with Chocolatey:\nOnce you have Chocolatey installed, you can install Node.js (and npm, which comes with it) by running:\n```powershell\nchoco install nodejs\n```\n\n### How to Install Chocolatey:\n1. Open PowerShell as an administrator.\n2. Run the following command to install Chocolatey:\n   ```powershell\n   Set-ExecutionPolicy Bypass -Scope Process -Force; ["]}, "status": "in_progress", "end_turn": null, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa2697d-2405-4bfe-9b8f-1078e45ed046", "finish_details": {"type": "interrupted"}, "request_id": "8d1fd0720bf2bb13-MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa2697d-2405-4bfe-9b8f-1078e45ed046", "children": ["aaa2ab94-25c8-4445-9aa8-cb66c9711bc1"]}, "aaa2ab94-25c8-4445-9aa8-cb66c9711bc1": {"id": "aaa2ab94-25c8-4445-9aa8-cb66c9711bc1", "message": {"id": "aaa2ab94-25c8-4445-9aa8-cb66c9711bc1", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728827615.415293, "update_time": null, "content": {"content_type": "multimodal_text", "parts": [{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-HD7IDcW1Sq6FX8Im4OpmJD3o", "size_bytes": 457587, "width": 1361,

"height": 2006, "fovea": null, "metadata": {"dalle": null, "gizmo": null,
"emu_omit_glimpse_image": null, "emu_patches_override": null, "sanitized": true}},
"but whats Chocolatey? "]}, "status": "finished_successfully", "end_turn": null,
"weight": 1.0, "metadata": {"attachments": [{"id": "file-HD7IDcWlSq6FX8Im4OpmJD3o",
"size": 457587, "name": "github.png", "mime_type": "image/png", "width": 1361,
"height": 2006}], "serialization_metadata": {"custom_symbol_offsets": []},
"request_id": "8d1fd10cfea183a0-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "d104266b-90ef-451a-9bfd-96d0274db5de", "children": ["45e131f1-2a1f-42c7-91dc-c58ec8d72a92"]},
"45e131f1-2a1f-42c7-91dc-c58ec8d72a92": {"id": "45e131f1-2a1f-42c7-91dc-c58ec8d72a92",
"message": {"id": "45e131f1-2a1f-42c7-91dc-c58ec8d72a92", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728827626.135143,
"update_time": null, "content": {"content_type": "text", "parts": ["Chocolatey is a
package manager designed specifically for Windows that simplifies installing, updating,
and managing software on your machine. It&#x27;s very similar to package managers like
`apt` on Linux or `brew` on macOS.\n\n### Key Points about Chocolatey:\n1. **Automated
Software Management**: With Chocolatey, you can install applications and development
tools with a single command in PowerShell or Command"]}, "status": "in_progress",
"end_turn": null, "weight": 1.0, "metadata": {"citations": [], "content_references":
[], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o",
"default_model_slug": "gpt-4o", "parent_id": "aaa2ab94-25c8-4445-9aa8-cb66c9711bc1",
"finish_details": {"type": "interrupted"}, "request_id": "8d1fd10cfea183a0-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa2ab94-25c8-4445-9aa8-cb66c9711bc1", "children": ["aaa28f60-59b4-48db-83d5-ed838d8eefa7"]},
"aaa28f60-59b4-48db-83d5-ed838d8eefa7": {"id": "aaa28f60-59b4-48db-83d5-ed838d8eefa7",
"message": {"id": "aaa28f60-59b4-48db-83d5-ed838d8eefa7", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728827658.919746, "update_time": null,
"content": {"content_type": "multimodal_text", "parts": [{"content_type":
"image_asset_pointer", "asset_pointer": "file-service://file-2CkRiEZy6TXnzmi2iD5vKZa2",
"size_bytes": 32995, "width": 511, "height": 393, "fovea": null, "metadata": {"dalle":
null, "gizmo": null, "emu_omit_glimpse_image": null, "emu_patches_override": null,
"sanitized": true}}, "but whats chocolatey? "]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"attachments": [{"id": "file-2CkRiEZy6TXnzmi2iD5vKZa2", "size": 32995, "name": "Screenshot 2024-10-13 155254.png",
"mime_type": "image/png", "width": 511, "height": 393}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d1fd21d7d6a4c52-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "45e131f1-2a1f-42c7-91dc-c58ec8d72a92", "children": ["8c130c93-99da-4cf1-a1ab-3ba9ef1ac5ff"]}, "8c130c93-99da-4cf1-a1ab-3ba9ef1ac5ff": {"id": "8c130c93-99da-4cf1-a1ab-3ba9ef1ac5ff", "message": {"id": "8c130c93-99da-4cf1-a1ab-3ba9ef1ac5ff",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728827676.529696, "update_time": null, "content": {"content_type": "text", "parts":
["The screenshot you shared is from the **Node.js** setup, which is asking whether you
want to install the necessary tools to compile native modules when using npm (Node
Package Manager). One of those tools is **Chocolatey**, a package manager for
Windows.\n\nHere&#x27;s a quick summary of **Chocolatey**:\n- **Chocolatey** is a
software management solution for Windows, automating the process of installing,
updating, and managing software packages. It&#x27;s similar to Linux package managers
like `apt` or `yum`.\n- **Why Chocolatey?** It simplifies installing not just software

but also system libraries and development tools with one command.\n- **Integration with Node.js**: As shown in your screenshot, Node.js uses Chocolatey to install additional required tools, like Python and Visual Studio Build Tools, which are necessary for compiling native C/C++ modules in npm packages.\n\nBy checking the box and proceeding, it will automatically install Chocolatey, along with the tools necessary for compiling native modules. You can either use Chocolatey afterward to manage other software or ignore it for now, as it\u2019s primarily used here to support Node.js installations."]}, "status": "finished_successfully", "end_turn": true , "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa28f60-59b4-48db-83d5-ed838d8eefa7", "request_id": "8d1fd21d7d6a4c52-MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa28f60-59b4-48db-83d5-ed838d8eefa7", "children": ["aaa298b8-fc5a-4afd-b8f8-78130ec8c174"]}, "aaa298b8-fc5a-4afd-b8f8-78130ec8c174": {"id": "aaa298b8-fc5a-4afd-b8f8-78130ec8c174", "message": {"id": "aaa298b8-fc5a-4afd-b8f8-78130ec8c174", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728846609.746973, "update_time": null, "content": {"content_type": "text", "parts": ["I got to here and then so yeah: \n\nWindows PowerShell\r\nCopyright (C) Microsoft Corporation. All rights reserved.\r\n\r\nInstall the latest PowerShell for new features and improvements! https://aka.ms/PSWindows\r\n\r\nPS C:\\Users\\alexi&gt; npm install -g ask-cli\r\nnpm : The term &#x27;npm&#x27; is not recognized as the name of a cmdlet, function, script file, or operable program. Check the\r\nspelling of the name, or if a path was included, verify that the path is correct and try again.\r\nAt line:1 char:1\r\n+ npm install -g ask-cli\r\n+ ~~~\r\n   + CategoryInfo   : ObjectNotFound: (npm:String) [], CommandNotFoundException\r\n   + FullyQualifiedErrorId : CommandNotFoundException\r\n\r\nPS C:\\Users\\alexi&gt; npm install -g ask-cli\r\nnpm : The term &#x27;npm&#x27; is not recognized as the name of a cmdlet, function, script file, or operable program. Check the\r\nspelling of the name, or if a path was included, verify that the path is correct and try again.\r\nAt line:1 char:1\r\n+ npm install -g ask-cli\r\n+ ~~~\r\n   + CategoryInfo   : ObjectNotFound: (npm:String) [], CommandNotFoundException\r\n   + FullyQualifiedErrorId : CommandNotFoundException\r\n\r\nPS C:\\Users\\alexi&gt;\r\n&gt;&gt; node -v\r\nnode : The term &#x27;node&#x27; is not recognized as the name of a cmdlet, function, script file, or operable program. Check\r\nthe spelling of the name, or if a path was included, verify that the path is correct and try again.\r\nAt line:2 char:1\r\n+ node -v\r\n+ ~~~~\r\n   + CategoryInfo   : ObjectNotFound: (node:String) [], CommandNotFoundException\r\n   + FullyQualifiedErrorId : CommandNotFoundException\r\n\r\nPS C:\\Users\\alexi&gt; npm -v\r\nnpm : The term &#x27;npm&#x27; is not recognized as the name of a cmdlet, function, script file, or operable program. Check the\r\nspelling of the name, or if a path was included, verify that the path is correct and try again.\r\nAt line:1 char:1\r\n+ npm -v\r\n+ ~~~\r\n   + CategoryInfo   : ObjectNotFound: (npm:String) [], CommandNotFoundException\r\n   + FullyQualifiedErrorId : CommandNotFoundException\r\n\r\nPS C:\\Users\\alexi&gt;\n\n\nwhat to do chat g? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "targeted_reply": "### 2\\. **Verify the Installation:**\n\n*   After installation, open **PowerShell** or **Command Prompt** and run the following commands to verify that Node.js and npm were

installed correctly:\n   \n   bash\n   \n   Copy code\n   \n   `node -v`",
"request_id": "8d21a0cc09d2bb1d-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "8c130c93-99da-
4cf1-a1ab-3ba9ef1ac5ff", "children": ["de6994d2-c25d-47e5-b888-67d54a3284ae"]},
"de6994d2-c25d-47e5-b888-67d54a3284ae": {"id": "de6994d2-c25d-47e5-b888-67d54a3284ae",
"message": {"id": "de6994d2-c25d-47e5-b888-67d54a3284ae", "author": {"role": "system",
"name": null, "metadata": {}}, "create_time": 1728846609.74773, "update_time": null,
"content": {"content_type": "text", "parts": [""]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"request_id": "8d21a0cc09d2bb1d-MXP",
"message_source": null, "timestamp_": "absolute", "message_type": null,
"is_visually_hidden_from_conversation": true}, "recipient": "all", "channel": null},
"parent": "aaa298b8-fc5a-4afd-b8f8-78130ec8c174", "children": ["18da55a6-8133-4348-
99eb-f61b6ca88eb8"]}, "18da55a6-8133-4348-99eb-f61b6ca88eb8": {"id": "18da55a6-8133-
4348-99eb-f61b6ca88eb8", "message": {"id": "18da55a6-8133-4348-99eb-f61b6ca88eb8",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728846639.271873, "update_time": null, "content": {"content_type": "text", "parts":
["It looks like Node.js and npm were not installed successfully, or your system might
not be recognizing the commands yet. Let&#x27;s troubleshoot and fix this step-by-
step:\n\n### 1. Ensure Node.js is Installed\nIf you completed the Node.js installation
(where you saw the &quot;Tools for Native Modules&quot; window with the checkbox to
install tools like Chocolatey), Node.js and npm should be installed on your system.
However, the command might not work due to the PATH not being updated.\n\n1. **Verify
Installation Directory**:\n   - Open File Explorer and navigate to `C:\\Program
Files\\nodejs`. If you can see `node.exe` and `npm.cmd` in this folder, it means
Node.js was installed but isn&#x27;t recognized yet.\n\n2. **Add Node.js to PATH**:\n
- If Node.js is installed but the command isn\u2019t recognized, it could be due to
the PATH variable not being set up correctly.\n   - To add Node.js and npm to your
system\u2019s PATH:\n   - Right-click on **This PC** or **My Computer**, and select
Properties.\n   - Click **Advanced System Settings** on the left panel.\n
- Under the **Advanced** tab, click **Environment Variables**.\n   - Under **System
Variables**, find the `Path` variable, select it, and click **Edit**.\n   - Click
New and add this path: `C:\\Program Files\\nodejs\\`.\n   - Click **OK** to
close the windows.\n   \n3. **Restart PowerShell or Command Prompt** and test if the
commands work:\n   - Run `node -v` to check the Node.js version.\n   - Run `npm -v`
to check the npm version.\n\n### 2. Reinstall Node.js (if necessary)\nIf Node.js is
not present in the expected folder or the PATH adjustments didn&#x27;t fix the issue,
try reinstalling Node.js.\n\n- Download Node.js again from the official website:
[Node.js Download](https://nodejs.org/).\n- Follow the installation process and make
sure the checkbox for installing npm and the necessary tools (like Chocolatey) is
checked.\n- Restart PowerShell or Command Prompt and try running `node -v` and `npm -
v` again.\n\n### 3. Continue with ASK-CLI\nOnce `node -v` and `npm -v` return the
correct version numbers, you can proceed with installing the ASK CLI by
running:\n\n```bash\nnpm install -g ask-cli\n```\n\nLet me know if any of these steps
still present issues, and we can troubleshoot further!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "de6994d2-c25d-47e5-b888-67d54a3284ae", "request_id": "8d21a0cc09d2bb1d-

MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "de6994d2-c25d-47e5-b888-67d54a3284ae", "children": ["aaa24ad4-33c2-4c2e-bbb6-0775ce7fc485"]}, "aaa24ad4-33c2-4c2e-bbb6-0775ce7fc485": {"id": "aaa24ad4-33c2-4c2e-bbb6-0775ce7fc485", "message": {"id": "aaa24ad4-33c2-4c2e-bbb6-0775ce7fc485", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728846794.390896, "update_time": null, "content": {"content_type": "multimodal_text", "parts": [{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-JFiZYlpw4wUrkPXQVVCTc1RI", "size_bytes": 34649, "width": 853, "height": 482, "fovea": null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null, "emu_patches_override": null, "sanitized": true}}, "okay now what? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments": [{"id": "file-JFiZYlpw4wUrkPXQVVCTc1RI", "size": 34649, "name": "Screenshot 2024-10-13 211248.png", "mime_type": "image/png", "width": 853, "height": 482}], "serialization_metadata": {"custom_symbol_offsets": []}, "request_id": "8d21a5489edbbb1d-MXP", "message_source": null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent": "18da55a6-8133-4348-99eb-f61b6ca88eb8", "children": ["b667ca24-4afc-402f-93fd-828353351412"]}, "b667ca24-4afc-402f-93fd-828353351412": {"id": "b667ca24-4afc-402f-93fd-828353351412", "message": {"id": "b667ca24-4afc-402f-93fd-828353351412", "author": {"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728846810.03761, "update_time": null, "content": {"content_type": "text", "parts": ["You&#x27;re getting closer! Since the `npm` files are present, it seems like Node.js was installed, but the `npm` command is not recognized in PowerShell. This usually happens if the Node.js path hasn&#x27;t been added to the system&#x27;s environment variables.\n\nHere&#x27;s what to do next:\n\n1. **Add Node.js to the System Path:**\n - Press `Windows + S` and type **Environment Variables**, then select **Edit the system environment variables**.\n   - In the System Properties window, click **Environment Variables** at the bottom.\n   - Under **System variables**, scroll down and find the **Path** variable, then click **Edit**.\n   - Click **New** and add the path to your Node.js installation. Based on your screenshot, it should be:\n   ```\n C:\\Program Files\\nodejs\\\n   ```\n   - Click **OK** to save and close all windows.\n\n2. **Verify Installation:**\n   - Open a new PowerShell window (important to open a new one to apply changes) and run:\n   ```bash\n   node -v\n   ```\n and\n   ```bash\n   npm -v\n   ```\n\n   If both commands return version numbers, you&#x27;re good to go!\n\n3. **Proceed with ASK CLI Installation:**\n   - Now that Node.js is set up, run the following to install the ASK CLI globally:\n   ```bash\n npm install -g ask-cli\n   ```\n\nLet me know if you run into any more issues!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa24ad4-33c2-4c2e-bbb6-0775ce7fc485", "request_id": "8d21a5489edbbb1d-MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent": "aaa24ad4-33c2-4c2e-bbb6-0775ce7fc485", "children": ["aaa2a185-daa4-47a0-948b-1b6ee4cad702"]}, "aaa2a185-daa4-47a0-948b-1b6ee4cad702": {"id": "aaa2a185-daa4-47a0-948b-1b6ee4cad702", "message": {"id": "aaa2a185-daa4-47a0-948b-1b6ee4cad702", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728846850.973942, "update_time": null, "content": {"content_type": "text", "parts": ["uhh so am i risking corrupting my system by changing these variables? "]}, "status":

```
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21a6afdd93bb1d-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "b667ca24-4afc-
402f-93fd-828353351412", "children": ["b4a9861b-530c-41ce-9cd8-384ccc44726f"]},
"b4a9861b-530c-41ce-9cd8-384ccc44726f": {"id": "b4a9861b-530c-41ce-9cd8-384ccc44726f",
"message": {"id": "b4a9861b-530c-41ce-9cd8-384ccc44726f", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728846867.519028,
"update_time": null, "content": {"content_type": "text", "parts": ["Modifying
environment variables, especially adding a path to the `Path` variable, is a common
and safe operation, as long as it&#x27;s done carefully. It won\u2019t corrupt your
system as long as you:\n\n1. **Do not remove existing paths:** Just add a new entry
without deleting or altering the other ones. You are simply telling the system where
to find additional programs.\n\n2. **Double-check the path you&#x27;re adding:** Make
sure the path is correct. Based on your screenshot, it should be:\n   ```\n
C:\\Program Files\\nodejs\\\n   ```\n\nHere are some key points:\n- **What is the
`Path` variable?** It tells your system where to find executable programs like `node`
and `npm`. By adding Node.js to the `Path`, you&#x27;re letting PowerShell or other
terminals know where to look when you type commands like `npm`.\n  \n- **Why is it
safe?** As long as you\u2019re just adding a path and not removing anything, you
won\u2019t damage the system\u2019s ability to locate other programs.\n\nJust be sure
not to delete any existing entries when you add the new path. If you follow the steps
correctly, the only thing you\u2019ll do is enable PowerShell to find Node.js and
`npm`."]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2a185-daa4-47a0-948b-1b6ee4cad702", "request_id": "8d21a6afdd93bb1d-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2a185-daa4-47a0-948b-1b6ee4cad702", "children": ["aaa2288b-4cd0-42c2-aed4-
35d0a9d68cdd"]}, "aaa2288b-4cd0-42c2-aed4-35d0a9d68cdd": {"id": "aaa2288b-4cd0-42c2-
aed4-35d0a9d68cdd", "message": {"id": "aaa2288b-4cd0-42c2-aed4-35d0a9d68cdd", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848089.044703,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
2HtQSCyS23jPmwUBCVXiuSjF", "size_bytes": 80145, "width": 931, "height": 616, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, {"content_type":
"image_asset_pointer", "asset_pointer": "file-service://file-DTmy4hczCy0fkHjhhzDSj5a3",
"size_bytes": 39541, "width": 641, "height": 581, "fovea": null, "metadata": {"dalle":
null, "gizmo": null, "emu_omit_glimpse_image": null, "emu_patches_override": null,
"sanitized": true}}, "its already there. should i just jump to verifying installation?
"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"attachments": [{"id": "file-2HtQSCyS23jPmwUBCVXiuSjF", "size": 80145, "name":
"Screenshot 2024-10-13 213351.png", "mime_type": "image/png", "width": 931, "height":
616}, {"id": "file-DTmy4hczCy0fkHjhhzDSj5a3", "size": 39541, "name": "Screenshot 2024-
10-13 213334.png", "mime_type": "image/png", "width": 641, "height": 581}],
"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21c4e56f710e0d-MXP", "message_source": null, "timestamp_": "absolute",
```

"message_type": null}, "recipient": "all", "channel": null}, "parent": "b4a9861b-530c-41ce-9cd8-384ccc44726f", "children": ["743b4285-214e-4b5b-99c9-db9b864aa690"]}, "743b4285-214e-4b5b-99c9-db9b864aa690": {"id": "743b4285-214e-4b5b-99c9-db9b864aa690", "message": {"id": "743b4285-214e-4b5b-99c9-db9b864aa690", "author": {"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728848105.614919, "update_time": null, "content": {"content_type": "text", "parts": ["Yes, based on the screenshots you&#x27;ve provided, Node.js is already present in your system&#x27;s environment variables. This means that the paths for Node.js and npm are set up properly. However, since the commands are still not recognized, you should try verifying the installation to see if Node.js and npm are working correctly.\n\n### Steps to verify installation:\n\n1. **Open a new PowerShell or Command Prompt window** (this is important as changes to environment variables only take effect in new sessions).\n   \n2. **Run the following commands**:\n   - `node -v`: This should return the installed version of Node.js (for example, `v16.20.0`).\n   - `npm -v`: This should return the installed version of npm (for example, `7.19.1`).\n\nIf both commands return versions, it means Node.js and npm are installed correctly.\n\nIf either command is not recognized, try restarting your computer to ensure the environment variable changes take full effect, and repeat the above steps.\n\nLet me know how it goes!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa2288b-4cd0-42c2-aed4-35d0a9d68cdd", "request_id": "8d21c4e56f710e0d-MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent": "aaa2288b-4cd0-42c2-aed4-35d0a9d68cdd", "children": ["aaa2ddc1-bfd8-4b98-8c36-5795d2b3b160"]}, "aaa2ddc1-bfd8-4b98-8c36-5795d2b3b160": {"id": "aaa2ddc1-bfd8-4b98-8c36-5795d2b3b160", "message": {"id": "aaa2ddc1-bfd8-4b98-8c36-5795d2b3b160", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728848219.653059, "update_time": null, "content": {"content_type": "text", "parts": ["okay we the recognition phase Chat G!!! what we gon do next?? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id": "8d21c81a597a0e0d-MXP", "message_source": null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent": "743b4285-214e-4b5b-99c9-db9b864aa690", "children": ["77e8c3bb-8606-4b13-91d7-c0eab2dc63c8"]}, "77e8c3bb-8606-4b13-91d7-c0eab2dc63c8": {"id": "77e8c3bb-8606-4b13-91d7-c0eab2dc63c8", "message": {"id": "77e8c3bb-8606-4b13-91d7-c0eab2dc63c8", "author": {"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728848244.831994, "update_time": null, "content": {"content_type": "text", "parts": ["Awesome! Now that Node.js and npm are recognized, let&#x27;s move to the next steps: installing the ASK CLI (Alexa Skills Kit Command Line Interface) and deploying the Alexa skill from the GitHub project you referenced earlier.\n\n### Next Steps:\n\n1. **Install ASK CLI**:\n   Run the following command in PowerShell or Command Prompt:\n   ```\n   npm install -g ask-cli\n   ```\n   This installs the ASK CLI globally on your machine.\n\n2. **Set Up Your Alexa Developer Account**:\n   - If you haven&#x27;t already, [create an Amazon Developer account](https://developer.amazon.com/).\n   - Once you have your account, you&#x27;ll use it to link the ASK CLI to your Alexa Developer console.\n\n3. **Initialize ASK CLI**:\n   After installing the ASK CLI, you need to initialize it. Run this command:\n   ```\n   ask init\n   ```\n   This will walk you through the

steps of linking your ASK CLI to your Alexa Developer account. Make sure to follow the
instructions on the screen, which will ask you to log in to your Amazon Developer
account.\n\n4. **Download or Clone the GitHub Repository**:\n   - Clone the repository
that contains the Alexa skill (the one you were looking at on GitHub). If you
don\u2019t already have it, run:\n   ```\n   git clone
https://github.com/Joao208/alexa-chatgpt.git\n   ```\n   This will download the
project to your local machine.\n\n5. **Navigate to the Project Folder**:\n   Once the
repository is cloned, navigate to the folder in PowerShell or Command Prompt:\n
```\n   cd path/to/alexa-chatgpt\n   ```\n\n6. **Install Dependencies**:\n   In the
project directory, install the required dependencies by running:\n   ```\n   npm
install\n   ```\n\n7. **Deploy the Skill**:\n   Now, you can deploy the skill to your
Alexa Developer account by running:\n   ```\n   ask deploy\n   ```\n\nThis command
will package and deploy the skill to your developer account.\n\n---\n\nAfter
deployment, the skill should appear in your Alexa Developer Console, and you can start
testing it.\n\nLet me know how these steps go, and we&#x27;ll tackle anything else
that comes up!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2ddc1-bfd8-4b98-8c36-5795d2b3b160", "request_id": "8d21c81a597a0e0d-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2ddc1-bfd8-4b98-8c36-5795d2b3b160", "children": ["aaa285de-2249-4634-ba63-
201f5c0d9498"]}, "aaa285de-2249-4634-ba63-201f5c0d9498": {"id": "aaa285de-2249-4634-
ba63-201f5c0d9498", "message": {"id": "aaa285de-2249-4634-ba63-201f5c0d9498", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848331.970828,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
Lm6jeC6V2G5GOhuqVqs9xFAF", "size_bytes": 52861, "width": 1103, "height": 585, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "are we malfunctioning? "]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"attachments": [{"id": "file-Lm6jeC6V2G5GOhuqVqs9xFAF", "size": 52861, "name":
"image.png", "mime_type": "image/png", "width": 1103, "height": 585}],
"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21cad32d5f0e0d-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "77e8c3bb-8606-
4b13-91d7-c0eab2dc63c8", "children": ["cfefc063-4ceb-4818-b901-18a8cebbb8cd"]},
"cfefc063-4ceb-4818-b901-18a8cebbb8cd": {"id": "cfefc063-4ceb-4818-b901-18a8cebbb8cd",
"message": {"id": "cfefc063-4ceb-4818-b901-18a8cebbb8cd", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728848346.944771,
"update_time": null, "content": {"content_type": "text", "parts": ["No, you\u2019re
not malfunctioning! What you see here are warnings, not errors. The installation is
giving you some heads up about deprecated packages (older versions of certain packages
that will no longer be supported in the future). These warnings do not stop the
process from continuing unless it explicitly throws an error that halts the
installation.\n\n### What to do next:\n1. **Wait until the installation completes.**
The warnings don&#x27;t interrupt the process.\n2. **Check if the `ask-cli` command
works** by running the following command:\n   ```bash\n   ask --version\n   ```\n   If
it returns the version number of the `ask-cli` tool, then you&#x27;re good to go and

the installation was successful.\n\nOnce installed, you can proceed with setting up
and deploying your Alexa skill as per the instructions in the GitHub
repository.\n\nLet me know if you encounter any further errors!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa285de-2249-4634-ba63-201f5c0d9498", "request_id": "8d21cad32d5f0e0d-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa285de-2249-4634-ba63-201f5c0d9498", "children": ["aaa207c1-74e5-49c5-a2ed-
f0fe48dbb648"]}, "aaa207c1-74e5-49c5-a2ed-f0fe48dbb648": {"id": "aaa207c1-74e5-49c5-
a2ed-f0fe48dbb648", "message": {"id": "aaa207c1-74e5-49c5-a2ed-f0fe48dbb648", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848455.063893,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
ZTCLhQ4yDxUpke7ZQfm9B7EF", "size_bytes": 75031, "width": 1116, "height": 646, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "we here. gonna use this: ask --
version \n\nand tell you what happen"]}, "status": "finished_successfully", "end_turn":
null, "weight": 1.0, "metadata": {"attachments": [{"id": "file-
ZTCLhQ4yDxUpke7ZQfm9B7EF", "size": 75031, "name": "image.png", "mime_type":
"image/png", "width": 1116, "height": 646}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d21cdd3ebc40e0d-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "cfefc063-4ceb-4818-b901-18a8cebbb8cd", "children": ["1190ff46-f0d6-
4b0f-b24a-dddd2d1eee64"]}, "1190ff46-f0d6-4b0f-b24a-dddd2d1eee64": {"id": "1190ff46-
f0d6-4b0f-b24a-dddd2d1eee64", "message": {"id": "1190ff46-f0d6-4b0f-b24a-dddd2d1eee64",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728848477.152329, "update_time": null, "content": {"content_type": "text", "parts":
["Great progress! Now that you&#x27;ve installed the ASK CLI, running the following
command `ask --version` will confirm that the ASK CLI is set up correctly. If
everything works well, you should see the version of the ASK CLI printed out.\n\nAfter
verifying that the ASK CLI is installed, the next steps are to configure and deploy
your skill using the instructions from the GitHub repository:\n\n1. **Configure ASK
CLI**:\n   - In the terminal, run the following command:\n   ```bash\n   ask
configure\n   ```\n   This command will guide you through setting up your Amazon
Developer account with the ASK CLI. You\u2019ll be asked to log in to your Amazon
Developer account and authorize the CLI to access it.\n\n2. **Initialize your
project**:\n   - Navigate to the folder where your Alexa skill project files are
located (the one cloned from GitHub).\n   - In the terminal, run:\n   ```bash\n
ask init\n   ```\n   This initializes the ASK CLI for your project.\n\n3. **Deploy
your skill**:\n   - Once initialized, deploy your skill with:\n   ```bash\n   ask
deploy\n   ```\n   This command will deploy the Alexa skill and any associated
Lambda function to your Amazon Developer account.\n\n4. **Test your skill**:\n   -
After deployment, you can test the skill on your Alexa device or by using the Alexa
Developer Console.\n\nIf you encounter any errors or issues along the way, let me know
and I can help troubleshoot!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,

"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa207c1-74e5-49c5-a2ed-f0fe48dbb648", "request_id": "8d21cdd3ebc40e0d-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa207c1-74e5-49c5-a2ed-f0fe48dbb648", "children": ["aaa23624-7909-400f-9054-
1c6f7e3846c5"]}, "aaa23624-7909-400f-9054-1c6f7e3846c5": {"id": "aaa23624-7909-400f-
9054-1c6f7e3846c5", "message": {"id": "aaa23624-7909-400f-9054-1c6f7e3846c5", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848542.640594,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
prXrJ6L1ZkSgDuOKoCRlcfxk", "size_bytes": 98973, "width": 1142, "height": 653, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "mayday mayday issue!!! what to do
wise sage software development HAL?"]}, "status": "finished_successfully", "end_turn":
null, "weight": 1.0, "metadata": {"attachments": [{"id": "file-
prXrJ6L1ZkSgDuOKoCRlcfxk", "size": 98973, "name": "Screenshot 2024-10-13 214129.png",
"mime_type": "image/png", "width": 1142, "height": 653}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d21cff7ab7eba8c-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "1190ff46-f0d6-4b0f-b24a-dddd2d1eee64", "children": ["2fac1eb1-f122-
4ab9-80ec-e61d27390ac1"]}, "2fac1eb1-f122-4ab9-80ec-e61d27390ac1": {"id": "2fac1eb1-
f122-4ab9-80ec-e61d27390ac1", "message": {"id": "2fac1eb1-f122-4ab9-80ec-e61d27390ac1",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728848566.315513, "update_time": null, "content": {"content_type": "text", "parts":
["It looks like you&#x27;re encountering an execution policy issue in PowerShell,
which is blocking the script from running. PowerShell has an execution policy that
controls the ability to run scripts and can sometimes block the execution of commands
like `ask --version`.\n\nHere\u2019s how you can fix this:\n\n1. **Open PowerShell as
Administrator**: \n   - Right-click the **Start** menu and select **Windows PowerShell
(Admin)** to open an elevated PowerShell window.\n\n2. **Check Current Execution
Policy**:\n   - Run this command to check the current execution policy:\n
```bash\n   Get-ExecutionPolicy\n   ```\n   - If it returns `Restricted`, it means
scripts are blocked.\n\n3. **Set the Execution Policy**:\n   - To allow the execution
of scripts, run the following command:\n   ```bash\n   Set-ExecutionPolicy
RemoteSigned -Scope CurrentUser\n   ```\n   - This command sets the execution policy
to `RemoteSigned`, which allows scripts to run locally (i.e., scripts created on your
computer). Any scripts downloaded from the internet will need to be signed by a
trusted publisher.\n\n4. **Confirm the Policy Change**:\n   - When prompted, type `Y`
to confirm the change.\n\n5. **Try Running the `ask` Command Again**:\n   - Once
you&#x27;ve set the execution policy, try running the `ask --version` command again to
check if it works.\n\nIf this process completes successfully, you should be able to
run your Alexa Skill CLI commands without further issues.\n\nLet me know how it
goes!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa23624-7909-400f-9054-1c6f7e3846c5", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d21cff7ab7eba8c-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa23624-
7909-400f-9054-1c6f7e3846c5", "children": ["aaa23dbe-fefe-4cde-80b9-02f68f2e9eed"]},
"aaa23dbe-fefe-4cde-80b9-02f68f2e9eed": {"id": "aaa23dbe-fefe-4cde-80b9-02f68f2e9eed",

"message": {"id": "aaa23dbe-fefe-4cde-80b9-02f68f2e9eed", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728848620.124639, "update_time": null,
"content": {"content_type": "text", "parts": ["should i close the powershell admin i
ha ve open now and get new one or just leave open and get new one or just keep using
that one? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21d1e16f220d80-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "2fac1eb1-f122-
4ab9-80ec-e61d27390ac1", "children": ["fa48a162-47ae-446c-8508-708082762783"]},
"fa48a162-47ae-446c-8508-708082762783": {"id": "fa48a162-47ae-446c-8508-708082762783",
"message": {"id": "fa48a162-47ae-446c-8508-708082762783", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728848629.744506,
"update_time": null, "content": {"content_type": "text", "parts": ["You can keep using
the **PowerShell (Admin)** window that you already have open. There&#x27;s no need to
close it and open a new one. Just continue by running the steps I mentioned for
changing the execution policy. Once the execution policy is updated, you should be
able to run your `ask` commands without issue in that same window.\n\nLet me know how
it goes!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa23dbe-fefe-4cde-80b9-02f68f2e9eed", "request_id": "8d21d1e16f220d80-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa23dbe-fefe-4cde-80b9-02f68f2e9eed", "children": ["aaa270c5-f0fd-45f5-8cb4-
2e57ea2c6b13"]}, "aaa270c5-f0fd-45f5-8cb4-2e57ea2c6b13": {"id": "aaa270c5-f0fd-45f5-
8cb4-2e57ea2c6b13", "message": {"id": "aaa270c5-f0fd-45f5-8cb4-2e57ea2c6b13", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848744.245158,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
kQsGoZP2XtiY5RShM6OoiYfA", "size_bytes": 94843, "width": 1114, "height": 632, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "it didn&#x27;t ask yes or no. it
just did this? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"attachments": [{"id": "file-kQsGoZP2XtiY5RShM6OoiYfA", "size": 94843,
"name": "image.png", "mime_type": "image/png", "width": 1114, "height": 632}],
"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21d4e39e290d80-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "fa48a162-47ae-
446c-8508-708082762783", "children": ["1ff44a1b-2347-47c8-9d2e-dc865815de87"]},
"1ff44a1b-2347-47c8-9d2e-dc865815de87": {"id": "1ff44a1b-2347-47c8-9d2e-dc865815de87",
"message": {"id": "1ff44a1b-2347-47c8-9d2e-dc865815de87", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728848761.460293,
"update_time": null, "content": {"content_type": "text", "parts": ["It looks like the
`Set-ExecutionPolicy` command was successfully executed. This change to the policy
will now allow running scripts on your system.\n\nTo confirm if the setting worked,
you can try running the following command again:\n\n```bash\nask --version\n```\n\nIf
this works, you should see the version of the `ask-cli` (Alexa Skills Kit Command Line
Interface).\n\nIf you still encounter issues or the command doesn&#x27;t work, it may
require a reboot or opening a **new** PowerShell window with admin rights. If all goes

~~~
well, you should be able to proceed with the `ask init` step to configure the Alexa
Skill next."]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa270c5-f0fd-45f5-8cb4-2e57ea2c6b13", "request_id": "8d21d4e39e290d80-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa270c5-f0fd-45f5-8cb4-2e57ea2c6b13", "children": ["aaa2587c-c18f-4c27-9268-
6368e2e8693b"]}, "aaa2587c-c18f-4c27-9268-6368e2e8693b": {"id": "aaa2587c-c18f-4c27-
9268-6368e2e8693b", "message": {"id": "aaa2587c-c18f-4c27-9268-6368e2e8693b", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848784.246236,
"update_time": null, "content": {"content_type": "text", "parts": ["we in business.
what next? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21d5e299620d80-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "1ff44a1b-2347-
47c8-9d2e-dc865815de87", "children": ["bb08dfab-ef1a-497e-9854-27a9f1d988db"]},
"bb08dfab-ef1a-497e-9854-27a9f1d988db": {"id": "bb08dfab-ef1a-497e-9854-27a9f1d988db",
"message": {"id": "bb08dfab-ef1a-497e-9854-27a9f1d988db", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728848809.175434,
"update_time": null, "content": {"content_type": "text", "parts": ["Great! Now that
the `ask-cli` (Alexa Skills Kit Command Line Interface) is set up, you&#x27;re ready
to start creating your Alexa skill.\n\nHere are the next steps:\n\n### 1. **Initialize
Your Skill Project**\n   Run the following command to initialize your skill
project:\n\n   ```bash\n   ask init\n   ```\n\n   - This command will prompt you to
log in to your Amazon Developer account (ensure that you&#x27;re signed up at [Amazon
Developer Console](https://developer.amazon.com)).\n   - Follow the prompts to link
the CLI with your Alexa Developer account.\n\n### 2. **Clone the Repository**\n
Since you are using the project from the GitHub repository, you need to clone it to
your local machine. Run the following command:\n\n   ```bash\n   git clone
https://github.com/Joao208/alexa-chatgpt.git\n   ```\n\n   Then navigate into the
project directory:\n\n   ```bash\n   cd alexa-chatgpt\n   ```\n\n### 3. **Insert Your
OpenAI API Key**\n   Open the `index.js` file in the project directory. This is where
you&#x27;ll add your OpenAI API key.\n\n   - Find the line where it mentions the API
key in the `index.js` file (likely where the ChatGPT API is called) and replace it
with your OpenAI API key.\n   - Save the file after adding the key.\n\n### 4. **Deploy
Your Skill**\n   Once you&#x27;ve added your OpenAI API key, deploy the skill using
the following command:\n\n   ```bash\n   ask deploy\n   ```\n\n   This will push your
skill to the Alexa Developer Console and set up the necessary Lambda functions.\n\n###
5. **Test the Skill**\n   After deployment, you can test the skill through the Alexa
Developer Console or on your Alexa device. You can also simulate requests with:\n\n
```bash\n   ask simulate\n   ```\n\n   Example of testing the skill with a sample
phrase:\n\n   ```bash\n   ask simulate --text &quot;Ask my ChatGPT skill how the
weather is&quot; --locale &quot;en-US&quot;\n   ```\n\nThis will allow you to see the
responses Alexa will give for the skill you&#x27;ve created.\n\nOnce you&#x27;ve
successfully deployed and tested it, you&#x27;re ready to refine or expand the
functionality further! Let me know if you&#x27;d like to adjust or add more
features."]}, "status": "finished_successfully","end_turn": true
~~~

, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa2587c-c18f-4c27-9268-6368e2e8693b", "request_id": "8d21d5e299620d80-MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent": "aaa2587c-c18f-4c27-9268-6368e2e8693b", "children": ["aaa2dd66-649a-49a0-a1a6-94b8bfdb49e6"]}, "aaa2dd66-649a-49a0-a1a6-94b8bfdb49e6": {"id": "aaa2dd66-649a-49a0-a1a6-94b8bfdb49e6", "message": {"id": "aaa2dd66-649a-49a0-a1a6-94b8bfdb49e6", "author": {"role": "user", "name": null, "metadata": {}}, "create_time": 1728848862.533398, "update_time": null, "content": {"content_type": "text", "parts": ["PS C:\\Users\\alexi&gt; Set-ExecutionPolicy RemoteSigned -Scope CurrentUser\r\nPS C:\\Users\\alexi&gt; ask --version\r\n2.30.7\r\nPS C:\\Users\\alexi&gt; ask init\r\n[Error]: File C:\\Users\\alexi\\.ask\\cli_config not exists.\n\n\nanother snag... what now? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id": "8d21d7cc2f9e5a19-MXP", "message_source": null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent": "bb08dfab-ef1a-497e-9854-27a9f1d988db", "children": ["589c5fe5-ffe4-4753-97f5-31a02a6ef30e"]}, "589c5fe5-ffe4-4753-97f5-31a02a6ef30e": {"id": "589c5fe5-ffe4-4753-97f5-31a02a6ef30e", "message": {"id": "589c5fe5-ffe4-4753-97f5-31a02a6ef30e", "author": {"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728848886.438359, "update_time": null, "content": {"content_type": "text", "parts": ["The error you&#x27;re seeing indicates that the CLI configuration file (`cli_config`) is missing, which is required for the `ask init` command to work properly.\n\nHere&#x27;s how you can resolve this:\n\n### 1. **Create the Configuration File Manually**\n   It seems that the initialization is having trouble creating the configuration file in the `.ask` folder. You can manually create it using the following steps:\n   \n   - Navigate to `C:\\Users\\alexi` in your file explorer.\n   - Create a new folder named `.ask`.\n   - Inside that folder, create a new file called `cli_config`.\n\n   The content of this file can be simple and looks like this:\n\n   ```json\n   {\n &quot;profiles&quot;: {\n   &quot;default&quot;: {\n &quot;aws_profile&quot;: &quot;default&quot;\n   }\n   }\n   }\n   ```\n\n Save this file and retry the `ask init` command.\n\n### 2. **Check File Permissions**\n   If the folder exists but the file couldn\u2019t be created, make sure you have the correct permissions to write in the `C:\\Users\\alexi` directory. You might want to run the terminal as an administrator or manually verify your write permissions for that folder.\n\n### 3. **Alternative Method - Reinitialize via Login**\n   If the above steps don&#x27;t work, you could try manually configuring the Alexa CLI via login. First, run the following command:\n\n   ```bash\n   ask configure\n   ```\n\n   This will prompt you to log in to your Amazon Developer account and set up the necessary files, bypassing the need for the missing `cli_config` file.\n\nOnce you\u2019ve done this, try running `ask init` again. If successful, it should proceed with the initialization process.\n\nLet me know if you need more help!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa2dd66-649a-49a0-a1a6-94b8bfdb49e6", "request_id": "8d21d7cc2f9e5a19-MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":

```
"aaa2dd66-649a-49a0-a1a6-94b8bfdb49e6", "children": ["aaa2a427-e49a-4185-b07d-
3eca73bc555a"]}, "aaa2a427-e49a-4185-b07d-3eca73bc555a": {"id": "aaa2a427-e49a-4185-
b07d-3eca73bc555a", "message": {"id": "aaa2a427-e49a-4185-b07d-3eca73bc555a", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728848987.76824,
"update_time": null, "content": {"content_type": "text", "parts": ["whats the file
type. like  cli_config. what? "]}, "status": "finished_successfully", "end_turn": null,
"weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []},
"request_id": "8d21dadaaa535a19-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "589c5fe5-ffe4-
4753-97f5-31a02a6ef30e", "children": ["437d56cc-2ffa-4bae-b1e9-8fd1a6df76bc"]},
"437d56cc-2ffa-4bae-b1e9-8fd1a6df76bc": {"id": "437d56cc-2ffa-4bae-b1e9-8fd1a6df76bc",
"message": {"id": "437d56cc-2ffa-4bae-b1e9-8fd1a6df76bc", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728849011.225532,
"update_time": null, "content": {"content_type": "text", "parts": ["The file type for
`cli_config` in this case should be **JSON**. So, when creating the file, make sure to
save it as `cli_config.json`.\n\nHere are the steps to follow:\n\n1. **Open a text
editor** like Notepad (or any other code editor).\n2. **Copy the content** provided
below into the editor:\n\n   ```json\n   {\n      &quot;profiles&quot;: {\n
&quot;default&quot;: {\n            &quot;aws_profile&quot;:
&quot;default&quot;\n         }\n      }\n   }\n   ```\n\n3. **Save the file as**
`cli_config.json` inside the `.ask` folder (which should be at
`C:\\Users\\alexi\\.ask\\`).\n4. Make sure that **file extensions are visible** in
Windows, so you don\u2019t accidentally save it as `cli_config.json.txt`. To verify,
when saving the file, select *All Files* in the Save As Type dropdown and manually add
`.json` to the filename.\n\nAfter saving, try running the `ask init` command again."]},
"status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2a427-e49a-4185-b07d-3eca73bc555a", "request_id": "8d21dadaaa535a19-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2a427-e49a-4185-b07d-3eca73bc555a", "children": ["aaa28a64-e66c-4d5b-82c3-
f83117cd48dd"]}, "aaa28a64-e66c-4d5b-82c3-f83117cd48dd": {"id": "aaa28a64-e66c-4d5b-
82c3-f83117cd48dd", "message": {"id": "aaa28a64-e66c-4d5b-82c3-f83117cd48dd", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728849211.386513,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
FWWTbE9KTVDNq302o5gHteRK", "size_bytes": 95698, "width": 1133, "height": 629, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "explain this: "]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-FWWTbE9KTVDNq302o5gHteRK", "size": 95698, "name": "Screenshot 2024-10-13
215259.png", "mime_type": "image/png", "width": 1133, "height": 629}],
"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21e1463f7d524a-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "437d56cc-2ffa-
4bae-b1e9-8fd1a6df76bc", "children": ["30c2e613-6618-4ff7-9924-f5bd6868db04",
"297c3765-ca12-4f86-86d5-a77e0770d396"]}, "30c2e613-6618-4ff7-9924-f5bd6868db04":
{"id": "30c2e613-6618-4ff7-9924-f5bd6868db04", "message": {"id": "30c2e613-6618-4ff7-
```

9924-f5bd6868db04", "author": {"role": "assistant", "name": null, "metadata": {}},
"create_time": 1728849221.071581, "update_time": null, "content": {"content_type":
"text", "parts": [""]}, "status": "in_progress", "end_turn": null, "weight": 1.0,
"metadata": {"citations": [], "content_references": [], "gizmo_id": null,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa28a64-e66c-4d5b-82c3-f83117cd48dd", "finish_details": {"type":
"interrupted"}, "request_id": "8d21e04d187d599b-MXP", "timestamp_": "absolute"},
"recipient": "all", "channel": null}, "parent": "aaa28a64-e66c-4d5b-82c3-f83117cd48dd",
"children": []}, "297c3765-ca12-4f86-86d5-a77e0770d396": {"id": "297c3765-ca12-4f86-
86d5-a77e0770d396", "message": {"id": "297c3765-ca12-4f86-86d5-a77e0770d396", "author":
{"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728849272.239236,
"update_time": null, "content": {"content_type": "text", "parts": ["The error message
you are encountering (`[Error]: File C:\\Users\\alexi\\.ask\\cli_config not exists.`)
means that the ASK CLI is unable to find a necessary configuration file. This
configuration file typically contains information about your AWS and Alexa Developer
accounts, allowing the ASK CLI to interact with these services.\n\nHere\u2019s how to
resolve this issue:\n\n1. **Create the Missing Configuration**:\n   Since the error
mentions the `.ask/cli_config` file is missing, you can use the `ask init` command to
create it properly.\n\n2. **Try Running `ask init` in Administrator Mode**:\n   Make
sure you&#x27;re still running the PowerShell as Administrator, and then retry the
following steps:\n   ```bash\n   ask init\n   ```\n\n   This command should guide you
through setting up the configuration by asking you for your **Amazon Developer**
account credentials and **AWS** details.\n\n3. **Follow the Setup Prompts**:\n   -
When you run `ask init`, it will ask for your credentials for the **Alexa Developer
Console** and **AWS**.\n   - It will also help configure the environments to work with
your Alexa skills.\n\nIf the configuration file is still not being created, you may
want to manually create a configuration file.\n\nLet me know if you&#x27;re able to
get through the setup or if you need further assistance!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa28a64-e66c-4d5b-82c3-f83117cd48dd", "request_id": "8d21e1463f7d524a-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa28a64-e66c-4d5b-82c3-f83117cd48dd", "children": ["aaa270e2-9d78-46ab-9921-
8c28dab32cb6"]}, "aaa270e2-9d78-46ab-9921-8c28dab32cb6": {"id": "aaa270e2-9d78-46ab-
9921-8c28dab32cb6", "message": {"id": "aaa270e2-9d78-46ab-9921-8c28dab32cb6", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728849385.466303,
"update_time": null, "content": {"content_type": "text", "parts": ["i don&#x27;t have
anything with AWs. I don&#x27;t want it. had a bad experience with them where they
stole my money"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21e48f4ce95a19-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "297c3765-ca12-
4f86-86d5-a77e0770d396", "children": ["9ad61837-4e10-4aed-88be-d6ff58332899"]},
"9ad61837-4e10-4aed-88be-d6ff58332899": {"id": "9ad61837-4e10-4aed-88be-d6ff58332899",
"message": {"id": "9ad61837-4e10-4aed-88be-d6ff58332899", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728849406.383489,
"update_time": null, "content": {"content_type": "text", "parts": ["If you&#x27;re

```
looking to avoid using AWS for your Alexa Skill due to a previous negative experience,
you can still create and deploy Alexa skills by working within the **Alexa Developer
Console** and using external APIs for the backend, like **OpenAI**. You don&#x27;t
necessarily have to rely on AWS for hosting your backend.\n\nHere are a few
alternatives:\n\n### 1. **Using External Services (like OpenAI) without AWS Lambda**\n
- You can use **OpenAI&#x27;s API** directly for the backend functionality, which will
handle your ChatGPT integration. You only need to make sure that your Alexa skill
invokes the API when needed.\n   - For hosting your API logic, you can use non-AWS
platforms like **Heroku**, **Vercel**, or **Glitch**, which offer free tiers for small
projects.\n\n### 2. **Alexa Developer Console**\n   The Alexa Developer Console allows
you to host some simple skill logic directly, without needing AWS. For more complex
scenarios (like integrating with ChatGPT), you will need some sort of backend service
to process requests and return responses.\n\n### Steps to Avoid AWS in Skill
Development:\n   - **Set up the Alexa Developer Console**: You can build skills
directly in the Alexa Developer Console.\n   - **Use an Alternative Hosting Solution**:
Host your backend logic using another service like Heroku or Vercel and integrate it
with your Alexa skill through API calls.\n   - **Implement a Direct API Integration**:
Use the `index.js` file in the skill\u2019s code to call external services like OpenAI
for ChatGPT responses.\n\n   This way, you can develop the skill without relying on
AWS and use alternative solutions for API calls and backend processing.\n\nLet me know
if you&#x27;d like more detailed steps on integrating with Heroku or Vercel, or how to
set up your Alexa skill in this alternative way!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa270e2-9d78-46ab-9921-8c28dab32cb6", "request_id": "8d21e48f4ce95a19-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa270e2-9d78-46ab-9921-8c28dab32cb6", "children": ["aaa24d89-d3c4-4e2a-9e70-
b1cc276a053e"]}, "aaa24d89-d3c4-4e2a-9e70-b1cc276a053e": {"id": "aaa24d89-d3c4-4e2a-
9e70-b1cc276a053e", "message": {"id": "aaa24d89-d3c4-4e2a-9e70-b1cc276a053e", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728849979.042524,
"update_time": null, "content": {"content_type": "text", "parts": ["which of the
services doesn&#x27;t require me to give billing information like a credit card? "]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21f30ddefeba9a-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "9ad61837-4e10-
4aed-88be-d6ff58332899", "children": ["867da153-52e3-4321-a080-538264391ae4"]},
"867da153-52e3-4321-a080-538264391ae4": {"id": "867da153-52e3-4321-a080-538264391ae4",
"message": {"id": "867da153-52e3-4321-a080-538264391ae4", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728850014.218904,
"update_time": null, "content": {"content_type": "text", "parts": ["If you want to
avoid providing billing information like a credit card, you can consider the following
platforms:\n\n### 1. **Glitch**\n   - **No credit card required**: Glitch offers a
free tier that allows you to build and host small web applications without the need to
provide any billing details.\n   - **Simple to use**: It\u2019s beginner-friendly and
you can deploy your app quickly without too much setup.\n   - **Limitations**: The
free tier is limited in terms of uptime, as projects will sleep after 5 minutes of
```

```
inactivity, but for testing an Alexa skill, it should be sufficient.\n\n   [Visit
Glitch](https://glitch.com/)\n\n### 2. **GitHub Pages**\n   - **No credit card
required**: GitHub Pages is completely free and requires no billing info.\n   -
**Great for static sites**: While it is designed for hosting static websites, you
could use it if your skill doesn&#x27;t require server-side logic.\n   -
**Limitations**: GitHub Pages only works for static content. If your Alexa skill
requires dynamic content (e.g., making API requests), this might not be the best
choice.\n\n   [Visit GitHub Pages](https://pages.github.com/)\n\n### 3. **Replit**\n
- **No credit card required**: Replit offers free hosting and an integrated
development environment (IDE). You can build, test, and deploy small applications
without providing billing information.\n   - **Limitations**: Free plans have some
limitations in terms of CPU and memory, but it\u2019s good for small-scale projects or
testing.\n\n   [Visit Replit](https://replit.com/)\n\n### 4. **Vercel**\n   - **No
credit card required initially**: Vercel provides a generous free tier, and you can
deploy projects without giving credit card information upfront. \n   - **Limitations**:
Vercel\u2019s free plan is good for small to medium projects, but you will need to
upgrade if your usage grows significantly.\n\n   [Visit
Vercel](https://vercel.com/)\n\n### 5. **Netlify**\n   - **No credit card required**:
Netlify offers free hosting for static sites, similar to GitHub Pages, but with
additional features like form handling and continuous deployment.\n   -
**Limitations**: Like GitHub Pages, it works well for static sites. Dynamic content
may require more complex workarounds.\n\n   [Visit
Netlify](https://www.netlify.com/)\n\nIf you are looking for a solution where you can
avoid providing billing details entirely, **Glitch** and **Replit** are the most
suitable options for small-scale Alexa Skill development with a free backend.\n\nLet
me know if you&#x27;d like guidance on how to get started with any of these
platforms!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa24d89-d3c4-4e2a-9e70-b1cc276a053e", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d21f30ddefeba9a-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa24d89-
d3c4-4e2a-9e70-b1cc276a053e", "children": ["aaa2e369-fe1b-48bc-a179-e85ff2f60774"]},
"aaa2e369-fe1b-48bc-a179-e85ff2f60774": {"id": "aaa2e369-fe1b-48bc-a179-e85ff2f60774",
"message": {"id": "aaa2e369-fe1b-48bc-a179-e85ff2f60774", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728850153.226532, "update_time": null,
"content": {"content_type": "text", "parts": ["okay i&#x27;m on glitch what do i do
now? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21f74e2c4cbafd-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "867da153-52e3-
4321-a080-538264391ae4", "children": ["59a0493f-92c9-431e-a38f-3f4303e34eb4"]},
"59a0493f-92c9-431e-a38f-3f4303e34eb4": {"id": "59a0493f-92c9-431e-a38f-3f4303e34eb4",
"message": {"id": "59a0493f-92c9-431e-a38f-3f4303e34eb4", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728850198.841614,
"update_time": null, "content": {"content_type": "text", "parts": ["Great! Now that
you&#x27;re on Glitch, let&#x27;s get your Alexa Skill running there. Here&#x27;s a
step-by-step guide to follow:\n\n### Step 1: Create a New Project\n1. Go to
[Glitch](https://glitch.com/).\n2. On the homepage, click **&quot;New
```

```
Project&quot;**.\n3. Select **&quot;Create a Node App&quot;** since your Alexa Skill
will use Node.js to communicate with the OpenAI API.\n\n### Step 2: Set Up Your
Project Environment\n1. Once the project is created, you&#x27;ll see some files like
`package.json`, `server.js`, etc.\n2. Rename **`server.js`** to **`index.js`**, as
that&#x27;s where you will place your code.\n3. Open the **`package.json`** file, and
add any necessary dependencies. You will need `axios` to make HTTP requests to
OpenAI&#x27;s API. It should look something like this:\n\n   ```json\n   {\n
&quot;name&quot;: &quot;alexa-chatgpt&quot;,\n     &quot;version&quot;:
&quot;1.0.0&quot;,\n     &quot;description&quot;: &quot;Alexa Skill powered by
OpenAI&#x27;s ChatGPT&quot;,\n     &quot;main&quot;: &quot;index.js&quot;,\n
&quot;scripts&quot;: {\n       &quot;start&quot;: &quot;node index.js&quot;\n     },\n
&quot;dependencies&quot;: {\n       &quot;express&quot;: &quot;^4.17.1&quot;,\n
&quot;axios&quot;: &quot;^0.21.1&quot;\n     }\n   }\n   ```\n\n4. After you modify
`package.json`, Glitch will automatically install the necessary dependencies.\n\n###
Step 3: Write the Alexa Skill Code\n1. In **`index.js`**, write the basic Express
server setup:\n\n   ```js\n   const express = require(&#x27;express&#x27;);\n   const
axios = require(&#x27;axios&#x27;);\n   const app = express();\n\n
app.use(express.json());\n\n   app.post(&#x27;/alexa&#x27;, async (req, res) =&gt; {\n
const alexaRequest = req.body;\n       const userQuery =
alexaRequest.request.intent.slots.query.value;\n\n       try {\n           // Make a
request to OpenAI API\n           const openAIResponse = await axios.post(\n
&#x27;https://api.openai.com/v1/chat/completions&#x27;,\n               {\n
model: &quot;gpt-3.5-turbo&quot;,\n                   messages: [{ role:
&quot;user&quot;, content: userQuery }]\n               },\n               {\n
headers: {\n                       &#x27;Authorization&#x27;: `Bearer
YOUR_OPENAI_API_KEY`,\n                       &#x27;Content-Type&#x27;:
&#x27;application/json&#x27;\n                   }\n               }\n           );\n\
n           const aiResponse = openAIResponse.data.choices[0].message.content;\n\n
// Construct a response for Alexa\n           res.json({\n               version:
&#x27;1.0&#x27;,\n               response: {\n                   outputSpeech: {\n
type: &#x27;PlainText&#x27;,\n                       text:
aiResponse,\n                   },\n                   shouldEndSession:
true,\n               }\n           });\n       } catch (error) {\n
console.error(error);\n           res.status(500).json({ error: &quot;Something went
wrong&quot; });\n       }\n   });\n\n   // Start the server\n   const PORT =
process.env.PORT || 3000;\n   app.listen(PORT, () =&gt; {\n       console.log(`Server
is running on port ${PORT}`);\n   });\n   ```\n\n2. Replace **`YOUR_OPENAI_API_KEY`**
with your actual OpenAI API key.\n\n### Step 4: Configure Alexa to Use Your Glitch
App\n1. Once the code is working, click **&quot;Share&quot;** in the Glitch project
and copy the live link (the URL that ends with `glitch.me`).\n2. Go to the [Alexa
Developer Console](https://developer.amazon.com/alexa/console/ask) and set up a new
Alexa Skill.\n3. In the **&quot;Endpoint&quot;** section of the Alexa Developer
Console, paste the **Glitch URL** followed by `/alexa` (e.g.,
`https://yourprojectname.glitch.me/alexa`).\n\n### Step 5: Test Your Alexa Skill\n1.
Use the Alexa Developer Console to test the skill by typing or speaking a request to
Alexa.\n2. Your Glitch server will receive the request, process it with OpenAI, and
return the response to Alexa.\n\n### Step 6: Make Sure the Server Keeps Running\nTo
ensure the server doesn&#x27;t go to sleep, you may want to set up a ping service like
[UptimeRobot](https://uptimerobot.com/) to periodically ping your Glitch URL.\n\nLet
```

```
me know if you need help with any of these steps!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2e369-fe1b-48bc-a179-e85ff2f60774", "request_id": "8d21f74e2c4cbafd-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa2e369-fe1b-48bc-a179-e85ff2f60774", "children": ["aaa24536-3817-4e13-aeb1-
111ab5c50696"]}, "aaa24536-3817-4e13-aeb1-111ab5c50696": {"id": "aaa24536-3817-4e13-
aeb1-111ab5c50696", "message": {"id": "aaa24536-3817-4e13-aeb1-111ab5c50696", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728850322.690586,
"update_time": null, "content": {"content_type": "text", "parts": ["my partner say
apollo oracle alexa skill already does what i want and connects to chat gpt"]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d21fb7239a0bb03-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "59a0493f-92c9-
431e-a38f-3f4303e34eb4", "children": ["c5754f56-7f27-4804-ba2e-c1b18790d654"]},
"c5754f56-7f27-4804-ba2e-c1b18790d654": {"id": "c5754f56-7f27-4804-ba2e-c1b18790d654",
"message": {"id": "c5754f56-7f27-4804-ba2e-c1b18790d654", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728850338.796531,
"update_time": null, "content": {"content_type": "text", "parts": ["It sounds like
your partner is referring to the **Apollo Oracle Alexa skill**, which connects Alexa
to OpenAI&#x27;s GPT models, allowing users to ask questions or have conversations
similar to what you&#x27;re aiming for. If this skill already meets your needs, it
might be easier to use it rather than developing your own from scratch.\n\nHere are a
couple of points to consider:\n1. **Ease of Use**: If Apollo Oracle already exists and
does what you want, this could save you time and effort. It might allow you to start
using ChatGPT with Alexa right away without needing to code or set up hosting.\n   \n2.
**Customization**: If you want something more personalized or specific, you might
still prefer to develop your own skill. Even though Apollo Oracle offers similar
functionality, building your own skill can allow you to tailor responses, integrate
with other services, or include features that the existing skill doesn&#x27;t
offer.\n\nIf Apollo Oracle fits your needs, you could try it out by enabling the skill
on your Alexa device through the Alexa Skills Store. If you find it lacking or want to
expand on it, you can always revisit building your custom skill.\n\nLet me know if
you&#x27;d like help comparing the features or exploring further customization
options!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa24536-3817-4e13-aeb1-111ab5c50696", "request_id": "8d21fb7239a0bb03-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa24536-3817-4e13-aeb1-111ab5c50696", "children": ["aaa2c3bf-5d4c-4b13-9b6e-
e8209a3fb592"]}, "aaa2c3bf-5d4c-4b13-9b6e-e8209a3fb592": {"id": "aaa2c3bf-5d4c-4b13-
9b6e-e8209a3fb592", "message": {"id": "aaa2c3bf-5d4c-4b13-9b6e-e8209a3fb592", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728850369.23076,
"update_time": null, "content": {"content_type": "text", "parts": ["how can I find out
more about apollo oracle? "]}, "status": "finished_successfully", "end_turn": null,
"weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []},
"request_id": "8d21fc94bb9e0e1a-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "c5754f56-7f27-
4804-ba2e-c1b18790d654", "children": ["455e8fe2-b3e6-45f9-8398-ac10e1e12382"]},
"455e8fe2-b3e6-45f9-8398-ac10e1e12382": {"id": "455e8fe2-b3e6-45f9-8398-ac10e1e12382",
"message": {"id": "455e8fe2-b3e6-45f9-8398-ac10e1e12382", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728850395.8925,
"update_time": null, "content": {"content_type": "code", "language": "unknown",
"response_format_name": null, "text": "search(&quot;Apollo Oracle Alexa skill&quot;)"},
"status": "finished_successfully", "end_turn": false, "weight": 1.0, "metadata":
{"citations": [], "content_references": [], "gizmo_id": null, "message_type": null,
"model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa2c3bf-5d4c-
4b13-9b6e-e8209a3fb592", "finish_details": {"type": "stop", "stop_tokens": [200007]},
"is_complete": true, "request_id": "8d21fc94bb9e0e1a-MXP", "timestamp_": "absolute"},
"recipient": "browser", "channel": null}, "parent": "aaa2c3bf-5d4c-4b13-9b6e-
e8209a3fb592", "children": ["50e79d24-6dcb-4e30-ab10-bd0a5bcf4ad9"]}, "50e79d24-6dcb-
4e30-ab10-bd0a5bcf4ad9": {"id": "50e79d24-6dcb-4e30-ab10-bd0a5bcf4ad9", "message":
{"id": "50e79d24-6dcb-4e30-ab10-bd0a5bcf4ad9", "author": {"role": "tool", "name":
"browser", "metadata": {}}, "create_time": 1728850395.892613, "update_time": null,
"content": {"content_type": "tether_browsing_display", "result": "#
\u30100\u2020chatGPT on top of Amazon Alexa - Page 9 -
Community\u2020community.openai.com\u3011\nabout the Alexa skill Oracle of Light
Amazon.com: oracle of light : Alexa Skills Advanced Mode. Marty and Lucy discuss the
new AI feature, Oracle of Light, for Amazon\u2019s Alexa (referred to as \u201cLady
A\u201d) ... La skill \u00e8 qui: Amazon.it: oracolo di apollo : Alexa Skills.\n#
\u30101\u2020chatGPT on top of Amazon Alexa - Community - OpenAI Developer
Forum\u2020community.openai.com\u3011\nA user shares several skills for Amazon Alexa
that use chatGPT, a text-generation model by OpenAI, to engage in conversations on
various topics. Another user asks how to launch the skill without opening a game on
Luna app.\n# \u30102\u2020Apollo Alexa Skill - GitHub\u2020github.com\u3011\nApollo
Alexa Skill \\n. An AWS Lambda function that bridges Alexa Smart Home commands to the
Apollo Home Control application. \\n Purpose \\n. While it would be possible to sent
commands directly to Apollo Home Control application, that would require opening up a
port inside the home network and managing the security that goes along with that.\n#
\u30103\u2020chatGPT on top of Amazon Alexa - Page 8 -
Community\u2020community.openai.com\u3011\nThe Alexa skill \u2018Oracle of Light\u2019
is the reference skill on Astro robot. It\u2019s the best way to talk to Astro as
chatGPT. ... L\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che dialoga con
chatGPT e ti invia la conversazione per mail! sono_matteo October 24, 2023, 8:37pm 152.
Latency ...\n# \u30104\u2020Amazon.com: oracle of light : Alexa
Skills\u2020www.amazon.com\u3011\nYou need to grant permission to use your email in
the Alexa App, by selecting the skill and then its Settings. Engage the Oracle of
Light in a conversation on a topic of your choice. The skill connects to the chatGPT
language model developed by OpenAI, trained using a large amount of textual data and
able to answer questions and provide ...\n# \u30105\u2020raypp2/Apollo-Alexa-Skill -
GitHub\u2020github.com\u3011\nWhile it would be possible to sent commands directly to
Apollo Home Control application, that would require opening up a port inside the home
network and managing the security that goes along with that. Lambda is reasonably fast,
free at low execution volumes, and integrates more easily with the ...\n#
```

\u30106\u2020Oracle Digital Assistant TechExchange - Oracle Blogs\u2020blogs.oracle.com\u3011\nConfigure the Alexa Skill Code App With the Alexa skill created in the Amazon developer console, the next step is to connect it with a remote service that implements the task to be executed when the skill is invoked. To connect Alexa with Oracle Digital Assistant, you will be using the code provided in the downloads for this Oracle\n# \u30107\u2020Build Your Skill | Alexa Skills Kit - developer.amazon.com\u2020developer.amazon.com\u3011\nLearn how to create and publish Alexa skills using the Alexa Skills Kit (ASK), which includes development tools and libraries. Find out the prerequisites, features, and resources for different skill types and voice interaction models.\n# \u30108\u2020Alexa Skill and Apex - has anyone tried this yet? - Oracle Forums\u2020forums.oracle.com\u3011\nHas anyone else on the forum tried any development with Apex and Voice Search (Alexa, Siri etc) I am planning on creating an Alexa skill to query a sample Oracle database using APEX and would be interested to learn if anyone has done this, and any advice/guidance they could share. The only example I could find is from Christoph Ruepprich\n# \u30109\u2020Querying an Oracle Database Using Amazon Alexa - DSP\u2020content.dsp.co.uk\u3011\nThis blog will show you how to get up and running with this and create an Alexa skill that queries your Oracle database. There are three main building blocks needed to get this working in its simplest form: ORDS Web Service (configured either in APEX or on the database). AWS Lambda function (anonymous function). Alexa Skill.\n# \u301010\u2020How to Build Your First Alexa Skill: 5 Steps to Get Started\u2020developer.amazon.com\u3011\nAlexa Skill Builder&#x27;s Guides; Connect with Alexa Evangelists on Twitch; Build Your First Skill, Get a T-Shirt. Bring your big idea to life with Alexa and earn perks through our milestone-based developer promotion. US developers, publish your first Alexa skill and earn a custom Alexa developer t-shirt.\n# \u301011\u2020La skill Alexa &quot;Oracolo di Apollo&quot; per parlare con ChatGPT\u2020www.youtube.com\u3011\n\ud83c\udf1f Prova la skill Oracolo di Apollo! \ud83c\udf10 Immergiti in conversazioni illimitate con ChatGPT su Alexa. Dalla scienza alla poesia, conosce tutto. \ud83d\udcda NOVIT\u00c0: In...\n# \u301012\u2020Alexa Skills - Amazon Customer Service\u2020www.amazon.com\u3011\nLearn how to use Alexa Skills, which are like apps that you can enable to help you perform more voice-activated actions. Find out how to turn skills on or off, troubleshoot issues, and access in-skill purchases and celebrity personality.\n# \u301013\u2020chatGPT on top of Amazon Alexa - Page 5 - Community\u2020community.openai.com\u3011\nNEW FUNCTIONALITY: \u2192 Orakel sends e-mail to the user I have just added a new feature to the German skill \u201cOrakel des Lichts\u201d: it is now possible to receive the current conversation by email. Whenever the user says \u201c&lt;&lt;Schick mir eine E-Mail&gt;&gt;,\u201d the Orakel will send the conversation to the email address associated with the user\u2019s Amazon account.\n# \u301014\u2020Oracolo di Apollo - la skill Alexa che si connette a chatGPT ... - YouTube\u2020www.youtube.com\u3011\nIngaggia l&#x27;Oracolo di Apollo in una conversazione su un argomento a tua scelta. L&#x27;Oracolo \u00e8 in grado di rispondere a tutte le tue curiosit\u00e0. Conosce tutte le...\n# \u301015\u2020Oracle APEX \u2013 Amazon ALEXA integration WITHOUT Amazon ... - Blogger\u2020apexutil.blogspot.com\u3011\nIn the last blog post I prepared Oracle side. In this post I will show, how to prepare ALEXA Skill and how to integrate all layers. Few weeks ago Amazon implemented new version of Alexa Skill Kit. Alexa Skill Kit is like a APEX builder. Low code technology, so you don&#x27;t need to write any code :) Let&#x27;s start ! Go to Amazon Developer Console 1.\n# \u301016\u2020Talk to Your ERP

with ALEXA and ORDS - JMJ CLOUD\u2020www.jmjcloud.com\u3011\nIt passes this speech to
the Alexa Voice Service (AVS) in the cloud. Amazon converts the speech to text and
attempts to determine the skill that was requested based on the invocation name. In
the above example, \u2018Ask My Database\u2019. Once it recognizes the skill, it forms
as structured JSON request and sends it to our ORDS REST end-point.\n#
\u301017\u2020Apollo Hospitals\u2019 Amazon Alexa Skill to enable users access
72 ...\u2020www.thehindubusinessline.com\u3011\nThe skill can also be accessed via the
Alexa app available on iOS and Android. Users can simply say, \u201cAlexa, open Apollo,
to find me a doctor for consultation, to book an appointment\u201d, among ...\n#
\u301018\u2020What Are Alexa In-Skill Purchases? - Amazon Customer
Service\u2020www.amazon.com\u3011\nTypes of in-skill purchases: One-time Purchase -
Unlock access to features or content within a skill. Typically available to use on any
compatible Alexa device registered to your account.\nVisible: 0% - 100%", "summary":
null, "assets": [], "tether_id": null}, "status": "finished_successfully", "end_turn":
null, "weight": 0.0, "metadata": {"_cite_metadata": {"citation_format": {"name":
"tether_og"}, "metadata_list": [{"type": "webpage", "title": "chatGPT on top of Amazon
Alexa - Page 9 - Community", "url": "https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=9", "text": "\nabout the Alexa skill Oracle of Light
Amazon.com: oracle of light : Alexa Skills Advanced Mode. Marty and Lucy discuss the
new AI feature, Oracle of Light, for Amazon\u2019s Alexa (referred to as \u201cLady
A\u201d) ... La skill \u00e8 qui: Amazon.it: oracolo di apollo : Alexa Skills.\n",
"pub_date": "2024-09-29T15:03:00.0000000Z", "extra": null}, {"type": "webpage",
"title": "chatGPT on top of Amazon Alexa - Community - OpenAI Developer Forum", "url":
"https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515", "text": "\nA
user shares several skills for Amazon Alexa that use chatGPT, a text-generation model
by OpenAI, to engage in conversations on various topics. Another user asks how to
launch the skill without opening a game on Luna app.\n", "pub_date": "2024-10-11T14:38:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Apollo Alexa
Skill - GitHub", "url": "https://github.com/raypp2/Apollo-Alexa-Skill/blob/main/README.md", "text": "\nApollo Alexa Skill \\n. An AWS Lambda function
that bridges Alexa Smart Home commands to the Apollo Home Control application. \\n
Purpose \\n. While it would be possible to sent commands directly to Apollo Home
Control application, that would require opening up a port inside the home network and
managing the security that goes along with that.\n", "pub_date": "2023-10-23T01:40:00.0000000Z", "extra": null}, {"type": "webpage", "title": "chatGPT on top of
Amazon Alexa - Page 8 - Community", "url": "https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=8", "text": "\nThe Alexa skill \u2018Oracle of
Light\u2019 is the reference skill on Astro robot. It\u2019s the best way to talk to
Astro as chatGPT. ... L\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che
dialoga con chatGPT e ti invia la conversazione per mail! sono_matteo October 24, 2023,
8:37pm 152. Latency ...\n", "pub_date": "2024-10-07T05:12:00.0000000Z", "extra": null},
{"type": "webpage", "title": "Amazon.com: oracle of light : Alexa Skills", "url":
"https://www.amazon.com/Revolutionary-Voice-oracle-of-light/dp/B0BRY898CK", "text":
"\nYou need to grant permission to use your email in the Alexa App, by selecting the
skill and then its Settings. Engage the Oracle of Light in a conversation on a topic
of your choice. The skill connects to the chatGPT language model developed by OpenAI,
trained using a large amount of textual data and able to answer questions and
provide ...\n", "pub_date": "2024-04-15T15:09:00.0000000Z", "extra": null}, {"type":
"webpage", "title": "raypp2/Apollo-Alexa-Skill - GitHub", "url":

"https://github.com/raypp2/Apollo-Alexa-Skill", "text": "\nWhile it would be possible to sent commands directly to Apollo Home Control application, that would require opening up a port inside the home network and managing the security that goes along with that. Lambda is reasonably fast, free at low execution volumes, and integrates more easily with the ...\n", "pub_date": "2024-10-05T07:10:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Oracle Digital Assistant TechExchange - Oracle Blogs", "url": "https://blogs.oracle.com/content/published/api/v1.1/assets/CONTF9BCB15A72F742E08BC49321AB27BAE8/native/Adding+Alexa+as+a+channel+in+your+chatbot_v3.pdf?channelToken=98b861d3404c487a8d8696715f2b694f", "text": "\nConfigure the Alexa Skill Code App With the Alexa skill created in the Amazon developer console, the next step is to connect it with a remote service that implements the task to be executed when the skill is invoked. To connect Alexa with Oracle Digital Assistant, you will be using the code provided in the downloads for this Oracle\n", "pub_date": "2024-10-12T06:24:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Build Your Skill | Alexa Skills Kit - developer.amazon.com", "url": "https://developer.amazon.com/en-US/docs/alexa/build/build-your-skill-overview.html", "text": "\nLearn how to create and publish Alexa skills using the Alexa Skills Kit (ASK), which includes development tools and libraries. Find out the prerequisites, features, and resources for different skill types and voice interaction models.\n", "pub_date": "2024-10-10T10:11:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Alexa Skill and Apex - has anyone tried this yet? - Oracle Forums", "url": "https://forums.oracle.com/ords/apexds/post/alexa-skill-and-apex-has-anyone-tried-this-yet-6036", "text": "\nHas anyone else on the forum tried any development with Apex and Voice Search (Alexa, Siri etc) I am planning on creating an Alexa skill to query a sample Oracle database using APEX and would be interested to learn if anyone has done this, and any advice/guidance they could share. The only example I could find is from Christoph Ruepprich\n", "pub_date": "2024-08-26T01:13:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Querying an Oracle Database Using Amazon Alexa - DSP", "url": "https://content.dsp.co.uk/apex/querying-an-oracle-database-using-amazon-alexa", "text": "\nThis blog will show you how to get up and running with this and create an Alexa skill that queries your Oracle database. There are three main building blocks needed to get this working in its simplest form: ORDS Web Service (configured either in APEX or on the database). AWS Lambda function (anonymous function). Alexa Skill.\n", "pub_date": "2024-10-06T23:28:00.0000000Z", "extra": null}, {"type": "webpage", "title": "How to Build Your First Alexa Skill: 5 Steps to Get Started", "url": "https://developer.amazon.com/en-US/blogs/alexa/post/a1080dd9-2a28-46b8-8fce-1dfd556b50ff/how-to-build-your-first-alexa-skill-5-steps-to-get-starte", "text": "\nAlexa Skill Builder&#x27;s Guides; Connect with Alexa Evangelists on Twitch; Build Your First Skill, Get a T-Shirt. Bring your big idea to life with Alexa and earn perks through our milestone-based developer promotion. US developers, publish your first Alexa skill and earn a custom Alexa developer t-shirt.\n", "pub_date": "2024-10-11T12:02:00.0000000Z", "extra": null}, {"type": "webpage", "title": "La skill Alexa &quot;Oracolo di Apollo&quot; per parlare con ChatGPT", "url": "https://www.youtube.com/watch?v=D6gGA5NkNp4", "text": "\n\ud83c\udf1f Prova la skill Oracolo di Apollo! \ud83c\udf10 Immergiti in conversazioni illimitate con ChatGPT su Alexa. Dalla scienza alla poesia, conosce tutto. \ud83d\udcda NOVIT\u00c0: In...\n", "pub_date": "2024-01-02T18:31:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Alexa Skills - Amazon Customer Service", "url":

"https://www.amazon.com/gp/help/customer/display.html?nodeId=G8QQCQL47RFVGAZC", "text":
"\nLearn how to use Alexa Skills, which are like apps that you can enable to help you
perform more voice-activated actions. Find out how to turn skills on or off,
troubleshoot issues, and access in-skill purchases and celebrity personality.\n",
"pub_date": "2024-10-12T13:06:00.0000000Z", "extra": null}, {"type": "webpage",
"title": "chatGPT on top of Amazon Alexa - Page 5 - Community", "url":
"https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=5", "text":
"\nNEW FUNCTIONALITY: \u2192 Orakel sends e-mail to the user I have just added a new
feature to the German skill \u201cOrakel des Lichts\u201d: it is now possible to
receive the current conversation by email. Whenever the user says \u201c&lt;&lt;Schick
mir eine E-Mail&gt;&gt;,\u201d the Orakel will send the conversation to the email
address associated with the user\u2019s Amazon account.\n", "pub_date": "2024-09-
08T17:55:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Oracolo di Apollo
- la skill Alexa che si connette a chatGPT ... - YouTube", "url":
"https://www.youtube.com/watch?v=GAGVssZRBMs", "text": "\nIngaggia l&#x27;Oracolo di
Apollo in una conversazione su un argomento a tua scelta. L&#x27;Oracolo \u00e8 in
grado di rispondere a tutte le tue curiosit\u00e0. Conosce tutte le...\n", "pub_date":
"2024-07-29T05:15:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Oracle
APEX \u2013 Amazon ALEXA integration WITHOUT Amazon ... - Blogger", "url":
"https://apexutil.blogspot.com/2018/03/oracle-apex-amazon-alexa-integration_28.html",
"text": "\nIn the last blog post I prepared Oracle side. In this post I will show, how
to prepare ALEXA Skill and how to integrate all layers. Few weeks ago Amazon
implemented new version of Alexa Skill Kit. Alexa Skill Kit is like a APEX builder.
Low code technology, so you don&#x27;t need to write any code :) Let&#x27;s start ! Go
to Amazon Developer Console 1.\n", "pub_date": "2024-09-04T12:53:00.0000000Z", "extra":
null}, {"type": "webpage", "title": "Talk to Your ERP with ALEXA and ORDS - JMJ CLOUD",
"url": "https://www.jmjcloud.com/blog/talk-to-your-erp-with-alexa-and-ords", "text":
"\nIt passes this speech to the Alexa Voice Service (AVS) in the cloud. Amazon
converts the speech to text and attempts to determine the skill that was requested
based on the invocation name. In the above example, \u2018Ask My Database\u2019. Once
it recognizes the skill, it forms as structured JSON request and sends it to our ORDS
REST end-point.\n", "pub_date": "2024-10-09T02:07:00.0000000Z", "extra": null},
{"type": "webpage", "title": "Apollo Hospitals\u2019 Amazon Alexa Skill to enable
users access 72 ...", "url": "https://www.thehindubusinessline.com/companies/apollo-
hospitals-amazon-alexa-skill-to-enable-users-access-72-hospitals-3500-
pharmacies/article30914910.ece", "text": "\nThe skill can also be accessed via the
Alexa app available on iOS and Android. Users can simply say, \u201cAlexa, open Apollo,
to find me a doctor for consultation, to book an appointment\u201d, among ...\n",
"pub_date": "2024-10-13T06:48:00.0000000Z", "extra": null}, {"type": "webpage",
"title": "What Are Alexa In-Skill Purchases? - Amazon Customer Service", "url":
"https://www.amazon.com/gp/help/customer/display.html?nodeId=GSTUK67FZSNLL5A2", "text":
"\nTypes of in-skill purchases: One-time Purchase - Unlock access to features or
content within a skill. Typically available to use on any compatible Alexa device
registered to your account.\nVisible: 0% - 100%", "pub_date": "2024-10-
10T03:23:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Search results
for query: &#x27;Apollo Oracle Alexa skill&#x27;", "url": "", "text": "#
\u30100\u2020chatGPT on top of Amazon Alexa - Page 9 -
Community\u2020community.openai.com\u3011\nabout the Alexa skill Oracle of Light
Amazon.com: oracle of light : Alexa Skills Advanced Mode. Marty and Lucy discuss the

new AI feature, Oracle of Light, for Amazon\u2019s Alexa (referred to as \u201cLady A\u201d) ... La skill \u00e8 qui: Amazon.it: oracolo di apollo : Alexa Skills.\n# \u30101\u2020chatGPT on top of Amazon Alexa - Community - OpenAI Developer Forum\u2020community.openai.com\u3011\nA user shares several skills for Amazon Alexa that use chatGPT, a text-generation model by OpenAI, to engage in conversations on various topics. Another user asks how to launch the skill without opening a game on Luna app.\n# \u30102\u2020Apollo Alexa Skill - GitHub\u2020github.com\u3011\nApollo Alexa Skill \\n. An AWS Lambda function that bridges Alexa Smart Home commands to the Apollo Home Control application. \\n Purpose \\n. While it would be possible to sent commands directly to Apollo Home Control application, that would require opening up a port inside the home network and managing the security that goes along with that.\n# \u30103\u2020chatGPT on top of Amazon Alexa - Page 8 - Community\u2020community.openai.com\u3011\nThe Alexa skill \u2018Oracle of Light\u2019 is the reference skill on Astro robot. It\u2019s the best way to talk to Astro as chatGPT. ... L\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che dialoga con chatGPT e ti invia la conversazione per mail! sono_matteo October 24, 2023, 8:37pm 152. Latency ...\n# \u30104\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011\nYou need to grant permission to use your email in the Alexa App, by selecting the skill and then its Settings. Engage the Oracle of Light in a conversation on a topic of your choice. The skill connects to the chatGPT language model developed by OpenAI, trained using a large amount of textual data and able to answer questions and provide ...\n# \u30105\u2020raypp2/Apollo-Alexa-Skill - GitHub\u2020github.com\u3011\nWhile it would be possible to sent commands directly to Apollo Home Control application, that would require opening up a port inside the home network and managing the security that goes along with that. Lambda is reasonably fast, free at low execution volumes, and integrates more easily with the ...\n# \u30106\u2020Oracle Digital Assistant TechExchange - Oracle Blogs\u2020blogs.oracle.com\u3011\nConfigure the Alexa Skill Code App With the Alexa skill created in the Amazon developer console, the next step is to connect it with a remote service that implements the task to be executed when the skill is invoked. To connect Alexa with Oracle Digital Assistant, you will be using the code provided in the downloads for this Oracle\n# \u30107\u2020Build Your Skill | Alexa Skills Kit - developer.amazon.com\u2020developer.amazon.com\u3011\nLearn how to create and publish Alexa skills using the Alexa Skills Kit (ASK), which includes development tools and libraries. Find out the prerequisites, features, and resources for different skill types and voice interaction models.\n# \u30108\u2020Alexa Skill and Apex - has anyone tried this yet? - Oracle Forums\u2020forums.oracle.com\u3011\nHas anyone else on the forum tried any development with Apex and Voice Search (Alexa, Siri etc) I am planning on creating an Alexa skill to query a sample Oracle database using APEX and would be interested to learn if anyone has done this, and any advice/guidance they could share. The only example I could find is from Christoph Ruepprich\n# \u30109\u2020Querying an Oracle Database Using Amazon Alexa - DSP\u2020content.dsp.co.uk\u3011\nThis blog will show you how to get up and running with this and create an Alexa skill that queries your Oracle database. There are three main building blocks needed to get this working in its simplest form: ORDS Web Service (configured either in APEX or on the database). AWS Lambda function (anonymous function). Alexa Skill.\n# \u301010\u2020How to Build Your First Alexa Skill: 5 Steps to Get Started\u2020developer.amazon.com\u3011\nAlexa Skill Builder&#x27;s Guides; Connect with Alexa Evangelists on Twitch; Build Your First Skill, Get a T-Shirt. Bring your big idea to life with Alexa and earn perks

```
through our milestone-based developer promotion. US developers, publish your first
Alexa skill and earn a custom Alexa developer t-shirt.\n# \u301011\u2020La skill Alexa
&quot;Oracolo di Apollo&quot; per parlare con
ChatGPT\u2020www.youtube.com\u3011\n\ud83c\udf1f Prova la skill Oracolo di Apollo!
\ud83c\udf10 Immergiti in conversazioni illimitate con ChatGPT su Alexa. Dalla scienza
alla poesia, conosce tutto. \ud83d\udcda NOVIT\u00c0: In...\n# \u301012\u2020Alexa
Skills - Amazon Customer Service\u2020www.amazon.com\u3011\nLearn how to use Alexa
Skills, which are like apps that you can enable to help you perform more voice-
activated actions. Find out how to turn skills on or off, troubleshoot issues, and
access in-skill purchases and celebrity personality.\n# \u301013\u2020chatGPT on top
of Amazon Alexa - Page 5 - Community\u2020community.openai.com\u3011\nNEW
FUNCTIONALITY: \u2192 Orakel sends e-mail to the user I have just added a new feature
to the German skill \u201cOrakel des Lichts\u201d: it is now possible to receive the
current conversation by email. Whenever the user says \u201c&lt;&lt;Schick mir eine E-
Mail&gt;&gt;,\u201d the Orakel will send the conversation to the email address
associated with the user\u2019s Amazon account.\n# \u301014\u2020Oracolo di Apollo -
la skill Alexa che si connette a chatGPT ... -
YouTube\u2020www.youtube.com\u3011\nIngaggia l&#x27;Oracolo di Apollo in una
conversazione su un argomento a tua scelta. L&#x27;Oracolo \u00e8 in grado di
rispondere a tutte le tue curiosit\u00e0. Conosce tutte le...\n# \u301015\u2020Oracle
APEX \u2013 Amazon ALEXA integration WITHOUT Amazon ... -
Blogger\u2020apexutil.blogspot.com\u3011\nIn the last blog post I prepared Oracle side.
In this post I will show, how to prepare ALEXA Skill and how to integrate all layers.
Few weeks ago Amazon implemented new version of Alexa Skill Kit. Alexa Skill Kit is
like a APEX builder. Low code technology, so you don&#x27;t need to write any code :)
Let&#x27;s start ! Go to Amazon Developer Console 1.\n# \u301016\u2020Talk to Your ERP
with ALEXA and ORDS - JMJ CLOUD\u2020www.jmjcloud.com\u3011\nIt passes this speech to
the Alexa Voice Service (AVS) in the cloud. Amazon converts the speech to text and
attempts to determine the skill that was requested based on the invocation name. In
the above example, \u2018Ask My Database\u2019. Once it recognizes the skill, it forms
as structured JSON request and sends it to our ORDS REST end-point.\n#
\u301017\u2020Apollo Hospitals\u2019 Amazon Alexa Skill to enable users access
72 ...\u2020www.thehindubusinessline.com\u3011\nThe skill can also be accessed via the
Alexa app available on iOS and Android. Users can simply say, \u201cAlexa, open Apollo,
to find me a doctor for consultation, to book an appointment\u201d, among ...\n#
\u301018\u2020What Are Alexa In-Skill Purchases? - Amazon Customer
Service\u2020www.amazon.com\u3011\nTypes of in-skill purchases: One-time Purchase -
Unlock access to features or content within a skill. Typically available to use on any
compatible Alexa device registered to your account.\nVisible: 0% - 100%", "pub_date":
null, "extra": null}], "original_query": null}, "command": "search", "args": ["Apollo
Oracle Alexa skill"], "status": "finished", "is_complete": true, "message_type": null,
"model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "455e8fe2-b3e6-
45f9-8398-ac10e1e12382", "request_id": "8d21fc94bb9e0e1a-MXP", "timestamp_":
"absolute"}, "recipient": "all", "channel": null}, "parent": "455e8fe2-b3e6-45f9-8398-
ac10e1e12382", "children": ["75323384-12ba-43f2-af40-e013cd8c84a9"]}, "75323384-12ba-
43f2-af40-e013cd8c84a9": {"id": "75323384-12ba-43f2-af40-e013cd8c84a9", "message":
{"id": "75323384-12ba-43f2-af40-e013cd8c84a9", "author": {"role": "assistant", "name":
null, "metadata": {}}, "create_time": 1728850395.892694, "update_time": null,
"content": {"content_type": "code", "language": "unknown", "response_format_name":
```

```
null, "text": "mclick([11,14,3])"}, "status": "finished_successfully", "end_turn":
false, "weight": 1.0, "metadata": {"citations": [], "content_references": [],
"gizmo_id": null, "finish_details": {"type": "stop", "stop_tokens": [200007]},
"is_complete": true, "message_type": null, "model_slug": "gpt-4o",
"default_model_slug": "gpt-4o", "parent_id": "50e79d24-6dcb-4e30-ab10-bd0a5bcf4ad9",
"request_id": "8d21fc94bb9e0e1a-MXP", "timestamp_": "absolute"}, "recipient":
"browser", "channel": null}, "parent": "50e79d24-6dcb-4e30-ab10-bd0a5bcf4ad9",
"children": ["a134b581-a17a-4055-8462-496b1bfae135"]}, "a134b581-a17a-4055-8462-
496b1bfae135": {"id": "a134b581-a17a-4055-8462-496b1bfae135", "message": {"id":
"a134b581-a17a-4055-8462-496b1bfae135", "author": {"role": "tool", "name": "browser",
"metadata": {}}, "create_time": 1728850395.892768, "update_time": null, "content":
{"content_type": "tether_quote", "url": "https://www.youtube.com/watch?v=GAGVssZRBMs",
"domain": "www.youtube.com", "text": "\nURL:
https://www.youtube.com/watch?v=GAGVssZRBMs\n\u30100\u2020About\u3011\u30101\u2020Pres
s\u3011\u30102\u2020Copyright\u3011\u30103\u2020Contact
us\u3011\u30104\u2020Creators\u3011\u30105\u2020Advertise\u3011\u30106\u2020Developers
\u2020developers.google.com\u3011\u30107\u2020Terms\u3011\u30108\u2020Privacy\u3011\u3
0109\u2020Policy &amp; Safety\u3011\u301010\u2020How YouTube
works\u3011\u301011\u2020Test new features\u3011\u301012\u2020NFL Sunday
Ticket\u2020tv.youtube.com\u3011\n\n\u00a9 2024 Google LLC\nVisible: 0% - 100%",
"title": "Oracolo di Apollo - la skill Alexa che si connette a chatGPT - YouTube",
"tether_id": null}, "status": "finished_successfully", "end_turn": null, "weight": 0.0,
"metadata": {"_cite_metadata": {"citation_format": {"name": "tether_og"},
"metadata_list": [{"type": "webpage", "title": "Oracolo di Apollo - la skill Alexa che
si connette a chatGPT - YouTube", "url": "https://www.youtube.com/watch?v=GAGVssZRBMs",
"text": "\nURL:
https://www.youtube.com/watch?v=GAGVssZRBMs\n\u30100\u2020About\u3011\u30101\u2020Pres
s\u3011\u30102\u2020Copyright\u3011\u30103\u2020Contact
us\u3011\u30104\u2020Creators\u3011\u30105\u2020Advertise\u3011\u30106\u2020Developers
\u2020developers.google.com\u3011\u30107\u2020Terms\u3011\u30108\u2020Privacy\u3011\u3
0109\u2020Policy &amp; Safety\u3011\u301010\u2020How YouTube
works\u3011\u301011\u2020Test new features\u3011\u301012\u2020NFL Sunday
Ticket\u2020tv.youtube.com\u3011\n\n\u00a9 2024 Google LLC\nVisible: 0% - 100%",
"pub_date": null, "extra": null}], "original_query": null}, "command": "mclick",
"args": ["[11, 14, 3]"], "status": "finished", "is_complete": true, "message_type":
null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "75323384-
12ba-43f2-af40-e013cd8c84a9", "request_id": "8d21fc94bb9e0e1a-MXP", "timestamp_":
"absolute"}, "recipient": "all", "channel": null}, "parent": "75323384-12ba-43f2-af40-
e013cd8c84a9", "children": ["645b2141-5c43-48c7-aab1-9817d991c898"]}, "645b2141-5c43-
48c7-aab1-9817d991c898": {"id": "645b2141-5c43-48c7-aab1-9817d991c898", "message":
{"id": "645b2141-5c43-48c7-aab1-9817d991c898", "author": {"role": "tool", "name":
"browser", "metadata": {}}, "create_time": 1728850395.892841, "update_time": null,
"content": {"content_type": "tether_quote", "url":
"https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=8", "domain":
"community.openai.com", "text": "Chaque fois que l\u2019utilisateur dit \u201cEnvoie-
moi un e-mail\u201d, l\u2019Oracle enverra la conversation \u00e0 l\u2019adresse e-
mail associ\u00e9e au compte Amazon de l\u2019utilisateur. [Image
3: :mailbox_with_no_mail:]  \nCela se produit si l\u2019utilisateur a donn\u00e9 la
permission d\u2019utiliser son adresse e-mail. Pour recevoir l\u2019e-mail,
l\u2019utilisateur doit accorder la permission dans l\u2019application
Alexa.\n\n\u30103\u2020sono_matteo\u3011  October 14, 2023, 11:17am
148\n\n\u301011\u2020mail oracle lumiere 732\u00d7335 45.3 KB\u2020global.discourse-
cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011  October 14, 2023, 9:58pm  149\n\nThis
is the new Instagram account for the Alexa skill \u2018Oracle of
Light\u2019:\n\n[Image 4] \u301012\u2020instagram.com\u2020www.instagram.com\u3011
[Image 5] \n\n### \u301012\u2020Oracle of Light (@oracle.of.light.alexa) \u2022
Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following,
1 Posts - See Instagram photos and videos from Oracle of Light
(@oracle.of.light.alexa)\n\nYou will find all the updates! I have plans to connect
Alexa, ChatGPT, and Instagram. Stay tuned!\n\n\u30103\u2020sono_matteo\u3011  October
21, 2023, 2:17pm  150\n\nDies ist das neue Instagram-Konto f\u00fcr die Alexa-Skill
\u2018Orakel des Lichts\u2019:\n\n[Image 6]
\u301013\u2020instagram.com\u2020www.instagram.com\u3011  [Image 7] \n\n###
\u301013\u2020Orakel des Lichts (@orakel.des.lichts) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following, 2 Posts - See
Instagram photos and videos from Orakel des Lichts (@orakel.des.lichts)\n\nDu wirst
alle Updates finden! Ich habe Pl\u00e4ne, Alexa, ChatGPT und Instagram zu verbinden.
Bleib dran!\n\n\u30103\u2020sono_matteo\u3011  October 22, 2023, 11:24am
151\n\nQuesto video sul canale Instagram della skill Alexa \u2018Oracolo di
Apollo\u2019 illustra un impiego tipico della skill.
\n\u301014\u2020https://www.instagram.com/p/CyrTyV8s12z/?igshid=MzRlODBiNWFlZA==\u2020
www.instagram.com\u3011  \nDopo aver conversato con chatGPT, \u00e8 possibile ricevere
per mail la conversazione dicendo inviami una mail.  \nRicevi la conversazione via
mail:  \n\n\u301015\u20201000005176 1080\u00d72400 241 KB\u2020global.discourse-
cdn.com\u3011\n\nL\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che dialoga
con chatGPT e ti invia la conversazione per mail!\n\n\u30103\u2020sono_matteo\u3011
October 24, 2023, 8:37pm  152\n\n\u301016\u2020Latency - 2023-10-24 203507
1200\u00d7800 40.2 KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of
the Alexa skill \u2018Oracle of Light\u2019 over the past month.  \nThe skill is here:
\u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011
\nPerformance for a conversation is good when the latency is around 3
sec.\n\n\u30103\u2020sono_matteo\u3011  October 26, 2023, 9:55pm  153\n\nThis video on
the Instagram channel of the Alexa skill \u2018Oracle of Light\u2019 illustrates a
typical use of the skill:\n\n[Image 8]
\u301018\u2020Instagram\u2020www.instagram.com\u3011  [Image 9] \n\n###
\u301018\u2020Oracle of Light on Instagram: &quot;Ask the Alexa skill &#x27;Oracle of
Light&#x27; any...\u2020www.instagram.com\u3011\n\n1 likes, 2 comments -
oracle.of.light.alexa on October 28, 2023: &quot;Ask the Alexa skill &#x27;Oracle of
Light&#x27; any recipe. You&#x27;ll get it directly from chatGPT. Save the
c...&quot;\n\nAfter chatting with ChatGPT, you can receive the conversation via email
by saying send me an email.  \nYou will receive the conversation via email:
\n\n\u301019\u2020Screenshot_20231026_224828_Gmail 1080\u00d72400 303
KB\u2020global.discourse-cdn.com\u3011\n\nThe Oracle of Light is the only Alexa skill
that interacts with ChatGPT and sends you the conversation via
email!\n\n\u30103\u2020sono_matteo\u3011  October 29, 2023, 9:13am  154\n\nNUOVA
FUNZIONALIT\u00c0: \u2192 La skill Oracolo di Apollo invia un\u2019email
all\u2019utente [Image 10: :mailbox:] [Image 11: :trumpet:]\n\nHo appena aggiunto una
nuova funzionalit\u00e0 alla skill italiana \u201cOracolo di Apollo\u201d: ora \u00e8
```

possibile ricevere la conversazione in corso via email. \nOgni volta che
l\u2019utente dice \u201cinviami una mail\u201d, l\u2019Oracolo invier\u00e0 la
conversazione all\u2019indirizzo email associato all\u2019account Amazon
dell\u2019utente. [Image 12: :love_letter:] \nCi\u00f2 avviene solo se l\u2019utente
ha dato il permesso di utilizzare il suo indirizzo email. Per ricevere l\u2019email,
l\u2019utente deve concedere il permesso nell\u2019applicazione Alexa.
\n\n\u301020\u2020mail 732\u00d7335 38.7 KB\u2020global.discourse-
cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 31, 2023, 6:35pm
155\n\nVoici le nouveau compte Instagram pour la skill Alexa \u2018Oracle de
Lumi\u00e8re\u2019\n\n[Image 13]
\u301021\u2020instagram.com\u2020www.instagram.com\u3011 [Image 14] \n\n###
\u301021\u2020Oracle de Lumi\u00e8re (@oracle.de.lumiere) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See
Instagram photos and videos from Oracle de Lumi\u00e8re (@oracle.de.lumiere)\n\nVous
trouverez toutes les mises \u00e0 jour ! J\u2019ai des projets pour connecter Alexa,
ChatGPT et Instagram. Restez \u00e0
l\u2019\u00e9coute !\n\n\u30103\u2020sono_matteo\u3011 November 1, 2023, 2:49pm
156\n\nCette vid\u00e9o sur le canal Instagram de la skill Alexa \u2018Oracle de
Lumi\u00e8re\u2019 illustre une utilisation typique de la skill:\n\n[Image 15]
\u301022\u2020Instagram\u2020www.instagram.com\u3011 [Image 16] \n\n###
\u301022\u2020Oracle de Lumi\u00e8re on Instagram: &quot;Demande \u00e0 la skill
Oracle de Lumi\u00e8re la...\u2020www.instagram.com\u3011\n\n1 likes, 1 comments -
oracle.de.lumiere on November 1, 2023: &quot;Demande \u00e0 la skill Oracle de
Lumi\u00e8re la signification de chaque \u0153uvre litt\u00e9raire. Demande
\u00e0 ...&quot;\n\nApr\u00e8s avoir discut\u00e9 avec ChatGPT, vous pouvez recevoir
la conversation par e-mail en disant envoie-moi un e-mail. Vous recevrez la
conversation par e-mail:\n\n\u301023\u20206_Screenshot_20231101_115912_Gmail
1080\u00d72400 386 KB\u2020global.discourse-cdn.com\u3011\n\nL\u2019Oracle de
Lumi\u00e8re est la seule skill Alexa qui interagit avec ChatGPT et vous envoie la
conversation par e-mail !\n\n\u30103\u2020sono_matteo\u3011 November 4, 2023, 7:55am
157\n\n\u301024\u2020Latency - 2023-11-04 075224 1200\u00d7600 30.1
KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of the Alexa skill
\u2018Oracle of Light\u2019 over the past month. \nThe skill is here:
\u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011
\nToday the performance for a conversation is good. Latency should stay around 3
sec.\n\n\u30103\u2020sono_matteo\u3011 November 11, 2023, 10:21am 158\n\nI\u2019m
planning to create a version of each skill using GPT4 Turbo. But the price is still
higher than GPT3.5. What is your opinion?\n\n\u30103\u2020sono_matteo\u3011 November
12, 2023, 8:58am 159\n\nEsta es la nueva cuenta de Instagram para la skill de Alexa
\u2018Or\u00e1culo de Luz\u2019:\n\n[Image 17]
\u301025\u2020instagram.com\u2020www.instagram.com\u3011 [Image 18] \n\n###
\u301025\u2020Or\u00e1culo de Luz (@oraculo.de.luz.alexa) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See
Instagram photos and videos from Or\u00e1culo de Luz
(@oraculo.de.luz.alexa)\n\n\u00a1Encontrar\u00e1n todas las novedades! Tengo en mente
conectar Alexa, chatGPT e Instagram. \u00a1Mant\u00e9nganse
atentos!\n\n\u30103\u2020sono_matteo\u3011 November 12, 2023, 9:42pm 160\n\nDieses
Video auf dem Instagram-Kanal der Alexa-Skill \u2018Orakel des Lichts\u2019 zeigt eine
typische Anwendung der Skill.\n\n[Image 19] \u30102", "title": "chatGPT on top of

Amazon Alexa - Page 8 - Community - OpenAI Developer Forum", "tether_id": null}, "status": "finished_successfully", "end_turn": null, "weight": 0.0, "metadata": {"_cite_metadata": {"citation_format": {"name": "tether_og"}, "metadata_list": [{"type": "webpage", "title": "chatGPT on top of Amazon Alexa - Page 8 - Community - OpenAI Developer Forum", "url": "https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=8", "text": "Chaque fois que l\u2019utilisateur dit \u201cEnvoie-moi un e-mail\u201d, l\u2019Oracle enverra la conversation \u00e0 l\u2019adresse e-mail associ\u00e9e au compte Amazon de l\u2019utilisateur. [Image 3: :mailbox_with_no_mail:] \nCela se produit si l\u2019utilisateur a donn\u00e9 la permission d\u2019utiliser son adresse e-mail. Pour recevoir l\u2019e-mail, l\u2019utilisateur doit accorder la permission dans l\u2019application Alexa.\n\n\u30103\u2020sono_matteo\u3011 October 14, 2023, 11:17am 148\n\n\u301011\u2020mail oracle lumiere 732\u00d7335 45.3 KB\u2020global.discourse-cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 14, 2023, 9:58pm 149\n\nThis is the new Instagram account for the Alexa skill \u2018Oracle of Light\u2019:\n\n[Image 4] \u301012\u2020instagram.com\u2020www.instagram.com\u3011 [Image 5] \n\n### \u301012\u2020Oracle of Light (@oracle.of.light.alexa) \u2022 Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following, 1 Posts - See Instagram photos and videos from Oracle of Light (@oracle.of.light.alexa)\n\nYou will find all the updates! I have plans to connect Alexa, ChatGPT, and Instagram. Stay tuned!\n\n\u30103\u2020sono_matteo\u3011 October 21, 2023, 2:17pm 150\n\nDies ist das neue Instagram-Konto f\u00fcr die Alexa-Skill \u2018Orakel des Lichts\u2019:\n\n[Image 6] \u301013\u2020instagram.com\u2020www.instagram.com\u3011 [Image 7] \n\n### \u301013\u2020Orakel des Lichts (@orakel.des.lichts) \u2022 Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following, 2 Posts - See Instagram photos and videos from Orakel des Lichts (@orakel.des.lichts)\n\nDu wirst alle Updates finden! Ich habe Pl\u00e4ne, Alexa, ChatGPT und Instagram zu verbinden. Bleib dran!\n\n\u30103\u2020sono_matteo\u3011 October 22, 2023, 11:24am 151\n\nQuesto video sul canale Instagram della skill Alexa \u2018Oracolo di Apollo\u2019 illustra un impiego tipico della skill. \n\u301014\u2020https://www.instagram.com/p/CyrTyV8s12z/?igshid=MzRlODBiNWFlZA==\u2020www.instagram.com\u3011 \nDopo aver conversato con chatGPT, \u00e8 possibile ricevere per mail la conversazione dicendo inviami una mail. \nRicevi la conversazione via mail: \n\n\u301015\u20201000005176 1080\u00d72400 241 KB\u2020global.discourse-cdn.com\u3011\n\nL\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che dialoga con chatGPT e ti invia la conversazione per mail!\n\n\u30103\u2020sono_matteo\u3011 October 24, 2023, 8:37pm 152\n\n\u301016\u2020Latency - 2023-10-24 203507 1200\u00d7800 40.2 KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of the Alexa skill \u2018Oracle of Light\u2019 over the past month. \nThe skill is here: \u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011 \nPerformance for a conversation is good when the latency is around 3 sec.\n\n\u30103\u2020sono_matteo\u3011 October 26, 2023, 9:55pm 153\n\nThis video on the Instagram channel of the Alexa skill \u2018Oracle of Light\u2019 illustrates a typical use of the skill:\n\n[Image 8] \u301018\u2020Instagram\u2020www.instagram.com\u3011 [Image 9] \n\n### \u301018\u2020Oracle of Light on Instagram: &quot;Ask the Alexa skill &#x27;Oracle of Light&#x27; any...\u2020www.instagram.com\u3011\n\n1 likes, 2 comments - oracle.of.light.alexa on October 28, 2023: &quot;Ask the Alexa skill &#x27;Oracle of

Light&#x27; any recipe. You&#x27;ll get it directly from chatGPT. Save the c...&quot;\n\nAfter chatting with ChatGPT, you can receive the conversation via email by saying send me an email. \nYou will receive the conversation via email: \n\n\u301019\u2020Screenshot_20231026_224828_Gmail 1080\u00d72400 303 KB\u2020global.discourse-cdn.com\u3011\n\nThe Oracle of Light is the only Alexa skill that interacts with ChatGPT and sends you the conversation via email!\n\n\u30103\u2020sono_matteo\u3011 October 29, 2023, 9:13am 154\n\nNUOVA FUNZIONALIT\u00c0: \u2192 La skill Oracolo di Apollo invia un\u2019email all\u2019utente [Image 10: :mailbox:] [Image 11: :trumpet:]\n\nHo appena aggiunto una nuova funzionalit\u00e0 alla skill italiana \u201cOracolo di Apollo\u201d: ora \u00e8 possibile ricevere la conversazione in corso via email. \nOgni volta che l\u2019utente dice \u201cinviami una mail\u201d, l\u2019Oracolo invier\u00e0 la conversazione all\u2019indirizzo email associato all\u2019account Amazon dell\u2019utente. [Image 12: :love_letter:] \nCi\u00f2 avviene solo se l\u2019utente ha dato il permesso di utilizzare il suo indirizzo email. Per ricevere l\u2019email, l\u2019utente deve concedere il permesso nell\u2019applicazione Alexa. \n\n\u301020\u2020mail 732\u00d7335 38.7 KB\u2020global.discourse-cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 31, 2023, 6:35pm 155\n\nVoici le nouveau compte Instagram pour la skill Alexa \u2018Oracle de Lumi\u00e8re\u2019\n\n[Image 13] \u301021\u2020instagram.com\u2020www.instagram.com\u3011 [Image 14] \n\n### \u301021\u2020Oracle de Lumi\u00e8re (@oracle.de.lumiere) \u2022 Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See Instagram photos and videos from Oracle de Lumi\u00e8re (@oracle.de.lumiere)\n\nVous trouverez toutes les mises \u00e0 jour ! J\u2019ai des projets pour connecter Alexa, ChatGPT et Instagram. Restez \u00e0 l\u2019\u00e9coute !\n\n\u30103\u2020sono_matteo\u3011 November 1, 2023, 2:49pm 156\n\nCette vid\u00e9o sur le canal Instagram de la skill Alexa \u2018Oracle de Lumi\u00e8re\u2019 illustre une utilisation typique de la skill:\n\n[Image 15] \u301022\u2020Instagram\u2020www.instagram.com\u3011 [Image 16] \n\n### \u301022\u2020Oracle de Lumi\u00e8re on Instagram: &quot;Demande \u00e0 la skill Oracle de Lumi\u00e8re la...\u2020www.instagram.com\u3011\n\n1 likes, 1 comments - oracle.de.lumiere on November 1, 2023: &quot;Demande \u00e0 la skill Oracle de Lumi\u00e8re la signification de chaque \u0153uvre litt\u00e9raire. Demande \u00e0 ...&quot;\n\nApr\u00e8s avoir discut\u00e9 avec ChatGPT, vous pouvez recevoir la conversation par e-mail en disant envoie-moi un e-mail. Vous recevrez la conversation par e-mail:\n\n\u301023\u20206_Screenshot_20231101_115912_Gmail 1080\u00d72400 386 KB\u2020global.discourse-cdn.com\u3011\n\nL\u2019Oracle de Lumi\u00e8re est la seule skill Alexa qui interagit avec ChatGPT et vous envoie la conversation par e-mail !\n\n\u30103\u2020sono_matteo\u3011 November 4, 2023, 7:55am 157\n\n\u301024\u2020Latency - 2023-11-04 075224 1200\u00d7600 30.1 KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of the Alexa skill \u2018Oracle of Light\u2019 over the past month. \nThe skill is here: \u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011 \nToday the performance for a conversation is good. Latency should stay around 3 sec.\n\n\u30103\u2020sono_matteo\u3011 November 11, 2023, 10:21am 158\n\nI\u2019m planning to create a version of each skill using GPT4 Turbo. But the price is still higher than GPT3.5. What is your opinion?\n\n\u30103\u2020sono_matteo\u3011 November 12, 2023, 8:58am 159\n\nEsta es la nueva cuenta de Instagram para la skill de Alexa

\u2018Or\u00e1culo de Luz\u2019:\n\n[Image 17]
\u301025\u2020instagram.com\u2020www.instagram.com\u3011 [Image 18] \n\n###
\u301025\u2020Or\u00e1culo de Luz (@oraculo.de.luz.alexa) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See
Instagram photos and videos from Or\u00e1culo de Luz
(@oraculo.de.luz.alexa)\n\n\u00a1Encontrar\u00e1n todas las novedades! Tengo en mente
conectar Alexa, chatGPT e Instagram. \u00a1Mant\u00e9nganse
atentos!\n\n\u30103\u2020sono_matteo\u3011 November 12, 2023, 9:42pm 160\n\nDieses
Video auf dem Instagram-Kanal der Alexa-Skill \u2018Orakel des Lichts\u2019 zeigt eine
typische Anwendung der Skill.\n\n[Image 19] \u30102", "pub_date": null, "extra":
null}], "original_query": null}, "command": "mclick", "args": ["[11, 14, 3]"],
"status": "finished", "is_complete": true, "message_type": null, "model_slug": "gpt-
4o", "default_model_slug": "gpt-4o", "parent_id": "a134b581-a17a-4055-8462-
496b1bfae135", "request_id": "8d21fc94bb9e0e1a-MXP", "timestamp_": "absolute"},
"recipient": "all", "channel": null}, "parent": "a134b581-a17a-4055-8462-496b1bfae135",
"children": ["ebe1c031-d409-4c7b-820f-9c60a4842bcf"]}, "ebe1c031-d409-4c7b-820f-
9c60a4842bcf": {"id": "ebe1c031-d409-4c7b-820f-9c60a4842bcf", "message": {"id":
"ebe1c031-d409-4c7b-820f-9c60a4842bcf", "author": {"role": "assistant", "name": null,
"metadata": {}}, "create_time": 1728850395.892915, "update_time": null, "content":
{"content_type": "text", "parts": ["Apollo Oracle seems to be a skill for Alexa that
integrates with ChatGPT to allow users to have interactive conversations with the AI
via voice commands. It processes various types of queries through Alexa and can even
send the conversation logs to users via email. This integration works similarly to
other Alexa-based skills that interact with OpenAI\u2019s language models, enabling
voice-driven conversations with GPT.\n\nIt is also linked to some other Alexa skills,
such as Oracle of Light, which offer similar functionalities like interacting with
ChatGPT and emailing conversation logs, as seen in various user
discussions\u3010159\u2020source\u3011\u3010160\u2020source\u3011.\n\nTo find more
detailed information about Apollo Oracle, you could try searching for it on
Alexa\u2019s skill store, YouTube, or developer forums where these integrations are
discussed. If you&#x27;re particularly interested in how it works or how it&#x27;s
built, looking into these developer resources or even trying the skill through your
Alexa-enabled device could provide more insight."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [{"start_ix": 623, "end_ix": 635,
"citation_format_type": "tether_og", "metadata": {"type": "webpage", "title": "Oracolo
di Apollo - la skill Alexa che si connette a chatGPT - YouTube", "url":
"https://www.youtube.com/watch?v=GAGVssZRBMs#:~:text=URL%3A%20https%3A%2F%2Fwww,100",
"text": "\nURL:
https://www.youtube.com/watch?v=GAGVssZRBMs\n\u30100\u2020About\u3011\u30101\u2020Pres
s\u3011\u30102\u2020Copyright\u3011\u30103\u2020Contact
us\u3011\u30104\u2020Creators\u3011\u30105\u2020Advertise\u3011\u30106\u2020Developers
\u2020developers.google.com\u3011\u30107\u2020Terms\u3011\u30108\u2020Privacy\u3011\u3
0109\u2020Policy &amp; Safety\u3011\u301010\u2020How YouTube
works\u3011\u301011\u2020Test new features\u3011\u301012\u2020NFL Sunday
Ticket\u2020tv.youtube.com\u3011\n\n\u00a9 2024 Google LLC\nVisible: 0% - 100%",
"pub_date": null, "extra": {"cited_message_idx": 159, "search_result_idx": null,
"evidence_text": "source", "cloud_doc_url": null}}}, {"start_ix": 635, "end_ix": 647,
"citation_format_type": "tether_og", "metadata": {"type": "webpage", "title": "chatGPT

on top of Amazon Alexa - Page 8 - Community - OpenAI Developer Forum", "url":
"https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=8", "text":
"Chaque fois que l\u2019utilisateur dit \u201cEnvoie-moi un e-mail\u201d,
l\u2019Oracle enverra la conversation \u00e0 l\u2019adresse e-mail associ\u00e9e au
compte Amazon de l\u2019utilisateur. [Image 3: :mailbox_with_no_mail:] \nCela se
produit si l\u2019utilisateur a donn\u00e9 la permission d\u2019utiliser son adresse
e-mail. Pour recevoir l\u2019e-mail, l\u2019utilisateur doit accorder la permission
dans l\u2019application Alexa.\n\n\u30103\u2020sono_matteo\u3011 October 14, 2023,
11:17am 148\n\n\u301011\u2020mail oracle lumiere 732\u00d7335 45.3
KB\u2020global.discourse-cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 14,
2023, 9:58pm 149\n\nThis is the new Instagram account for the Alexa skill
\u2018Oracle of Light\u2019:\n\n[Image 4]
\u301012\u2020instagram.com\u2020www.instagram.com\u3011 [Image 5] \n\n###
\u301012\u2020Oracle of Light (@oracle.of.light.alexa) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following, 1 Posts - See
Instagram photos and videos from Oracle of Light (@oracle.of.light.alexa)\n\nYou will
find all the updates! I have plans to connect Alexa, ChatGPT, and Instagram. Stay
tuned!\n\n\u30103\u2020sono_matteo\u3011 October 21, 2023, 2:17pm 150\n\nDies ist
das neue Instagram-Konto f\u00fcr die Alexa-Skill \u2018Orakel des
Lichts\u2019:\n\n[Image 6] \u301013\u2020instagram.com\u2020www.instagram.com\u3011
[Image 7] \n\n### \u301013\u2020Orakel des Lichts (@orakel.des.lichts) \u2022
Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following,
2 Posts - See Instagram photos and videos from Orakel des Lichts
(@orakel.des.lichts)\n\nDu wirst alle Updates finden! Ich habe Pl\u00e4ne, Alexa,
ChatGPT und Instagram zu verbinden. Bleib dran!\n\n\u30103\u2020sono_matteo\u3011
October 22, 2023, 11:24am 151\n\nQuesto video sul canale Instagram della skill Alexa
\u2018Oracolo di Apollo\u2019 illustra un impiego tipico della skill.
\n\u301014\u2020https://www.instagram.com/p/CyrTyV8s12z/?igshid=MzRlODBiNWFlZA==\u2020
www.instagram.com\u3011 \nDopo aver conversato con chatGPT, \u00e8 possibile ricevere
per mail la conversazione dicendo inviami una mail. \nRicevi la conversazione via
mail: \n\n\u301015\u20201000005176 1080\u00d72400 241 KB\u2020global.discourse-
cdn.com\u3011\n\nL\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che dialoga
con chatGPT e ti invia la conversazione per mail!\n\n\u30103\u2020sono_matteo\u3011
October 24, 2023, 8:37pm 152\n\n\u301016\u2020Latency - 2023-10-24 203507
1200\u00d7800 40.2 KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of
the Alexa skill \u2018Oracle of Light\u2019 over the past month. \nThe skill is here:
\u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011
\nPerformance for a conversation is good when the latency is around 3
sec.\n\n\u30103\u2020sono_matteo\u3011 October 26, 2023, 9:55pm 153\n\nThis video on
the Instagram channel of the Alexa skill \u2018Oracle of Light\u2019 illustrates a
typical use of the skill:\n\n[Image 8]
\u301018\u2020Instagram\u2020www.instagram.com\u3011 [Image 9] \n\n###
\u301018\u2020Oracle of Light on Instagram: &quot;Ask the Alexa skill &#x27;Oracle of
Light&#x27; any...\u2020www.instagram.com\u3011\n\n1 likes, 2 comments -
oracle.of.light.alexa on October 28, 2023: &quot;Ask the Alexa skill &#x27;Oracle of
Light&#x27; any recipe. You&#x27;ll get it directly from chatGPT. Save the
c...&quot;\n\nAfter chatting with ChatGPT, you can receive the conversation via email
by saying send me an email. \nYou will receive the conversation via email:
\n\n\u301019\u2020Screenshot_20231026_224828_Gmail 1080\u00d72400 303

KB\u2020global.discourse-cdn.com\u3011\n\nThe Oracle of Light is the only Alexa skill that interacts with ChatGPT and sends you the conversation via email!\n\n\u30103\u2020sono_matteo\u3011 October 29, 2023, 9:13am 154\n\nNUOVA FUNZIONALIT\u00c0: \u2192 La skill Oracolo di Apollo invia un\u2019email all\u2019utente [Image 10: :mailbox:] [Image 11: :trumpet:]\n\nHo appena aggiunto una nuova funzionalit\u00e0 alla skill italiana \u201cOracolo di Apollo\u201d: ora \u00e8 possibile ricevere la conversazione in corso via email. \nOgni volta che l\u2019utente dice \u201cinviami una mail\u201d, l\u2019Oracolo invier\u00e0 la conversazione all\u2019indirizzo email associato all\u2019account Amazon dell\u2019utente. [Image 12: :love_letter:] \nCi\u00f2 avviene solo se l\u2019utente ha dato il permesso di utilizzare il suo indirizzo email. Per ricevere l\u2019email, l\u2019utente deve concedere il permesso nell\u2019applicazione Alexa. \n\n\u301020\u2020mail 732\u00d7335 38.7 KB\u2020global.discourse-cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 31, 2023, 6:35pm 155\n\nVoici le nouveau compte Instagram pour la skill Alexa \u2018Oracle de Lumi\u00e8re\u2019\n\n[Image 13] \u301021\u2020instagram.com\u2020www.instagram.com\u3011 [Image 14] \n\n### \u301021\u2020Oracle de Lumi\u00e8re (@oracle.de.lumiere) \u2022 Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See Instagram photos and videos from Oracle de Lumi\u00e8re (@oracle.de.lumiere)\n\nVous trouverez toutes les mises \u00e0 jour ! J\u2019ai des projets pour connecter Alexa, ChatGPT et Instagram. Restez \u00e0 l\u2019\u00e9coute !\n\n\u30103\u2020sono_matteo\u3011 November 1, 2023, 2:49pm 156\n\nCette vid\u00e9o sur le canal Instagram de la skill Alexa \u2018Oracle de Lumi\u00e8re\u2019 illustre une utilisation typique de la skill:\n\n[Image 15] \u301022\u2020Instagram\u2020www.instagram.com\u3011 [Image 16] \n\n### \u301022\u2020Oracle de Lumi\u00e8re on Instagram: &quot;Demande \u00e0 la skill Oracle de Lumi\u00e8re la...\u2020www.instagram.com\u3011\n\n1 likes, 1 comments - oracle.de.lumiere on November 1, 2023: &quot;Demande \u00e0 la skill Oracle de Lumi\u00e8re la signification de chaque \u0153uvre litt\u00e9raire. Demande \u00e0 ...&quot;\n\nApr\u00e8s avoir discut\u00e9 avec ChatGPT, vous pouvez recevoir la conversation par e-mail en disant envoie-moi un e-mail. Vous recevrez la conversation par e-mail:\n\n\u301023\u20206_Screenshot_20231101_115912_Gmail 1080\u00d72400 386 KB\u2020global.discourse-cdn.com\u3011\n\nL\u2019Oracle de Lumi\u00e8re est la seule skill Alexa qui interagit avec ChatGPT et vous envoie la conversation par e-mail !\n\n\u30103\u2020sono_matteo\u3011 November 4, 2023, 7:55am 157\n\n\u301024\u2020Latency - 2023-11-04 075224 1200\u00d7600 30.1 KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of the Alexa skill \u2018Oracle of Light\u2019 over the past month. \nThe skill is here: \u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011 \nToday the performance for a conversation is good. Latency should stay around 3 sec.\n\n\u30103\u2020sono_matteo\u3011 November 11, 2023, 10:21am 158\n\nI\u2019m planning to create a version of each skill using GPT4 Turbo. But the price is still higher than GPT3.5. What is your opinion?\n\n\u30103\u2020sono_matteo\u3011 November 12, 2023, 8:58am 159\n\nEsta es la nueva cuenta de Instagram para la skill de Alexa \u2018Or\u00e1culo de Luz\u2019:\n\n[Image 17] \u301025\u2020instagram.com\u2020www.instagram.com\u3011 [Image 18] \n\n### \u301025\u2020Or\u00e1culo de Luz (@oraculo.de.luz.alexa) \u2022 Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See

Instagram photos and videos from Or\u00e1culo de Luz
(@oraculo.de.luz.alexa)\n\n\u00a1Encontrar\u00e1n todas las novedades! Tengo en mente
conectar Alexa, chatGPT e Instagram. \u00a1Mant\u00e9nganse
atentos!\n\n\u30103\u2020sono_matteo\u3011 November 12, 2023, 9:42pm 160\n\nDieses
Video auf dem Instagram-Kanal der Alexa-Skill \u2018Orakel des Lichts\u2019 zeigt eine
typische Anwendung der Skill.\n\n[Image 19] \u30102", "pub_date": null, "extra":
{"cited_message_idx": 160, "search_result_idx": null, "evidence_text": "source",
"cloud_doc_url": null}}}], "content_references": [{"matched_text":
"\u3010159\u2020source\u3011", "start_idx": 623, "end_idx": 635, "alt": null, "type":
"webpage_extended", "title": "Oracolo di Apollo - la skill Alexa che si connette a
chatGPT - YouTube", "url": "https://www.youtube.com/watch?v=GAGVssZRBMs", "pub_date":
null, "snippet": "\nURL:
https://www.youtube.com/watch?v=GAGVssZRBMs\n\u30100\u2020About\u3011\u30101\u2020Pres
s\u3011\u30102\u2020Copyright\u3011\u30103\u2020Contact
us\u3011\u30104\u2020Creators\u3011\u30105\u2020Advertise\u3011\u30106\u2020Developers
\u2020developers.google.com\u3011\u30107\u2020Terms\u3011\u30108\u2020Privacy\u3011\u3
0109\u2020Policy &amp; Safety\u3011\u301010\u2020How YouTube
works\u3011\u301011\u2020Test new features\u3011\u301012\u2020NFL Sunday
Ticket\u2020tv.youtube.com\u3011\n\n\u00a9 2024 Google LLC\nVisible: 0% - 100%"},
{"matched_text": "\u3010160\u2020source\u3011", "start_idx": 635, "end_idx": 647,
"alt": null, "type": "webpage_extended", "title": "chatGPT on top of Amazon Alexa -
Page 8 - Community - OpenAI Developer Forum", "url":
"https://community.openai.com/t/chatgpt-on-top-of-amazon-alexa/58515?page=8",
"pub_date": null, "snippet": "Chaque fois que l\u2019utilisateur dit \u201cEnvoie-moi
un e-mail\u201d, l\u2019Oracle enverra la conversation \u00e0 l\u2019adresse e-mail
associ\u00e9e au compte Amazon de l\u2019utilisateur. [Image 3: :mailbox_with_no_mail:]
\nCela se produit si l\u2019utilisateur a donn\u00e9 la permission d\u2019utiliser son
adresse e-mail. Pour recevoir l\u2019e-mail, l\u2019utilisateur doit accorder la
permission dans l\u2019application Alexa.\n\n\u30103\u2020sono_matteo\u3011 October
14, 2023, 11:17am 148\n\n\u301011\u2020mail oracle lumiere 732\u00d7335 45.3
KB\u2020global.discourse-cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 14,
2023, 9:58pm 149\n\nThis is the new Instagram account for the Alexa skill
\u2018Oracle of Light\u2019:\n\n[Image 4]
\u301012\u2020instagram.com\u2020www.instagram.com\u3011 [Image 5] \n\n###
\u301012\u2020Oracle of Light (@oracle.of.light.alexa) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following, 1 Posts - See
Instagram photos and videos from Oracle of Light (@oracle.of.light.alexa)\n\nYou will
find all the updates! I have plans to connect Alexa, ChatGPT, and Instagram. Stay
tuned!\n\n\u30103\u2020sono_matteo\u3011 October 21, 2023, 2:17pm 150\n\nDies ist
das neue Instagram-Konto f\u00fcr die Alexa-Skill \u2018Orakel des
Lichts\u2019:\n\n[Image 6] \u301013\u2020instagram.com\u2020www.instagram.com\u3011
[Image 7] \n\n### \u301013\u2020Orakel des Lichts (@orakel.des.lichts) \u2022
Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 1 Following,
2 Posts - See Instagram photos and videos from Orakel des Lichts
(@orakel.des.lichts)\n\nDu wirst alle Updates finden! Ich habe Pl\u00e4ne, Alexa,
ChatGPT und Instagram zu verbinden. Bleib dran!\n\n\u30103\u2020sono_matteo\u3011
October 22, 2023, 11:24am 151\n\nQuesto video sul canale Instagram della skill Alexa
\u2018Oracolo di Apollo\u2019 illustra un impiego tipico della skill.
\n\u301014\u2020https://www.instagram.com/p/CyrTyV8s12z/?igshid=MzRlODBiNWFlZA==\u2020

www.instagram.com\u3011 \nDopo aver conversato con chatGPT, \u00e8 possibile ricevere per mail la conversazione dicendo inviami una mail. \nRicevi la conversazione via mail: \n\n\u301015\u20201000005176 1080\u00d72400 241 KB\u2020global.discourse-cdn.com\u3011\n\nL\u2019Oracolo di Apollo \u00e8 l\u2019unica skill Alexa che dialoga con chatGPT e ti invia la conversazione per mail!\n\n\u30103\u2020sono_matteo\u3011 October 24, 2023, 8:37pm 152\n\n\u301016\u2020Latency - 2023-10-24 203507 1200\u00d7800 40.2 KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of the Alexa skill \u2018Oracle of Light\u2019 over the past month. \nThe skill is here: \u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011 \nPerformance for a conversation is good when the latency is around 3 sec.\n\n\u30103\u2020sono_matteo\u3011 October 26, 2023, 9:55pm 153\n\nThis video on the Instagram channel of the Alexa skill \u2018Oracle of Light\u2019 illustrates a typical use of the skill:\n\n[Image 8] \u301018\u2020Instagram\u2020www.instagram.com\u3011 [Image 9] \n\n### \u301018\u2020Oracle of Light on Instagram: &quot;Ask the Alexa skill &#x27;Oracle of Light&#x27; any...\u2020www.instagram.com\u3011\n\n1 likes, 2 comments - oracle.of.light.alexa on October 28, 2023: &quot;Ask the Alexa skill &#x27;Oracle of Light&#x27; any recipe. You&#x27;ll get it directly from chatGPT. Save the c...&quot;\n\nAfter chatting with ChatGPT, you can receive the conversation via email by saying send me an email. \nYou will receive the conversation via email: \n\n\u301019\u2020Screenshot_20231026_224828_Gmail 1080\u00d72400 303 KB\u2020global.discourse-cdn.com\u3011\n\nThe Oracle of Light is the only Alexa skill that interacts with ChatGPT and sends you the conversation via email!\n\n\u30103\u2020sono_matteo\u3011 October 29, 2023, 9:13am 154\n\nNUOVA FUNZIONALIT\u00c0: \u2192 La skill Oracolo di Apollo invia un\u2019email all\u2019utente [Image 10: :mailbox:] [Image 11: :trumpet:]\n\nHo appena aggiunto una nuova funzionalit\u00e0 alla skill italiana \u201cOracolo di Apollo\u201d: ora \u00e8 possibile ricevere la conversazione in corso via email. \nOgni volta che l\u2019utente dice \u201cinviami una mail\u201d, l\u2019Oracolo invier\u00e0 la conversazione all\u2019indirizzo email associato all\u2019account Amazon dell\u2019utente. [Image 12: :love_letter:] \nCi\u00f2 avviene solo se l\u2019utente ha dato il permesso di utilizzare il suo indirizzo email. Per ricevere l\u2019email, l\u2019utente deve concedere il permesso nell\u2019applicazione Alexa. \n\n\u301020\u2020mail 732\u00d7335 38.7 KB\u2020global.discourse-cdn.com\u3011\n\n\u30103\u2020sono_matteo\u3011 October 31, 2023, 6:35pm 155\n\nVoici le nouveau compte Instagram pour la skill Alexa \u2018Oracle de Lumi\u00e8re\u2019\n\n[Image 13] \u301021\u2020instagram.com\u2020www.instagram.com\u3011 [Image 14] \n\n### \u301021\u2020Oracle de Lumi\u00e8re (@oracle.de.lumiere) \u2022 Instagram photos and videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See Instagram photos and videos from Oracle de Lumi\u00e8re (@oracle.de.lumiere)\n\nVous trouverez toutes les mises \u00e0 jour ! J\u2019ai des projets pour connecter Alexa, ChatGPT et Instagram. Restez \u00e0 l\u2019\u00e9coute !\n\n\u30103\u2020sono_matteo\u3011 November 1, 2023, 2:49pm 156\n\nCette vid\u00e9o sur le canal Instagram de la skill Alexa \u2018Oracle de Lumi\u00e8re\u2019 illustre une utilisation typique de la skill:\n\n[Image 15] \u301022\u2020Instagram\u2020www.instagram.com\u3011 [Image 16] \n\n### \u301022\u2020Oracle de Lumi\u00e8re on Instagram: &quot;Demande \u00e0 la skill Oracle de Lumi\u00e8re la...\u2020www.instagram.com\u3011\n\n1 likes, 1 comments -

oracle.de.lumiere on November 1, 2023: &quot;Demande \u00e0 la skill Oracle de
Lumi\u00e8re la signification de chaque \u0153uvre litt\u00e9raire. Demande
\u00e0 ...&quot;\n\nApr\u00e8s avoir discut\u00e9 avec ChatGPT, vous pouvez recevoir
la conversation par e-mail en disant envoie-moi un e-mail. Vous recevrez la
conversation par e-mail:\n\n\u301023\u20206_Screenshot_20231101_115912_Gmail
1080\u00d72400 386 KB\u2020global.discourse-cdn.com\u3011\n\nL\u2019Oracle de
Lumi\u00e8re est la seule skill Alexa qui interagit avec ChatGPT et vous envoie la
conversation par e-mail !\n\n\u30103\u2020sono_matteo\u3011 November 4, 2023, 7:55am
157\n\n\u301024\u2020Latency - 2023-11-04 075224 1200\u00d7600 30.1
KB\u2020global.discourse-cdn.com\u3011\n\nThis is the latency of the Alexa skill
\u2018Oracle of Light\u2019 over the past month. \nThe skill is here:
\u301017\u2020Amazon.com: oracle of light : Alexa Skills\u2020www.amazon.com\u3011
\nToday the performance for a conversation is good. Latency should stay around 3
sec.\n\n\u30103\u2020sono_matteo\u3011 November 11, 2023, 10:21am 158\n\nI\u2019m
planning to create a version of each skill using GPT4 Turbo. But the price is still
higher than GPT3.5. What is your opinion?\n\n\u30103\u2020sono_matteo\u3011 November
12, 2023, 8:58am 159\n\nEsta es la nueva cuenta de Instagram para la skill de Alexa
\u2018Or\u00e1culo de Luz\u2019:\n\n[Image 17]
\u301025\u2020instagram.com\u2020www.instagram.com\u3011 [Image 18] \n\n###
\u301025\u2020Or\u00e1culo de Luz (@oraculo.de.luz.alexa) \u2022 Instagram photos and
videos\u2020www.instagram.com\u3011\n\n0 Followers, 0 Following, 1 Posts - See
Instagram photos and videos from Or\u00e1culo de Luz
(@oraculo.de.luz.alexa)\n\n\u00a1Encontrar\u00e1n todas las novedades! Tengo en mente
conectar Alexa, chatGPT e Instagram. \u00a1Mant\u00e9nganse
atentos!\n\n\u30103\u2020sono_matteo\u3011 November 12, 2023, 9:42pm 160\n\nDieses
Video auf dem Instagram-Kanal der Alexa-Skill \u2018Orakel des Lichts\u2019 zeigt eine
typische Anwendung der Skill.\n\n[Image 19] \u30102"}], "gizmo_id": null,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "645b2141-5c43-48c7-aab1-9817d991c898", "request_id": "8d21fc94bb9e0e1a-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"645b2141-5c43-48c7-aab1-9817d991c898", "children": ["aaa2c545-6ec0-47c0-ba29-
74cf02568f09"]}, "aaa2c545-6ec0-47c0-ba29-74cf02568f09": {"id": "aaa2c545-6ec0-47c0-
ba29-74cf02568f09", "message": {"id": "aaa2c545-6ec0-47c0-ba29-74cf02568f09", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728850687.890501,
"update_time": null, "content": {"content_type": "text", "parts": ["it might cost.
lets continue to setup our own"]}, "status": "finished_successfully", "end_turn": null,
"weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []},
"request_id": "8d22045cdf44badc-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "ebe1c031-d409-
4c7b-820f-9c60a4842bcf", "children": ["dfb179bd-a2e9-4d10-8467-02e1719aa4ca"]},
"dfb179bd-a2e9-4d10-8467-02e1719aa4ca": {"id": "dfb179bd-a2e9-4d10-8467-02e1719aa4ca",
"message": {"id": "dfb179bd-a2e9-4d10-8467-02e1719aa4ca", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728850730.691109,
"update_time": null, "content": {"content_type": "text", "parts": ["To continue
setting up your own ChatGPT-powered Alexa skill, we&#x27;ll now focus on finalizing
the integration using **Glitch** and linking it to Alexa without relying on AWS
billing. Here&#x27;s how to proceed:\n\n### Step-by-Step Guide to Using Glitch for
Your Alexa Skill\n\n1. **Create a Project on Glitch**:\n   - Go to

```
[Glitch](https://glitch.com/).\n   - Create a new project by clicking on \u201cNew
Project\u201d and choosing a template, such as the &quot;Hello Node&quot; project.\n
- This will give you a basic Node.js server, which you can edit for your Alexa
skill.\n\n2. **Clone or Fork the GitHub Repository**:\n   - Go to the GitHub
repository (such as the one you were initially using), copy the code, and paste it
into the relevant sections of your Glitch project.\n   - Alternatively, you can
&quot;Import from GitHub&quot; directly in Glitch if you&#x27;re more comfortable
using GitHub for version control.\n\n3. **Set Up the OpenAI API**:\n   - Open the
`index.js` or any relevant file in the project. Here, you will need to insert your
**OpenAI API key**.\n   - You can replace any placeholder keys with your real OpenAI
API key, which you should have already set up in earlier steps.\n   - The OpenAI key
usually goes in a part of the code that interacts with the API (e.g., using the
`axios` library or another HTTP request package).\n\n4. **Configure the Alexa Skill
Interaction**:\n   - Ensure that the code handles incoming Alexa requests correctly.
In the file handling the Alexa skill&#x27;s logic (usually `index.js` or similar),
make sure it sends the response from ChatGPT back to the Alexa device.\n   -
You\u2019ll also want to check that the request format Alexa sends matches what your
server expects. Adjust as needed to ensure smooth communication.\n\n5. **Deploy Your
Glitch Project**:\n   - Once the code is ready and you&#x27;ve saved the project, it
will automatically be live with a public URL. This URL will act as your skill&#x27;s
endpoint.\n   - You can find this URL by clicking on &quot;Share&quot; and copying the
live app URL. It will look something like `https://&lt;project-
name&gt;.glitch.me`.\n\n6. **Set Up Your Alexa Developer Console**:\n   - Go to the
[Alexa Developer Console](https://developer.amazon.com/alexa/console/ask).\n   -
Create a new skill or edit your existing one.\n   - Under **Endpoint**, provide the
public URL from Glitch.\n   - Make sure you select **HTTPS** for the endpoint and
check that it\u2019s properly secured (Glitch URLs are HTTPS by default).\n\n7. **Test
the Skill**:\n   - Use the Alexa Developer Console\u2019s built-in tools or your own
Alexa device to test the skill.\n   - Make sure that when you ask Alexa a question, it
hits your Glitch endpoint, sends a request to OpenAI\u2019s API, and reads the
response back to you.\n\n### Final Considerations\n- **Debugging**: If things
aren\u2019t working as expected, use the **Alexa Developer Console logs** and
**Glitch\u2019s logs** (available under the \u201cLogs\u201d tab on the Glitch project)
to troubleshoot.\n- **Enhancing**: Once the core functionality works, you can enhance
it by adding more complex responses, email integration, or even monetizing the skill
via Alexa\u2019s developer tools.\n\nThis approach allows you to avoid AWS entirely,
while still developing a custom Alexa skill that integrates with ChatGPT. Let me know
if you hit any snags!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2c545-6ec0-47c0-ba29-74cf02568f09", "request_id": "8d22045cdf44badc-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa2c545-6ec0-47c0-ba29-74cf02568f09", "children": ["aaa2dffe-7e00-4713-8a25-
8d3836f672ff"]}, "aaa2dffe-7e00-4713-8a25-8d3836f672ff": {"id": "aaa2dffe-7e00-4713-
8a25-8d3836f672ff", "message": {"id": "aaa2dffe-7e00-4713-8a25-8d3836f672ff", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728850930.011711,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
```

~~~
yiLaUJR1QvTRWVrFvdFcR6yR", "size_bytes": 109143, "width": 1343, "height": 581, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "I&#x27;m here. where to click next?
"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"attachments": [{"id": "file-yiLaUJR1QvTRWVrFvdFcR6yR", "size": 109143, "name":
"image.png", "mime_type": "image/png", "width": 1343, "height": 581}],
"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d220a40fc105255-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "dfb179bd-a2e9-
4d10-8467-02e1719aa4ca", "children": ["0cea8d50-d6d0-4ef3-aa06-7a43c54b7a17"]},
"0cea8d50-d6d0-4ef3-aa06-7a43c54b7a17": {"id": "0cea8d50-d6d0-4ef3-aa06-7a43c54b7a17",
"message": {"id": "0cea8d50-d6d0-4ef3-aa06-7a43c54b7a17", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728850973.841694,
"update_time": null, "content": {"content_type": "text", "parts": ["You&#x27;re in the
right place to get started! Now, follow these steps to integrate your Alexa skill with
OpenAI&#x27;s ChatGPT using the Glitch platform:\n\n1. **Set up your project
files**:\n   - Click on the `server.js` file on the left to open it.\n   - In this
file, you&#x27;ll be writing the code that connects your Alexa skill with the OpenAI
API.\n   - You can copy the template from GitHub that connects to OpenAI\u2019s
ChatGPT, like the one you found earlier.\n\n2. **Modify your `package.json`**:\n   -
Open the `package.json` file.\n   - Add the dependencies required for OpenAI API and
Alexa, like `ask-sdk-core`, `ask-sdk-model`, and `axios`. Here\u2019s an example:\n
```json\n     {\n       &quot;name&quot;: &quot;alexa-chatgpt&quot;,\n
&quot;version&quot;: &quot;1.0.0&quot;,\n       &quot;description&quot;: &quot;Alexa
skill with ChatGPT integration&quot;,\n       &quot;dependencies&quot;: {\n
&quot;ask-sdk-core&quot;: &quot;^2.10.0&quot;,\n         &quot;ask-sdk-model&quot;:
&quot;^1.25.0&quot;,\n         &quot;axios&quot;: &quot;^0.21.1&quot;\n       },\n
&quot;engines&quot;: {\n         &quot;node&quot;:
&quot;14.x&quot;\n       }\n     }\n     ```\n\n3. **Write server-side logic**:\n   -
In `server.js`, use the OpenAI and Alexa SDKs to create your logic.\n   - An example
of server code to handle Alexa intents and route them to OpenAI API might look like
this:\n     ```javascript\n     const Alexa = require(&#x27;ask-sdk-core&#x27;);\n
const axios = require(&#x27;axios&#x27;);\n\n     const getOpenAIResponse = async
(userQuery) =&gt; {\n       const response = await
axios.post(&#x27;https://api.openai.com/v1/engines/davinci-codex/completions&#x27;,
{\n         prompt: userQuery,\n         max_tokens: 150\n       }, {\n
headers: {\n           &#x27;Authorization&#x27;: `Bearer
YOUR_OPENAI_API_KEY`\n         }\n       });\n       return
response.data.choices[0].text;\n     };\n\n     const LaunchRequestHandler = {\n
canHandle(handlerInput) {\n         return
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&#x27;LaunchRequest&#x27;;\n       },\n       async handle(handlerInput) {\n
const speakOutput = &#x27;Welcome to your ChatGPT Alexa skill!&#x27;;\n         return
handlerInput.responseBuilder\n           .speak(speakOutput)\n           .getResponse()
;\n       }\n     };\n\n     const OpenAIIntentHandler = {\n
canHandle(handlerInput) {\n         return
Alexa.getRequestType(handlerInput.requestEnvelope) === &#x27;IntentRequest&#x27;\n
&amp;&amp; Alexa.getIntentName(handlerInput.requestEnvelope) ===
&#x27;GetAIResponseIntent&#x27;;\n       },\n       async handle(handlerInput) {\n
~~~

~~~
const query = handlerInput.requestEnvelope.request.intent.slots.query.value;\n
const response = await getOpenAIResponse(query);\n        return
handlerInput.responseBuilder\n          .speak(response)\n          .getResponse();\
n      }\n    };\n\n    const skillBuilder = Alexa.SkillBuilders.custom();\n\n
exports.handler = skillBuilder\n      .addRequestHandlers(\n
LaunchRequestHandler,\n        OpenAIIntentHandler\n      )\n      .lambda();\n
```\n\n4. **Set your OpenAI API key**:\n   - In the `.env` file, store your OpenAI API
key:\n     ```\n     OPENAI_API_KEY=your-openai-api-key-here\n     ```\n\n5. **Run
your project**:\n   - Once the code is written, you can test it directly on Glitch.\n
- Click on `Tools` at the bottom and select `Logs` to see how the app behaves and if
any errors pop up.\n\n6. **Connect with Alexa**:\n   - You can now use Glitch&#x27;s
public URL (found in the `Share` button) as the backend for your Alexa skill by adding
this URL in the endpoint section when configuring your Alexa skill in the Amazon
Developer Console.\n\nLet me know how it goes or if you have any questions!"]},
"status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2dffe-7e00-4713-8a25-8d3836f672ff", "request_id": "8d220a40fc105255-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2dffe-7e00-4713-8a25-8d3836f672ff", "children": ["aaa25ac9-6762-408f-83d3-
9e44bb5885d9"]}, "aaa25ac9-6762-408f-83d3-9e44bb5885d9": {"id": "aaa25ac9-6762-408f-
83d3-9e44bb5885d9", "message": {"id": "aaa25ac9-6762-408f-83d3-9e44bb5885d9", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728851027.323069,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
P1rSNve6zujCKLywzBIcxhOT", "size_bytes": 37281, "width": 604, "height": 412, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "which to click? "]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-P1rSNve6zujCKLywzBIcxhOT", "size": 37281, "name": "image.png",
"mime_type": "image/png", "width": 604, "height": 412}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d220c9f1d965255-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "0cea8d50-d6d0-4ef3-aa06-7a43c54b7a17", "children": ["2ac4bb06-b00d-
4a42-bfe0-b862468419c4"]}, "2ac4bb06-b00d-4a42-bfe0-b862468419c4": {"id": "2ac4bb06-
b00d-4a42-bfe0-b862468419c4", "message": {"id": "2ac4bb06-b00d-4a42-bfe0-b862468419c4",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728851041.633692, "update_time": null, "content": {"content_type": "text", "parts":
["Click **&quot;Download ZIP&quot;** if you want to download the entire repository as
a zip file to your computer. Once downloaded, you can extract it and work with the
project files directly.\n\nAlternatively, if you have GitHub installed or want to use
Git commands, you can copy the HTTPS link and run:\n\n```bash\ngit clone
https://github.com/Joao208/alexa-chatgpt.git\n```\n\nin your terminal to clone the
repository to your machine or your Glitch project directly. Since you are working on
Glitch, you&#x27;ll likely want to download the ZIP and upload the files there, or use
the HTTPS link to import it into Glitch. \n\nLet me know how you want to proceed and
I&#x27;ll guide you!"]}, "status": "finished_successfully","end_turn": true
~~~

```
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa25ac9-6762-408f-83d3-9e44bb5885d9", "request_id": "8d220c9f1d965255-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa25ac9-6762-408f-83d3-9e44bb5885d9", "children": ["aaa29809-1b20-4cae-90c5-
cc0ff8b7950c"]}, "aaa29809-1b20-4cae-90c5-cc0ff8b7950c": {"id": "aaa29809-1b20-4cae-
90c5-cc0ff8b7950c", "message": {"id": "aaa29809-1b20-4cae-90c5-cc0ff8b7950c", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728851083.352508,
"update_time": null, "content": {"content_type": "text", "parts": ["okay direct import
into glitch seems clean. what to click? "]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d220e040f045255-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "2ac4bb06-b00d-4a42-bfe0-b862468419c4", "children": ["057586bd-7848-
49c6-8e11-50028073eb8a"]}, "057586bd-7848-49c6-8e11-50028073eb8a": {"id": "057586bd-
7848-49c6-8e11-50028073eb8a", "message": {"id": "057586bd-7848-49c6-8e11-50028073eb8a",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728851093.803313, "update_time": null, "content": {"content_type": "text", "parts":
["To directly import the project into Glitch using the repository link, follow these
steps:\n\n1. Go to your Glitch project dashboard (where you see the file editor).\n2.
In the left sidebar, click on the &quot;Tools&quot; button at the bottom.\n3. Choose
&quot;Import from GitHub.&quot;\n4. Enter the repository URL
`https://github.com/Joao208/alexa-chatgpt.git`.\n5. Click **OK** or
**Import**.\n\nThis will import the project files from GitHub directly into your
Glitch project. Once imported, you can start modifying or deploying the Alexa skill
based on your needs. Let me know how it goes!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa29809-1b20-4cae-90c5-cc0ff8b7950c", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d220e040f045255-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa29809-
1b20-4cae-90c5-cc0ff8b7950c", "children": ["aaa230f5-70ec-4a75-a1dd-f9bbc4a2fbc4"]},
"aaa230f5-70ec-4a75-a1dd-f9bbc4a2fbc4": {"id": "aaa230f5-70ec-4a75-a1dd-f9bbc4a2fbc4",
"message": {"id": "aaa230f5-70ec-4a75-a1dd-f9bbc4a2fbc4", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728851234.969793, "update_time": null,
"content": {"content_type": "multimodal_text", "parts": [{"content_type":
"image_asset_pointer", "asset_pointer": "file-service://file-RJxXqvbd4NuNHLbmrxiUrcKs",
"size_bytes": 82290, "width": 1340, "height": 542, "fovea": null, "metadata": {"dalle":
null, "gizmo": null, "emu_omit_glimpse_image": null, "emu_patches_override": null,
"sanitized": true}}, "I think i did it and now I have this: "]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-RJxXqvbd4NuNHLbmrxiUrcKs", "size": 82290, "name": "image.png",
"mime_type": "image/png", "width": 1340, "height": 542}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d2211b2da305255-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "057586bd-7848-49c6-8e11-50028073eb8a", "children": ["7bd7d02e-37b3-
4188-8eb3-3b1dbc1da1ab"]}, "7bd7d02e-37b3-4188-8eb3-3b1dbc1da1ab": {"id": "7bd7d02e-
```

~~~
37b3-4188-8eb3-3b1dbc1da1ab", "message": {"id": "7bd7d02e-37b3-4188-8eb3-3b1dbc1da1ab",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728851254.406013, "update_time": null, "content": {"content_type": "text", "parts":
["It looks like you successfully cloned the project into Glitch! Next steps:\n\n1.
**Check if Glitch Installed Dependencies**: After importing the project, Glitch
usually installs the required dependencies like `ask-cli` or any other dependencies
defined in `package.json` automatically. You can confirm by checking the logs or
ensuring that there\u2019s no error related to missing dependencies in the terminal
logs.\n\n2. **Edit the `.env` File**: \n    - You\u2019ll need to add your **OpenAI
API key** into the `.env` file in the project (usually Glitch keeps this file hidden
for security, but you should find it in the sidebar). You can add it like this:\n
```\n      OPENAI_API_KEY=your-api-key\n      ```\n    - Make sure you replace `your-
api-key` with your actual API key from OpenAI.\n\n3. **Run the Project**: After
configuring the `.env` file, ensure the app is running smoothly by checking the
**Logs** section. If there are no issues, you can try sending some test requests to
see if everything works.\n\n4. **Test It Live**: \n   - Once the project is set up,
you can preview the project using the **Show** option in Glitch. \n   - From here, you
should be able to test your custom Alexa ChatGPT skill.\n\n5. **Deploy on Alexa**: You
can continue to follow the instructions on configuring your Alexa skill using ASK-CLI,
but in Glitch, the main focus is on ensuring the backend logic is functioning smoothly
with OpenAI before linking it to Alexa.\n\nIf you need any specific help with any step,
let me know!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa230f5-70ec-4a75-a1dd-f9bbc4a2fbc4", "request_id": "8d2211b2da305255-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa230f5-70ec-4a75-a1dd-f9bbc4a2fbc4", "children": ["aaa299ba-51d7-4ada-8a24-
cd7b4a282f58"]}, "aaa299ba-51d7-4ada-8a24-cd7b4a282f58": {"id": "aaa299ba-51d7-4ada-
8a24-cd7b4a282f58", "message": {"id": "aaa299ba-51d7-4ada-8a24-cd7b4a282f58", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728851562.733874,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
P72XIFXzN31shXp8MlWqm8fs", "size_bytes": 101018, "width": 1056, "height": 398, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, {"content_type":
"image_asset_pointer", "asset_pointer": "file-service://file-GHVsOmlVvIeMbJMN8Y2yAmWb",
"size_bytes": 77141, "width": 706, "height": 683, "fovea": null, "metadata": {"dalle":
null, "gizmo": null, "emu_omit_glimpse_image": null, "emu_patches_override": null,
"sanitized": true}}, "sooo what does it mean? what should i do? "]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-P72XIFXzN31shXp8MlWqm8fs", "size": 101018, "name": "image.png",
"mime_type": "image/png", "width": 1056, "height": 398}, {"id": "file-
GHVsOmlVvIeMbJMN8Y2yAmWb", "size": 77141, "name":
"11d862b8d07f235ecf56804bee15f110.png", "mime_type": "image/png", "width": 706,
"height": 683}], "serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2219b38fa7ba8c-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "7bd7d02e-37b3-
4188-8eb3-3b1dbc1da1ab", "children": ["ae86fc6f-31f1-437f-aa76-cb3eeade4cba"]},
~~~

```
"ae86fc6f-31f1-437f-aa76-cb3eeade4cba": {"id": "ae86fc6f-31f1-437f-aa76-cb3eeade4cba",
"message": {"id": "ae86fc6f-31f1-437f-aa76-cb3eeade4cba", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728851584.603576,
"update_time": null, "content": {"content_type": "text", "parts": ["It looks like
you&#x27;re in the `.env` file of your Glitch project, which is used for storing
environment variables. In this file, you need to add sensitive or configuration-
specific values like your OpenAI API key for connecting to ChatGPT.\n\nHere\u2019s
what you need to do:\n\n1. **Add your OpenAI API Key**:  \n   In the `.env` file,
create a new environment variable for your API key. You can do this by adding a new
variable like this:\n   \n   ```plaintext\n
OPENAI_API_KEY=your_openai_api_key_here\n   ```\n\n   Replace
`your_openai_api_key_here` with the actual API key you received from OpenAI. \n\n2.
**Save the file**:  \n   After adding your key, just save the file (Glitch
automatically saves changes most of the time).\n\n3. **Modify your code to use the API
key**:  \n   In your `index.js` (or wherever the code makes the API call), ensure that
you&#x27;re using the key from the environment. For example, your code would look like
this:\n\n   ```javascript\n   const apiKey = process.env.OPENAI_API_KEY;\n   ```\n\n
This ensures that the key is read securely from the environment variable instead of
being hardcoded.\n\n4. **Run the project**:  \n   Once you\u2019ve saved the changes,
you can test your project by clicking the &quot;Show&quot; button in Glitch to see the
live version of your application.\n\nLet me know if you need help with any part of the
code, like where exactly to place the API key in the existing project!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa299ba-51d7-4ada-8a24-cd7b4a282f58", "request_id": "8d2219b38fa7ba8c-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa299ba-51d7-4ada-8a24-cd7b4a282f58", "children": ["aaa26b2f-f06a-432f-bba3-
0934f4b5ac84"]}, "aaa26b2f-f06a-432f-bba3-0934f4b5ac84": {"id": "aaa26b2f-f06a-432f-
bba3-0934f4b5ac84", "message": {"id": "aaa26b2f-f06a-432f-bba3-0934f4b5ac84", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728851903.367292,
"update_time": null, "content": {"content_type": "text", "parts": ["const Alexa =
require(&quot;ask-sdk-core&quot;);\r\nconst axios =
require(&quot;axios&quot;);\r\n\r\nasync function askToGPT(prompt) {\r\n  const
response = await axios.post(\r\n    &quot;https://api.openai.com/v1/engines/text-
davinci-003/completions&quot;,\r\n    {\r\n      prompt,\r\n      max_tokens: 100,\r\n
n: 1,\r\n      temperature: 0.5,\r\n      stop: &quot;.&quot;,\r\n      top_p:
1,\r\n    },\r\n    {\r\n      headers: {\r\n        &quot;Content-Type&quot;:
&quot;application/json&quot;,\r\n        Authorization: `Bearer
${process.env.OPENAI_API_KEY}}`,\r\n      },\r\n    }\r\n  );\r\n\r\n  return
response.data.choices[0].text;\r\n}\r\n\r\nfunction formatString(text) {\r\n  return
text.replace(/\\n+/g, &quot; &quot;);\r\n}\r\n\r\nconst LaunchRequestHandler = {\r\n
canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;LaunchRequest&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const response =\r\n      &#x27;Seja bem-vindo ao GPT-3. Voc\u00ea pode perguntar algo
como &quot;chat, qual a capital da Fran\u00e7a?&quot;&#x27;;\r\n\r\n    return
handlerInput.responseBuilder.speak(response).getResponse();\r\n  },\r\n};\r\n\r\nconst
```

```
HelloWorldIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AskToGPTIntent&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const question = Alexa.getSlotValue(\r\n      handlerInput.requestEnvelope,\r\n
&quot;question&quot;\r\n    );\r\n\r\n    const responseFromGPT = await
askToGPT(question);\r\n\r\n    const formattedResponse =
formatString(responseFromGPT);\r\n\r\n    return
handlerInput.responseBuilder.speak(formattedResponse).getResponse();\r\n  },\r\n};\r\n
\r\nconst HelpIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AMAZON.HelpIntent&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput =\r\n      &quot;Voc\u00ea pode perguntar algo como &#x27;chat,
ideia de nome para beb\u00eas&#x27;&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst CancelAndStopIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      (Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.CancelIntent&quot; ||\r\n
Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.StopIntent&quot;)\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput = &quot;At\u00e9 mais!&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst FallbackIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.FallbackIntent&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput =\r\n      &quot;Desculpe, n\u00e3o tenho conhecimento sobre isso.
Por favor, tente novamente.&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst SessionEndedRequestHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===\r\n
&quot;SessionEndedRequest&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
return handlerInput.responseBuilder.getResponse();\r\n  },\r\n};\r\n\r\nconst
IntentReflectorHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;IntentRequest&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n    const
speakOutput = `Que tal me perguntar algo como &#x27;quanto est\u00e1 o d\u00f3lar
hoje?&#x27;`;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst ErrorHandler = {\r\n  canHandle() {\r\n    return true;\r\n  },\r\n
handle(handlerInput) {\r\n    const speakOutput =\r\n      &quot;Desculpe, parece que
tivemos um problema aqui! Acesse github.com/joao208/alexa-chatgpt para mais
informa\u00e7\u00f5es.&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nex
ports.handler = Alexa.SkillBuilders.custom()\r\n  .addRequestHandlers(\r\n
LaunchRequestHandler,\r\n    HelloWorldIntentHandler,\r\n    HelpIntentHandler,\r\n
CancelAndStopIntentHandler,\r\n    FallbackIntentHandler,\r\n
SessionEndedRequestHandler,\r\n
IntentReflectorHandler\r\n  )\r\n  .addErrorHandlers(ErrorHandler)\r\n  .lambda();\r\n
\n\neverything looking good here?"]}, "status": "finished_successfully", "end_turn":
null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets":
[]}, "request_id": "8d222208a85b0e01-MXP", "message_source": null, "timestamp_":
"absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent":
"ae86fc6f-31f1-437f-aa76-cb3eeade4cba", "children": ["ec2261dc-602f-49f3-8191-
1b1a4546ba7e"]}, "ec2261dc-602f-49f3-8191-1b1a4546ba7e": {"id": "ec2261dc-602f-49f3-
8191-1b1a4546ba7e", "message": {"id": "ec2261dc-602f-49f3-8191-1b1a4546ba7e", "author":
{"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728851941.78415,
"update_time": null, "content": {"content_type": "text", "parts": ["Your code looks
mostly good, but there are a couple of things to check and potentially correct:\n\n###
1. **Authorization Header Syntax**\nThere is a small issue with the `Authorization`
header. It seems like there&#x27;s an extra `}` at the end. Here\u2019s the correct
syntax:\n\n```javascript\nAuthorization: `Bearer
${process.env.OPENAI_API_KEY}`,\n```\n\nNotice the removal of the extra curly brace
`}`.\n\n### 2. **Check your `askToGPT` function for API endpoint**:\nMake sure
you\u2019re using the correct endpoint for the OpenAI GPT-3 model. If you&#x27;re
using GPT-4, for example, the engine or model name might be different. Ensure your
OpenAI account has the correct API access for the engine you&#x27;re using.\n\n### 3.
**Ensure OpenAI API key is properly stored**\nMake sure your `.env` file in Glitch is
correctly set up with your OpenAI API
key:\n```\nOPENAI_API_KEY=your_openai_api_key_here\n```\n\n### 4. **Improve Error
Handling**\nConsider adding better error handling to your `askToGPT` function in case
the OpenAI API request fails. For example:\n\n```javascript\nasync function
askToGPT(prompt) {\n  try {\n    const response = await axios.post(\n
&quot;https://api.openai.com/v1/engines/text-davinci-003/completions&quot;,\n      {\n
prompt,\n        max_tokens: 100,\n        n: 1,\n        temperature: 0.5,\n
stop: &quot;.&quot;,\n        top_p: 1,\n      },\n      {\n        headers: {\n
&quot;Content-Type&quot;: &quot;application/json&quot;,\n          Authorization:
`Bearer ${process.env.OPENAI_API_KEY}`,\n        },\n      }\n    );\n\n    return
response.data.choices[0].text;\n  } catch (error) {\n    console.error(&quot;Error
calling GPT:&quot;, error);\n    return &quot;Sorry, something went wrong while
contacting GPT.&quot;;\n  }\n}\n```\n\n### 5. **General Checks**\n- Make sure that
your Alexa skill is correctly configured with the necessary intents in the Alexa
Developer Console. For example, the `AskToGPTIntent` must be defined with a slot for
`question`.\n  \n- Ensure the Alexa Skill is able to correctly communicate with your
hosted Glitch project by configuring the endpoint in your Alexa Developer Console to
point to your Glitch app\u2019s URL.\n\nOnce you\u2019ve done these things, you should
be in a good position to test the skill! Let me know if any part needs further
clarification!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa26b2f-f06a-432f-bba3-0934f4b5ac84", "request_id": "8d222208a85b0e01-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa26b2f-f06a-432f-bba3-0934f4b5ac84", "children": ["aaa22d12-be82-4380-8626-
18be11b4585f"]}, "aaa22d12-be82-4380-8626-18be11b4585f": {"id": "aaa22d12-be82-4380-
8626-18be11b4585f", "message": {"id": "aaa22d12-be82-4380-8626-18be11b4585f", "author":
```

```
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728851978.669227,
"update_time": null, "content": {"content_type": "text", "parts": ["i want to use gpt
4o"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2223dc5fc383a6-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "ec2261dc-602f-
49f3-8191-1b1a4546ba7e", "children": ["7dd8663c-b349-456a-ae45-340a7f96b44e"]},
"7dd8663c-b349-456a-ae45-340a7f96b44e": {"id": "7dd8663c-b349-456a-ae45-340a7f96b44e",
"message": {"id": "7dd8663c-b349-456a-ae45-340a7f96b44e", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852001.534577,
"update_time": null, "content": {"content_type": "text", "parts": ["To switch to GPT-4,
you will need to update the endpoint in your `askToGPT` function. Here&#x27;s what you
should change:\n\n1. **API Endpoint**: The endpoint for GPT-4 is different from the
one for GPT-3.\n2. **Model Name**: Replace the model/engine name with GPT-
4.\n\nHere&#x27;s how you can modify the `askToGPT` function to use GPT-4:\n\n###
Updated Code for GPT-4:\n```javascript\nasync function askToGPT(prompt) {\n  const
response = await axios.post(\n
&quot;https://api.openai.com/v1/chat/completions&quot;,\n    {\n      model:
&quot;gpt-4&quot;,\n      messages: [{ role: &quot;user&quot;, content: prompt }],\n
max_tokens: 100,\n      temperature: 0.5,\n    },\n    {\n      headers: {\n
&quot;Content-Type&quot;: &quot;application/json&quot;,\n        Authorization:
`Bearer ${process.env.OPENAI_API_KEY}`,\n      },\n    }\n  );\n\n  return
response.data.choices[0].message.content;\n}\n```\n\n### Key Changes:\n1. **Endpoint**:
Switched to `https://api.openai.com/v1/chat/completions` (used for GPT-4 and the chat-
based models).\n2. **Model Name**: Updated to `&quot;gpt-4&quot;`.\n3. **Messages
Format**: The GPT-4 API expects a list of messages in the `messages` parameter, with a
`role` for each message (e.g., user, assistant).\n4. **Response**: The response
structure for GPT-4 is slightly different, so we return
`response.data.choices[0].message.content`.\n\n### .env File:\nEnsure your `.env` file
contains the correct OpenAI API
key:\n\n```\nOPENAI_API_KEY=your_openai_api_key_here\n```\n\nOnce these changes are
made, your Alexa skill should now use GPT-4 for processing user queries! Let me know
if you need any further help with this!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa22d12-be82-4380-8626-18be11b4585f", "request_id": "8d2223dc5fc383a6-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa22d12-be82-4380-8626-18be11b4585f", "children": ["aaa2def4-ca52-4d94-bdd0-
e03b200d3d54"]}, "aaa2def4-ca52-4d94-bdd0-e03b200d3d54": {"id": "aaa2def4-ca52-4d94-
bdd0-e03b200d3d54", "message": {"id": "aaa2def4-ca52-4d94-bdd0-e03b200d3d54", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728852195.80992,
"update_time": null, "content": {"content_type": "text", "parts": ["async function
askToGPT(prompt) {\r\n  const response = await axios.post(\r\n
&quot;https://api.openai.com/v1/chat/completions&quot;,\r\n    {\r\n      model:
&quot;gpt-4&quot;,\r\n      messages: [{ role: &quot;user&quot;, content:
prompt }],\r\n      max_tokens: 100,\r\n      temperature: 0.5,\r\n    },\r\n    {\r\n
headers: {\r\n        &quot;Content-Type&quot;: &quot;application/json&quot;,\r\n
```

```
Authorization: `Bearer
${process.env.OPENAI_API_KEY}`,\r\n      },\r\n    }\r\n  );\r\n\r\n  return
response.data.choices[0].message.content;\r\n}\r\n\n\nso on what line should i past
this assuming the text starts at 1:\n\nconst Alexa = require(&quot;ask-sdk-
core&quot;);\r\nconst axios = require(&quot;axios&quot;);\r\n\r\nasync function
askToGPT(prompt) {\r\n  const response = await axios.post(\r\n
&quot;https://api.openai.com/v1/engines/text-davinci-003/completions&quot;,\r\n
{\r\n      prompt,\r\n      max_tokens: 100,\r\n      n: 1,\r\n      temperature:
0.5,\r\n      stop: &quot;.&quot;,\r\n      top_p: 1,\r\n    },\r\n    {\r\n
headers: {\r\n        &quot;Content-Type&quot;: &quot;application/json&quot;,\r\n
Authorization: `Bearer
${process.env.OPENAI_API_KEY}}`,\r\n      },\r\n    }\r\n  );\r\n\r\n  return
response.data.choices[0].text;\r\n}\r\n\r\nfunction formatString(text) {\r\n  return
text.replace(/\\n+/g, &quot; &quot;);\r\n}\r\n\r\nconst LaunchRequestHandler = {\r\n
canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;LaunchRequest&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const response =\r\n      &#x27;Seja bem-vindo ao GPT-3. Voc\u00ea pode perguntar algo
como &quot;chat, qual a capital da Fran\u00e7a?&quot;&#x27;;\r\n\r\n    return
handlerInput.responseBuilder.speak(response).getResponse();\r\n  },\r\n};\r\n\r\nconst
HelloWorldIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AskToGPTIntent&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const question = Alexa.getSlotValue(\r\n      handlerInput.requestEnvelope,\r\n
&quot;question&quot;\r\n    );\r\n\r\n    const responseFromGPT = await
askToGPT(question);\r\n\r\n    const formattedResponse =
formatString(responseFromGPT);\r\n\r\n    return
handlerInput.responseBuilder.speak(formattedResponse).getResponse();\r\n  },\r\n};\r\n
\r\nconst HelpIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AMAZON.HelpIntent&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput =\r\n      &quot;Voc\u00ea pode perguntar algo como &#x27;chat,
ideia de nome para beb\u00eas&#x27;&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst CancelAndStopIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      (Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.CancelIntent&quot; ||\r\n
Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.StopIntent&quot;)\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput = &quot;At\u00e9 mais!&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst FallbackIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.FallbackIntent&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput =\r\n      &quot;Desculpe, n\u00e3o tenho conhecimento sobre isso.
```

Por favor, tente novamente.&quot;;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nco
nst SessionEndedRequestHandler = {\r\n canHandle(handlerInput) {\r\n   return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===\r\n
&quot;SessionEndedRequest&quot;\r\n   );\r\n },\r\n handle(handlerInput) {\r\n
return handlerInput.responseBuilder.getResponse();\r\n },\r\n};\r\n\r\nconst
IntentReflectorHandler = {\r\n canHandle(handlerInput) {\r\n   return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;IntentRequest&quot;\r\n   );\r\n },\r\n handle(handlerInput) {\r\n   const
speakOutput = `Que tal me perguntar algo como &#x27;quanto est\u00e1 o d\u00f3lar
hoje?&#x27;`;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nco
nst ErrorHandler = {\r\n canHandle() {\r\n   return true;\r\n },\r\n
handle(handlerInput) {\r\n   const speakOutput =\r\n   &quot;Desculpe, parece que
tivemos um problema aqui! Acesse github.com/joao208/alexa-chatgpt para mais
informa\u00e7\u00f5es.&quot;;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nex
ports.handler = Alexa.SkillBuilders.custom()\r\n .addRequestHandlers(\r\n
LaunchRequestHandler,\r\n   HelloWorldIntentHandler,\r\n   HelpIntentHandler,\r\n
CancelAndStopIntentHandler,\r\n   FallbackIntentHandler,\r\n
SessionEndedRequestHandler,\r\n
IntentReflectorHandler\r\n)\r\n .addErrorHandlers(ErrorHandler)\r\n .lambda();\r\n
"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d22292c6b82bae5-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "7dd8663c-b349-
456a-ae45-340a7f96b44e", "children": ["2bfe857c-39e4-4e86-bcc7-647cd19c7945"]},
"2bfe857c-39e4-4e86-bcc7-647cd19c7945": {"id": "2bfe857c-39e4-4e86-bcc7-647cd19c7945",
"message": {"id": "2bfe857c-39e4-4e86-bcc7-647cd19c7945", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852215.261523,
"update_time": null, "content": {"content_type": "text", "parts": ["You should replace
the existing `askToGPT` function (starting at line 3) with the updated GPT-4
version.\n\nHere are the steps:\n\n1. **Find the `askToGPT` function**: This starts at
line 3 in your code:\n   ```javascript\n   async function askToGPT(prompt) {\n
const response = await axios.post(\n
&quot;https://api.openai.com/v1/engines/text-davinci-003/completions&quot;,\n
{\n   prompt,\n   max_tokens: 100,\n   n: 1,\n
temperature: 0.5,\n   stop: &quot;.&quot;,\n   top_p: 1,\n   },\n
{\n   headers: {\n   &quot;Content-Type&quot;:
&quot;application/json&quot;,\n   Authorization: `Bearer
${process.env.OPENAI_API_KEY}`,\n   },\n   }\n   );\n   \n
return response.data.choices[0].text;\n   }\n   ```\n\n2. **Replace it with the new
GPT-4 code**:\n   ```javascript\n   async function askToGPT(prompt) {\n   const
response = await axios.post(\n
&quot;https://api.openai.com/v1/chat/completions&quot;,\n   {\n   model:
&quot;gpt-4&quot;,\n   messages: [{ role: &quot;user&quot;, content:
prompt }],\n   max_tokens: 100,\n   temperature: 0.5,\n   },\n
{\n   headers: {\n   &quot;Content-Type&quot;:
&quot;application/json&quot;,\n   Authorization: `Bearer

~~~
${process.env.OPENAI_API_KEY}`,\n          },\n        }\n      );\n    \n      return
response.data.choices[0].message.content;\n    }\n    ```\n\n3. **Save the
file**.\n\nWith this change, your skill will now use GPT-4 instead of GPT-3. If you
encounter any issues, feel free to let me know!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2def4-ca52-4d94-bdd0-e03b200d3d54", "request_id": "8d22292c6b82bae5-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2def4-ca52-4d94-bdd0-e03b200d3d54", "children": ["aaa2b354-8d10-4566-aa72-
2a612bc3eeff"]}, "aaa2b354-8d10-4566-aa72-2a612bc3eeff": {"id": "aaa2b354-8d10-4566-
aa72-2a612bc3eeff", "message": {"id": "aaa2b354-8d10-4566-aa72-2a612bc3eeff", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728852343.77109,
"update_time": null, "content": {"content_type": "text", "parts": ["okay it auto saves
what next? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d222cc9e8220e23-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "2bfe857c-39e4-
4e86-bcc7-647cd19c7945", "children": ["e0648988-f815-4d99-b185-ea04aca4faf9"]},
"e0648988-f815-4d99-b185-ea04aca4faf9": {"id": "e0648988-f815-4d99-b185-ea04aca4faf9",
"message": {"id": "e0648988-f815-4d99-b185-ea04aca4faf9", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852363.922483,
"update_time": null, "content": {"content_type": "text", "parts": ["Now that your code
is updated to use GPT-4 and Glitch has auto-saved it, here are the next steps to fully
set up your Alexa skill:\n\n### 1. **Test the project on Glitch**\n   - Go to the
&quot;Tools&quot; tab at the bottom of the Glitch project page, and select
**Terminal**.\n   - In the terminal, run the following command to install any missing
packages:\n     ```bash\n     npm install\n     ```\n\n### 2. **Set your OpenAI API
key**\n   - Go to the `.env` file in the Glitch project.\n   - Add your OpenAI API key
like this:\n     ```plaintext\n     OPENAI_API_KEY=your_openai_api_key_here\n
```\n\n### 3. **Deploy the project**\n   - You can run the server by clicking on
**Show Live** at the top of the Glitch page.\n   - This should start the server and
allow you to make sure it is running properly.\n\n### 4. **Testing the Alexa skill**\n
- If you are using the Alexa Developer Console, link your project to this Glitch URL.
You can add the endpoint in your Alexa Skill&#x27;s configuration (make sure it uses
an HTTPS URL).\n   - You can now test the functionality through the Alexa Developer
Console or an Alexa-enabled device.\n\nIf everything runs as expected, your Alexa
skill should now be using GPT-4 to respond to user questions!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2b354-8d10-4566-aa72-2a612bc3eeff", "request_id": "8d222cc9e8220e23-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2b354-8d10-4566-aa72-2a612bc3eeff", "children": ["aaa2cf20-6a20-4528-b489-
3fb674d48da3"]}, "aaa2cf20-6a20-4528-b489-3fb674d48da3": {"id": "aaa2cf20-6a20-4528-
b489-3fb674d48da3", "message": {"id": "aaa2cf20-6a20-4528-b489-3fb674d48da3", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728852533.853318,
~~~

"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
LR1COLL1zfLRuTh5KwpuWguP", "size_bytes": 68017, "width": 1358, "height": 591, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "we okay or do we have a problem?
"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"attachments": [{"id": "file-LR1COLL1zfLRuTh5KwpuWguP", "size": 68017, "name":
"image.png", "mime_type": "image/png", "width": 1358, "height": 591}],
"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d223168d802bb03-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "e0648988-f815-
4d99-b185-ea04aca4faf9", "children": ["25602a8d-c4ef-499a-80b2-f31c5b6829af"]},
"25602a8d-c4ef-499a-80b2-f31c5b6829af": {"id": "25602a8d-c4ef-499a-80b2-f31c5b6829af",
"message": {"id": "25602a8d-c4ef-499a-80b2-f31c5b6829af", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852544.501958,
"update_time": null, "content": {"content_type": "text", "parts": ["It looks like the
installation completed successfully, but there are some warnings about deprecated
packages and vulnerabilities. Here\u2019s what the output means:\n\n1. **Deprecated
Warnings**: This happens when certain packages are outdated but still functional. You
can update these packages later, but they shouldn&#x27;t prevent your current project
from working.\n2. **Vulnerabilities**: It found 4 vulnerabilities (2 moderate, 2 high).
You can fix them by running:\n   - `npm audit fix` (this will automatically fix the
vulnerabilities if possible)\n   - `npm audit` (this will give you more detailed
information about the vulnerabilities and manual steps if needed)\n\nFor now, the
warnings and vulnerabilities won\u2019t stop your app from working. You can proceed to
test the app as is, and later, if needed, address these issues by updating the
packages.\n\nTo proceed, you can:\n1. Run the application and test it out on
Glitch.\n2. Set up your OpenAI API key in the `.env` file if you haven&#x27;t already
done so.\n\nIf you&#x27;d like to fix the vulnerabilities before proceeding, you can
run `npm audit fix` to resolve them automatically. Otherwise, everything looks okay to
move forward with testing!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2cf20-6a20-4528-b489-3fb674d48da3", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d223168d802bb03-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa2cf20-
6a20-4528-b489-3fb674d48da3", "children": ["aaa26bd4-26de-46eb-81f7-d702422c81ce"]},
"aaa26bd4-26de-46eb-81f7-d702422c81ce": {"id": "aaa26bd4-26de-46eb-81f7-d702422c81ce",
"message": {"id": "aaa26bd4-26de-46eb-81f7-d702422c81ce", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728852737.135383, "update_time": null,
"content": {"content_type": "multimodal_text", "parts": [{"content_type":
"image_asset_pointer", "asset_pointer": "file-service://file-TquaNDhzLNtk5SOGQMakq39Y",
"size_bytes": 61005, "width": 1328, "height": 494, "fovea": null, "metadata": {"dalle":
null, "gizmo": null, "emu_omit_glimpse_image": null, "emu_patches_override": null,
"sanitized": true}}, "I should click on what? "]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"attachments": [{"id": "file-
TquaNDhzLNtk5SOGQMakq39Y", "size": 61005, "name": "image.png", "mime_type":
"image/png", "width": 1328, "height": 494}], "serialization_metadata":
{"custom_symbol_offsets": []}, "targeted_reply": "### 3\\. **Deploy the project**\n\n*

```
You can run the server by clicking on **Show Live** at the top of the Glitch page.\n*
This should start the server and allow you to make sure it is running properly.",
"request_id": "8d22365faa2abae8-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "25602a8d-c4ef-
499a-80b2-f31c5b6829af", "children": ["c6930824-ad6a-430c-a310-c9af74fc595c"]},
"c6930824-ad6a-430c-a310-c9af74fc595c": {"id": "c6930824-ad6a-430c-a310-c9af74fc595c",
"message": {"id": "c6930824-ad6a-430c-a310-c9af74fc595c", "author": {"role": "system",
"name": null, "metadata": {}}, "create_time": 1728852737.136083, "update_time": null,
"content": {"content_type": "text", "parts": [""]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"request_id": "8d22365faa2abae8-MXP",
"message_source": null, "timestamp_": "absolute", "message_type": null,
"is_visually_hidden_from_conversation": true}, "recipient": "all", "channel": null},
"parent": "aaa26bd4-26de-46eb-81f7-d702422c81ce", "children": ["8d5b764a-2132-471b-
90e2-cf7421257693"]}, "8d5b764a-2132-471b-90e2-cf7421257693": {"id": "8d5b764a-2132-
471b-90e2-cf7421257693", "message": {"id": "8d5b764a-2132-471b-90e2-cf7421257693",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728852746.803216, "update_time": null, "content": {"content_type": "text", "parts":
["Now that you have everything set up, you should click the **&quot;Show Live&quot;**
button at the top of your Glitch project page to run the server. This will start your
app and allow you to verify that everything is functioning properly.\n\nOnce the
server starts running, you should test it with a sample prompt to see if it connects
to GPT-4 successfully and returns responses via the Alexa Skill you\u2019ve
built.\n\nLet me know how it goes or if you need more guidance from this point
onward!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "c6930824-ad6a-430c-a310-c9af74fc595c", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d22365faa2abae8-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "c6930824-
ad6a-430c-a310-c9af74fc595c", "children": ["aaa2573b-f531-4c54-8fc0-0091797ee8c2"]},
"aaa2573b-f531-4c54-8fc0-0091797ee8c2": {"id": "aaa2573b-f531-4c54-8fc0-0091797ee8c2",
"message": {"id": "aaa2573b-f531-4c54-8fc0-0091797ee8c2", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728852796.770491, "update_time": null,
"content": {"content_type": "text", "parts": ["i don&#x27;t see a show live button"]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2237d82838bae8-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "8d5b764a-2132-
471b-90e2-cf7421257693", "children": ["b6eb6abc-f7bd-48de-ba0a-095fe22388d3"]},
"b6eb6abc-f7bd-48de-ba0a-095fe22388d3": {"id": "b6eb6abc-f7bd-48de-ba0a-095fe22388d3",
"message": {"id": "b6eb6abc-f7bd-48de-ba0a-095fe22388d3", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852808.550686,
"update_time": null, "content": {"content_type": "text", "parts": ["In Glitch, if you
don&#x27;t see a **&quot;Show Live&quot;** button, you might be working on a project
that doesn&#x27;t automatically serve as a live website. To activate the live preview,
you can try the following steps:\n\n1. Click on the **&quot;Tools&quot;** button
located at the bottom of the Glitch sidebar.\n2. Select **&quot;Terminal&quot;** from
the dropdown list to open a terminal window.\n3. In the terminal window, run the
command `npm start` to manually start the app.\n\nThis should start your server, and
you will see a live preview URL appear at the top of the terminal or on the project
dashboard.\n\nLet me know if that works for you!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2573b-f531-4c54-8fc0-0091797ee8c2", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d2237d82838bae8-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa2573b-
f531-4c54-8fc0-0091797ee8c2", "children": ["aaa2e81a-b793-4403-be3a-d73144e76b7b"]},
"aaa2e81a-b793-4403-be3a-d73144e76b7b": {"id": "aaa2e81a-b793-4403-be3a-d73144e76b7b",
"message": {"id": "aaa2e81a-b793-4403-be3a-d73144e76b7b", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728852882.612595, "update_time": null,
"content": {"content_type": "text", "parts": ["its missing script start"]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2239f0e922bae8-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "b6eb6abc-f7bd-
48de-ba0a-095fe22388d3", "children": ["07b20a14-d87f-4d9a-836c-695f526be1f1"]},
"07b20a14-d87f-4d9a-836c-695f526be1f1": {"id": "07b20a14-d87f-4d9a-836c-695f526be1f1",
"message": {"id": "07b20a14-d87f-4d9a-836c-695f526be1f1", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852895.480249,
"update_time": null, "content": {"content_type": "text", "parts": ["If you&#x27;re
missing the `&quot;start&quot;` script in your `package.json` file, you can add it
manually so Glitch knows how to start your app. Here&#x27;s how you can do it:\n\n1.
Open your **`package.json`** file.\n2. Look for the `&quot;scripts&quot;` section. If
it\u2019s missing, you can add it, or if it exists, update it to include a
`&quot;start&quot;` script.\n\nHere\u2019s what the `scripts` section should look
like:\n\n```json\n&quot;scripts&quot;: {\n  &quot;start&quot;: &quot;node
index.js&quot;\n}\n```\n\n3. After making that change, Glitch should now be able to
run your project when you press the **&quot;Show&quot;** button (which might appear
after you add the start script) or when you run `npm start` in the terminal.\n\nOnce
you add this, Glitch will know how to start the app, and you should be able to see the
live preview!\n\nLet me know if that works or if you need further assistance!"]},
"status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2e81a-b793-4403-be3a-d73144e76b7b", "request_id": "8d2239f0e922bae8-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa2e81a-b793-4403-be3a-d73144e76b7b", "children": ["aaa281f5-5163-43b1-8f49-
d6989ecda2ef"]}, "aaa281f5-5163-43b1-8f49-d6989ecda2ef": {"id": "aaa281f5-5163-43b1-
8f49-d6989ecda2ef", "message": {"id": "aaa281f5-5163-43b1-8f49-d6989ecda2ef", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728852974.561838,
"update_time": null, "content": {"content_type": "text", "parts": ["its this:
&quot;scripts&quot;: {\r\n    &quot;test&quot;: &quot;echo \\&quot;Error: no test
specified\\&quot; &amp;&amp; exit 1&quot;\r\n  },"]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d223c2f0de8bae8-MXP", "message_source": null, "timestamp_": "absolute",
```

```
"message_type": null}, "recipient": "all", "channel": null}, "parent": "07b20a14-d87f-
4d9a-836c-695f526be1f1", "children": ["f3ce4de4-623c-46ff-b0c9-d7c57c8d0526"]},
"f3ce4de4-623c-46ff-b0c9-d7c57c8d0526": {"id": "f3ce4de4-623c-46ff-b0c9-d7c57c8d0526",
"message": {"id": "f3ce4de4-623c-46ff-b0c9-d7c57c8d0526", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728852984.340374,
"update_time": null, "content": {"content_type": "text", "parts": ["To fix the missing
start script, modify your `package.json` file by replacing the existing
`&quot;scripts&quot;` section with the following:\n\n```json\n&quot;scripts&quot;: {\n
&quot;start&quot;: &quot;node index.js&quot;,\n  &quot;test&quot;: &quot;echo
\\&quot;Error: no test specified\\&quot; &amp;&amp; exit 1&quot;\n}\n```\n\nThis adds
the necessary `&quot;start&quot;` script, which tells Glitch to start your app by
running `node index.js`.\n\nOnce you add this, save the file, and then the
**&quot;Show&quot;** button should appear in the Glitch interface for previewing your
project. Alternatively, you can run the project manually in the terminal by
typing:\n\n```\nnpm start\n```\n\nLet me know if this works or if you encounter any
further issues!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa281f5-5163-43b1-8f49-d6989ecda2ef", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d223c2f0de8bae8-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa281f5-
5163-43b1-8f49-d6989ecda2ef", "children": ["aaa2d7ae-bc70-4bfc-b83b-50b3e5dd6d0c"]},
"aaa2d7ae-bc70-4bfc-b83b-50b3e5dd6d0c": {"id": "aaa2d7ae-bc70-4bfc-b83b-50b3e5dd6d0c",
"message": {"id": "aaa2d7ae-bc70-4bfc-b83b-50b3e5dd6d0c", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728853215.669108, "update_time": null,
"content": {"content_type": "text", "parts": ["    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\n\n\n\n\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
```

```
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
```

```
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
```

```
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\r\n/rbd/pnpm-
volume/71b15dc5-1fc5-4fe7-95d9-
a91d013c926a/node_modules/.registry.npmjs.org/axios/1.3.2/node_modules/axios/index.js:
1\r\nimport axios from &#x27;./lib/axios.js&#x27;;\r\n       ^^^^^\r\nSyntaxError:
Unexpected identifier\r\n    at Module._compile
(internal/modules/cjs/loader.js:723:23)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n    at Module.load
(internal/modules/cjs/loader.js:653:32)\r\n    at tryModuleLoad
(internal/modules/cjs/loader.js:593:12)\r\n    at Function.Module._load
(internal/modules/cjs/loader.js:585:3)\r\n    at Module.require
(internal/modules/cjs/loader.js:692:17)\r\n    at require
(internal/modules/cjs/helpers.js:25:18)\r\nat Object.&lt;anonymous&gt;
(/app/index.js:2:15)\r\n    at Module._compile
(internal/modules/cjs/loader.js:778:30)\r\n    at Object.Module._extensions..js
(internal/modules/cjs/loader.js:789:10)\r\n\ud83d\udc1e The debugger could not be
started. See https://glitch.happyfox.com/kb/search/?q=debugger\n\n\nwhat happening? is
it working? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2242106a4cbae8-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "f3ce4de4-623c-
46ff-b0c9-d7c57c8d0526", "children": ["450511c3-8581-4378-82c4-ab6a74bd98aa"]},
"450511c3-8581-4378-82c4-ab6a74bd98aa": {"id": "450511c3-8581-4378-82c4-ab6a74bd98aa",
"message": {"id": "450511c3-8581-4378-82c4-ab6a74bd98aa", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728853236.983871,
"update_time": null, "content": {"content_type": "text", "parts": ["The error
you&#x27;re seeing occurs because of how Node.js handles module imports. The code
currently uses the ES module syntax (`import axios from &quot;axios&quot;`), but your
environment is likely set up to use CommonJS, which uses `require()` instead. Glitch
by default runs on Node.js using CommonJS, so you&#x27;ll need to adjust the way Axios
is imported.\n\nTo fix the issue, replace the `import axios from &quot;axios&quot;;`
line with:\n\n```javascript\nconst axios = require(&quot;axios&quot;);\n```\n\nYou can
then re-save your `index.js` file, and the code should work in your current setup.
This way, you&#x27;re using `require()`, which is compatible with the CommonJS module
system that Glitch is using.\n\nAfter making this change, run the project again, and
let me know if any additional issues arise."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2d7ae-bc70-4bfc-b83b-50b3e5dd6d0c", "request_id": "8d2242106a4cbae8-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2d7ae-bc70-4bfc-b83b-50b3e5dd6d0c", "children": ["aaa246bb-0c0a-4080-ad16-
a300aa62d418"]}, "aaa246bb-0c0a-4080-ad16-a300aa62d418": {"id": "aaa246bb-0c0a-4080-
```

ad16-a300aa62d418", "message": {"id": "aaa246bb-0c0a-4080-ad16-a300aa62d418", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728853252.427501,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
yqwNxpPht8H4hhu4MKIE4gzg", "size_bytes": 76223, "width": 1360, "height": 598, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, ""]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-yqwNxpPht8H4hhu4MKIE4gzg", "size": 76223, "name": "image.png",
"mime_type": "image/png", "width": 1360, "height": 598}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d2242f24aaabb1d-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "450511c3-8581-4378-82c4-ab6a74bd98aa", "children": ["bd3536dc-ebc4-
419e-86f0-20ed31dad152"]}, "bd3536dc-ebc4-419e-86f0-20ed31dad152": {"id": "bd3536dc-
ebc4-419e-86f0-20ed31dad152", "message": {"id": "bd3536dc-ebc4-419e-86f0-20ed31dad152",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728853265.606382, "update_time": null, "content": {"content_type": "text", "parts":
["The error you&#x27;re encountering (`SyntaxError: Unexpected identifier`) suggests
there&#x27;s an issue with the way the `axios` library is being imported into your
project. In Node.js, the `import` statement is typically used with ES modules, but
Node.js commonly uses the `require` syntax for module imports unless you&#x27;re using
an ES module setup.\n\nSince your error points to using `import axios`, but the rest
of your code seems to be using CommonJS (`require` syntax), you&#x27;ll need to change
the `import` statement for `axios` to `require`.\n\nHere\u2019s what to do next:\n\n1.
Open the `index.js` file.\n2. At the very top, where you&#x27;re importing `axios`,
make sure you&#x27;re using this line:\n\n   ```javascript\n   const axios =
require(&#x27;axios&#x27;);\n   ```\n\nThis will ensure you&#x27;re using the correct
syntax for CommonJS modules in Node.js.\n\nOnce you&#x27;ve made this change, you
should also try running `npm start` again.\n\nIf you&#x27;re still encountering issues,
please double-check the package versions in your `package.json` to ensure all
dependencies are correctly installed and up-to-date."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa246bb-0c0a-4080-ad16-a300aa62d418", "request_id": "8d2242f24aaabb1d-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa246bb-0c0a-4080-ad16-a300aa62d418", "children": ["aaa21fb2-fc7d-4217-8833-
db80dee7a1be"]}, "aaa21fb2-fc7d-4217-8833-db80dee7a1be": {"id": "aaa21fb2-fc7d-4217-
8833-db80dee7a1be", "message": {"id": "aaa21fb2-fc7d-4217-8833-db80dee7a1be", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728853415.621166,
"update_time": null, "content": {"content_type": "text", "parts": ["there is no such
line\n\nconst Alexa = require(&quot;ask-sdk-core&quot;);\r\nconst axios =
require(&quot;axios&quot;);\r\n\r\nasync function askToGPT(prompt) {\r\n const
response = await axios.post(\r\n
&quot;https://api.openai.com/v1/chat/completions&quot;,\r\n   {\r\n   model:
&quot;gpt-4&quot;,\r\n   messages: [{ role: &quot;user&quot;, content:
prompt }],\r\n   max_tokens: 100,\r\n   temperature: 0.5,\r\n   },\r\n   {\r\n
headers: {\r\n   &quot;Content-Type&quot;: &quot;application/json&quot;,\r\n

```
Authorization: `Bearer
${process.env.OPENAI_API_KEY}`,\r\n      },\r\n    }\r\n  );\r\n\r\n  return
response.data.choices[0].message.content;\r\n}\r\n\r\n\r\n\r\nfunction formatString(text)
{\r\n  return text.replace(/\\n+/g, &quot; &quot;);\r\n}\r\n\r\nconst
LaunchRequestHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;LaunchRequest&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const response =\r\n      &#x27;Seja bem-vindo ao GPT-3. Voc\u00ea pode perguntar algo
como &quot;chat, qual a capital da Fran\u00e7a?&quot;&#x27;;\r\n\r\n    return
handlerInput.responseBuilder.speak(response).getResponse();\r\n  },\r\n};\r\n\r\nconst
HelloWorldIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AskToGPTIntent&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const question = Alexa.getSlotValue(\r\n      handlerInput.requestEnvelope,\r\n
&quot;question&quot;\r\n    );\r\n\r\n    const responseFromGPT = await
askToGPT(question);\r\n\r\n    const formattedResponse =
formatString(responseFromGPT);\r\n\r\n    return
handlerInput.responseBuilder.speak(formattedResponse).getResponse();\r\n  },\r\n};\r\n
\r\nconst HelpIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AMAZON.HelpIntent&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput =\r\n      &quot;Voc\u00ea pode perguntar algo como &#x27;chat,
ideia de nome para beb\u00eas&#x27;&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst CancelAndStopIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      (Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.CancelIntent&quot; ||\r\n
Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.StopIntent&quot;)\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput = &quot;At\u00e9 mais!&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst FallbackIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.FallbackIntent&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
const speakOutput =\r\n      &quot;Desculpe, n\u00e3o tenho conhecimento sobre isso.
Por favor, tente novamente.&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst SessionEndedRequestHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===\r\n
&quot;SessionEndedRequest&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n
return handlerInput.responseBuilder.getResponse();\r\n  },\r\n};\r\n\r\nconst
IntentReflectorHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;IntentRequest&quot;\r\n    );\r\n  },\r\n  handle(handlerInput) {\r\n    const
speakOutput = `Que tal me perguntar algo como &#x27;quanto est\u00e1 o d\u00f3lar
```

~~~
hoje?&#x27;`;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nco
nst ErrorHandler = {\r\n  canHandle() {\r\n    return true;\r\n  },\r\n
handle(handlerInput) {\r\n    const speakOutput =\r\n      &quot;Desculpe, parece que
tivemos um problema aqui! Acesse github.com/joao208/alexa-chatgpt para mais
informa\u00e7\u00f5es.&quot;;\r\n\r\n    return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n  },\r\n};\r\n\r\nex
ports.handler = Alexa.SkillBuilders.custom()\r\n  .addRequestHandlers(\r\n
LaunchRequestHandler,\r\n    HelloWorldIntentHandler,\r\n    HelpIntentHandler,\r\n
CancelAndStopIntentHandler,\r\n    FallbackIntentHandler,\r\n
SessionEndedRequestHandler,\r\n
IntentReflectorHandler\r\n  )\r\n  .addErrorHandlers(ErrorHandler)\r\n  .lambda();\r"]}
, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2246f41fa95246-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "bd3536dc-ebc4-
419e-86f0-20ed31dad152", "children": ["5a0fa8ad-0915-41ba-810d-f2dde3439848"]},
"5a0fa8ad-0915-41ba-810d-f2dde3439848": {"id": "5a0fa8ad-0915-41ba-810d-f2dde3439848",
"message": {"id": "5a0fa8ad-0915-41ba-810d-f2dde3439848", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728853439.339837,
"update_time": null, "content": {"content_type": "text", "parts": ["It looks like
you&#x27;re already using `require(&#x27;axios&#x27;)`, so the problem seems to lie in
how Axios is used or something else in the environment setup. The error indicates that
there might be some incompatibility or issue with how Node.js is interpreting the
code.\n\nHere\u2019s what to try next:\n\n1. **Check the Axios import**: It should be
using `require`, which you&#x27;ve already done. So you&#x27;re fine there:\n
```js\n   const axios = require(&quot;axios&quot;);\n   ```\n\n2. **Update Node.js
version**: It seems that Node.js might be running an outdated version on Glitch. Axios
uses modern JavaScript syntax, and if the Node version on Glitch is too old, this
might be causing the problem. Glitch sometimes defaults to older Node.js versions.\n\n
- Go to your `package.json` file.\n   - Add the following to specify a more recent
Node version (e.g., 16.x or later):\n\n     ```json\n     &quot;engines&quot;: {\n
&quot;node&quot;: &quot;16.x&quot;\n     }\n     ```\n\n   - Then try running `npm
install` again and restart your project.\n\n3. **Check the `package.json` file for
dependencies**: Ensure `axios` is listed under the dependencies in `package.json`:\n\n
```json\n   &quot;dependencies&quot;: {\n     &quot;axios&quot;:
&quot;^1.3.2&quot;\n   }\n   ```\n\n4. **If this doesn&#x27;t work, try clearing the
npm cache**: Sometimes package conflicts can persist. You can clear the npm cache
with:\n   ```bash\n   npm cache clean --force\n   ```\n\n5. **Final option: Try
switching to `require` for everything**: If you&#x27;re running into issues with the
environment recognizing `import`, stick with `require` throughout your code.\n\nIf
after these steps the issue persists, you can also try checking your project&#x27;s
logs for more detailed error information by running:\n\n```bash\nnpm start\n```\n\nLet
me know if these steps help or if further assistance is needed!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa21fb2-fc7d-4217-8833-db80dee7a1be", "request_id": "8d2246f41fa95246-
~~~

MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa21fb2-fc7d-4217-8833-db80dee7a1be", "children": ["aaa2a46e-1d21-43ef-9935-
7b8d4e85bba3"]}, "aaa2a46e-1d21-43ef-9935-7b8d4e85bba3": {"id": "aaa2a46e-1d21-43ef-
9935-7b8d4e85bba3", "message": {"id": "aaa2a46e-1d21-43ef-9935-7b8d4e85bba3", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728853967.598559,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
mJoIbLeCGkA9hDIFQfslskAA", "size_bytes": 53987, "width": 1360, "height": 566, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, ""]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-mJoIbLeCGkA9hDIFQfslskAA", "size": 53987, "name": "image.png",
"mime_type": "image/png", "width": 1360, "height": 566}], "serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d22546a0b3c0d65-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "5a0fa8ad-0915-41ba-810d-f2dde3439848", "children": ["88c2c38b-6741-
4bc8-8c38-058a08769c70"]}, "88c2c38b-6741-4bc8-8c38-058a08769c70": {"id": "88c2c38b-
6741-4bc8-8c38-058a08769c70", "message": {"id": "88c2c38b-6741-4bc8-8c38-058a08769c70",
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728853999.358043, "update_time": null, "content": {"content_type": "text", "parts":
["The issue you&#x27;re seeing is related to using ES6 module syntax (`import`) in a
Node.js environment that likely doesn&#x27;t support it. The error occurs because the
Node.js environment is expecting the older `require` syntax.\n\nTo fix this, you
should change the ES6 `import` statement to CommonJS `require`. Specifically,
replace:\n\n```js\nimport axios from
&#x27;./lib/axios.js&#x27;;\n```\n\nwith:\n\n```js\nconst axios =
require(&#x27;./lib/axios.js&#x27;);\n```\n\nIf you&#x27;re using the `axios` package
from npm (which it looks like you are), you should instead have:\n\n```js\nconst axios
= require(&#x27;axios&#x27;);\n```\n\nThis change should resolve the error and allow
Node.js to load the `axios` module correctly. After making this adjustment, run `npm
start` again, and it should proceed without this specific issue."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2a46e-1d21-43ef-9935-7b8d4e85bba3", "request_id": "8d22546a0b3c0d65-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2a46e-1d21-43ef-9935-7b8d4e85bba3", "children": ["aaa2d3f7-e26a-4eef-bf83-
ab710f9b20d2"]}, "aaa2d3f7-e26a-4eef-bf83-ab710f9b20d2": {"id": "aaa2d3f7-e26a-4eef-
bf83-ab710f9b20d2", "message": {"id": "aaa2d3f7-e26a-4eef-bf83-ab710f9b20d2", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728854083.750115,
"update_time": null, "content": {"content_type": "text", "parts": ["which file are
these lines likely invoked or found in "]}, "status": "finished_successfully",
"end_turn": null, "weight": 1.0, "metadata": {"serialization_metadata":
{"custom_symbol_offsets": []}, "request_id": "8d225744cbcb374f-MXP", "message_source":
null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel":
null}, "parent": "88c2c38b-6741-4bc8-8c38-058a08769c70", "children": ["e0a1a3bb-172c-
41c3-9aa0-5cd68c9bc643"]}, "e0a1a3bb-172c-41c3-9aa0-5cd68c9bc643": {"id": "e0a1a3bb-
172c-41c3-9aa0-5cd68c9bc643", "message": {"id": "e0a1a3bb-172c-41c3-9aa0-5cd68c9bc643",

```
"author": {"role": "assistant", "name": null, "metadata": {}}, "create_time":
1728854100.897175, "update_time": null, "content": {"content_type": "text", "parts":
["The lines `import axios from &#x27;./lib/axios.js&#x27;` or `const axios =
require(&#x27;axios&#x27;)` are likely found in the **index.js** file, as that is the
main entry point for most Node.js applications. Based on the previous errors you
encountered, it&#x27;s clear that the system tried to start `index.js` and encountered
the issue there.\n\nYou should check your **index.js** file and locate the line that
includes an `import` statement. Replace it with the `require` version as I outlined in
the previous response:\n\n```js\nconst axios =
require(&#x27;axios&#x27;);\n```\n\nMake sure this change is in **index.js**, and then
try running `npm start` again to see if the issue resolves."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2d3f7-e26a-4eef-bf83-ab710f9b20d2", "finish_details": {"type": "stop",
"stop_tokens": [200002]}, "is_complete": true, "request_id": "8d225744cbcb374f-MXP",
"timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent": "aaa2d3f7-
e26a-4eef-bf83-ab710f9b20d2", "children": ["aaa2f62e-895e-4bf4-a1f2-3bff0057ec54"]},
"aaa2f62e-895e-4bf4-a1f2-3bff0057ec54": {"id": "aaa2f62e-895e-4bf4-a1f2-3bff0057ec54",
"message": {"id": "aaa2f62e-895e-4bf4-a1f2-3bff0057ec54", "author": {"role": "user",
"name": null, "metadata": {}}, "create_time": 1728854177.275807, "update_time": null,
"content": {"content_type": "text", "parts": ["const Alexa = require(&quot;ask-sdk-
core&quot;);\r\nconst axios = require(&quot;axios&quot;);\r\n\r\nasync function
askToGPT(prompt) {\r\n  const response = await axios.post(\r\n
&quot;https://api.openai.com/v1/chat/completions&quot;,\r\n    {\r\n      model:
&quot;gpt-4&quot;,\r\n      messages: [{ role: &quot;user&quot;, content:
prompt }],\r\n      max_tokens: 100,\r\n      temperature: 0.5,\r\n    },\r\n    {\r\n
headers: {\r\n        &quot;Content-Type&quot;: &quot;application/json&quot;,\r\n
Authorization: `Bearer
${process.env.OPENAI_API_KEY}`,\r\n      },\r\n    }\r\n  );\r\n\r\n  return
response.data.choices[0].message.content;\r\n}\r\n\r\n\r\nfunction formatString(text)
{\r\n  return text.replace(/\\n+/g, &quot; &quot;);\r\n}\r\n\r\nconst
LaunchRequestHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;LaunchRequest&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const response =\r\n      &#x27;Seja bem-vindo ao GPT-3. Voc\u00ea pode perguntar algo
como &quot;chat, qual a capital da Fran\u00e7a?&quot;&#x27;;\r\n\r\n    return
handlerInput.responseBuilder.speak(response).getResponse();\r\n  },\r\n};\r\n\r\nconst
HelloWorldIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n      Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AskToGPTIntent&quot;\r\n    );\r\n  },\r\n  async handle(handlerInput) {\r\n
const question = Alexa.getSlotValue(\r\n      handlerInput.requestEnvelope,\r\n
&quot;question&quot;\r\n    );\r\n\r\n    const responseFromGPT = await
askToGPT(question);\r\n\r\n    const formattedResponse =
formatString(responseFromGPT);\r\n\r\n    return
handlerInput.responseBuilder.speak(formattedResponse).getResponse();\r\n  },\r\n};\r\n
\r\nconst HelpIntentHandler = {\r\n  canHandle(handlerInput) {\r\n    return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
```

&amp;&amp;\r\n   Alexa.getIntentName(handlerInput.requestEnvelope) ===
&quot;AMAZON.HelpIntent&quot;\r\n   );\r\n },\r\n handle(handlerInput) {\r\n
const speakOutput =\r\n   &quot;Voc\u00ea pode perguntar algo como &#x27;chat,
ideia de nome para beb\u00eas&#x27;&quot;;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nco
nst CancelAndStopIntentHandler = {\r\n canHandle(handlerInput) {\r\n   return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n   (Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.CancelIntent&quot; ||\r\n
Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.StopIntent&quot;)\r\n   );\r\n },\r\n handle(handlerInput) {\r\n
const speakOutput = &quot;At\u00e9 mais!&quot;;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nco
nst FallbackIntentHandler = {\r\n canHandle(handlerInput) {\r\n   return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) === &quot;IntentRequest&quot;
&amp;&amp;\r\n   Alexa.getIntentName(handlerInput.requestEnvelope) ===\r\n
&quot;AMAZON.FallbackIntent&quot;\r\n   );\r\n },\r\n handle(handlerInput) {\r\n
const speakOutput =\r\n   &quot;Desculpe, n\u00e3o tenho conhecimento sobre isso.
Por favor, tente novamente.&quot;;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nco
nst SessionEndedRequestHandler = {\r\n canHandle(handlerInput) {\r\n   return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===\r\n
&quot;SessionEndedRequest&quot;\r\n   );\r\n },\r\n handle(handlerInput) {\r\n
return handlerInput.responseBuilder.getResponse();\r\n },\r\n};\r\n\r\nconst
IntentReflectorHandler = {\r\n canHandle(handlerInput) {\r\n   return (\r\n
Alexa.getRequestType(handlerInput.requestEnvelope) ===
&quot;IntentRequest&quot;\r\n   );\r\n },\r\n handle(handlerInput) {\r\n   const
speakOutput = `Que tal me perguntar algo como &#x27;quanto est\u00e1 o d\u00f3lar
hoje?&#x27;`;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nco
nst ErrorHandler = {\r\n canHandle() {\r\n   return true;\r\n },\r\n
handle(handlerInput) {\r\n   const speakOutput =\r\n   &quot;Desculpe, parece que
tivemos um problema aqui! Acesse github.com/joao208/alexa-chatgpt para mais
informa\u00e7\u00f5es.&quot;;\r\n\r\n   return
handlerInput.responseBuilder.speak(speakOutput).getResponse();\r\n },\r\n};\r\n\r\nex
ports.handler = Alexa.SkillBuilders.custom()\r\n .addRequestHandlers(\r\n
LaunchRequestHandler,\r\n   HelloWorldIntentHandler,\r\n   HelpIntentHandler,\r\n
CancelAndStopIntentHandler,\r\n   FallbackIntentHandler,\r\n
SessionEndedRequestHandler,\r\n
IntentReflectorHandler\r\n)\r\n .addErrorHandlers(ErrorHandler)\r\n .lambda();\r\n
can"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d22598aed4d374f-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "e0a1a3bb-172c-
41c3-9aa0-5cd68c9bc643", "children": ["6e98462b-ea59-44b2-b451-fa6506196b26"]},
"6e98462b-ea59-44b2-b451-fa6506196b26": {"id": "6e98462b-ea59-44b2-b451-fa6506196b26",
"message": {"id": "6e98462b-ea59-44b2-b451-fa6506196b26", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728854195.944799,
"update_time": null, "content": {"content_type": "text", "parts": ["It seems like

```
you&#x27;re sharing your code for integrating Alexa with GPT-4. Based on your code, it
looks good at a glance, but I want to check if there&#x27;s a specific issue
you&#x27;re encountering or any question you might have with this code.\n\nIf you are
encountering the same &quot;Unexpected identifier&quot; issue or any specific error
message, make sure that:\n\n1. **You are using `require()` instead of `import`:**
Since your code uses `require()`, ensure that all modules are imported using
`require()`. In this case, you have `axios` and `ask-sdk-core` correctly required.\n
\n2. **Check for missing dependencies:** If you haven&#x27;t already done so, make
sure to run `npm install axios ask-sdk-core` to ensure that these packages are
available in your environment.\n\n3. **Environment variables:** Ensure that your
`.env` file contains the correct `OPENAI_API_KEY`. This should look something
like:\n\n   ```\n   OPENAI_API_KEY=your-api-key-here\n   ```\n\nOnce you&#x27;re done
with these, you can run `npm start` to test the integration. Let me know if you
encounter any further errors or if you&#x27;d like more guidance on a specific
issue!"]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2f62e-895e-4bf4-a1f2-3bff0057ec54", "request_id": "8d22598aed4d374f-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa2f62e-895e-4bf4-a1f2-3bff0057ec54", "children": ["aaa2d208-695c-48e4-a8ac-
64bb93b7e965"]}, "aaa2d208-695c-48e4-a8ac-64bb93b7e965": {"id": "aaa2d208-695c-48e4-
a8ac-64bb93b7e965", "message": {"id": "aaa2d208-695c-48e4-a8ac-64bb93b7e965", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728857351.712152,
"update_time": null, "content": {"content_type": "text", "parts": ["Ta6m8j
nnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnn88888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888888888888888888888888888888888888888888888888888888888888
88888888888888888888888888888]&#x27;"]}, "status": "finished_successfully", "end_turn":
null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets":
[]}, "request_id": "8d22a70a5f065237-MXP", "message_source": null, "timestamp_":
"absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent":
"6e98462b-ea59-44b2-b451-fa6506196b26", "children": ["8e1fd831-4ad8-475b-b558-
63ba33119027"]}, "8e1fd831-4ad8-475b-b558-63ba33119027": {"id": "8e1fd831-4ad8-475b-
b558-63ba33119027", "message": {"id": "8e1fd831-4ad8-475b-b558-63ba33119027", "author":
{"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728857362.034321,
"update_time": null, "content": {"content_type": "text", "parts": ["It seems like you
typed in a lot of random characters! Can I assist you with something specific? Feel
free to ask about your project or any other questions you may have."]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2d208-695c-48e4-a8ac-64bb93b7e965", "request_id": "8d22a70a5f065237-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa2d208-695c-48e4-a8ac-64bb93b7e965", "children": ["aaa22d83-a0a8-4e36-8247-
```

```
c89dccca1b49"]}, "aaa22d83-a0a8-4e36-8247-c89dccca1b49": {"id": "aaa22d83-a0a8-4e36-
8247-c89dccca1b49", "message": {"id": "aaa22d83-a0a8-4e36-8247-c89dccca1b49", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728942223.967561,
"update_time": null, "content": {"content_type": "text", "parts": ["i think npm start
worked. not sure tho. what next? "]}, "status": "finished_successfully", "end_turn":
null, "weight": 1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets":
[]}, "request_id": "8d2abf1e4bef4be1-MXP", "message_source": null, "timestamp_":
"absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent":
"8e1fd831-4ad8-475b-b558-63ba33119027", "children": ["86222d0e-4227-4797-acfd-
f545244e0b09"]}, "86222d0e-4227-4797-acfd-f545244e0b09": {"id": "86222d0e-4227-4797-
acfd-f545244e0b09", "message": {"id": "86222d0e-4227-4797-acfd-f545244e0b09", "author":
{"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728942248.970061,
"update_time": null, "content": {"content_type": "text", "parts": ["If you think `npm
start` worked successfully, here&#x27;s what you can do next to verify the
functionality of your project:\n\n### Steps to Test and Deploy:\n\n1. **Test
Locally**:\n   - Check the logs or console output to ensure the application started
without errors. If you see something like &quot;Server running on port...&quot;, it
means your application is running fine.\n   \n2. **Simulate an Alexa Request**:\n   -
You can use the **Alexa Developer Console** to test your skill without deploying it to
a real device. Sign in to the [Alexa Developer
Console](https://developer.amazon.com/alexa/console/ask), find your skill, and use the
**Test** tab.\n   - Alternatively, you can try sending a sample request to your
deployed URL (if hosted online) to see if it\u2019s working as expected.\n\n3. **Check
Logs**:\n   - If the application is running on Glitch or another service, monitor the
**logs** section to see if any errors or issues are occurring when the skill is
invoked.\n\n4. **Deploy to Alexa**:\n   - Assuming you want to connect this to Alexa,
make sure to **link your skill** to the OpenAI API by ensuring that your Alexa
Developer Console is correctly configured with your skill\u2019s endpoint.\n   - Use
`ask deploy` from the Alexa Skills Kit CLI to deploy the skill to your developer
account.\n\n5. **Invoke the Skill**:\n   - Test the skill by interacting with it via
your Alexa device or the Alexa simulator. Use any intents you have defined, such as
\u201cAskToGPTIntent,\u201d to ensure that Alexa is communicating properly with GPT-4
through the OpenAI API.\n\nLet me know how it goes! If you encounter any errors, we
can troubleshoot them further."]}, "status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true,
"message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa22d83-a0a8-4e36-8247-c89dccca1b49", "request_id": "8d2abf1e4bef4be1-
MXP", "timestamp_": "absolute"}, "recipient": "all", "channel": null}, "parent":
"aaa22d83-a0a8-4e36-8247-c89dccca1b49", "children": ["aaa2ee7c-b536-4a7d-824f-
4087327072df"]}, "aaa2ee7c-b536-4a7d-824f-4087327072df": {"id": "aaa2ee7c-b536-4a7d-
824f-4087327072df", "message": {"id": "aaa2ee7c-b536-4a7d-824f-4087327072df", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728942314.067707,
"update_time": null, "content": {"content_type": "text", "parts": ["how do i simulate
an alexa request? "]}, "status": "finished_successfully", "end_turn": null, "weight":
1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []},
"request_id": "8d2ac154cc814bed-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "86222d0e-4227-
4797-acfd-f545244e0b09", "children": ["47a9ea5f-2c8c-4f72-b9b0-b953b89b2e0f"]},
"47a9ea5f-2c8c-4f72-b9b0-b953b89b2e0f": {"id": "47a9ea5f-2c8c-4f72-b9b0-b953b89b2e0f",
"message": {"id": "47a9ea5f-2c8c-4f72-b9b0-b953b89b2e0f", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728942343.362646,
"update_time": null, "content": {"content_type": "text", "parts": ["To simulate an
Alexa request, you can use the **Alexa Developer Console**&#x27;s built-in testing
tools. Here&#x27;s a step-by-step guide on how to simulate requests and test your
skill:\n\n### Steps to Simulate an Alexa Request:\n\n1. **Sign In to the Alexa
Developer Console**:\n   - Go to the [Alexa Developer
Console](https://developer.amazon.com/alexa/console/ask).\n   - Sign in with your
Amazon account.\n\n2. **Select Your Skill**:\n   - Once you are logged in, navigate to
the **Skills** tab in the console.\n   - Select the skill you want to test (which
should be the one you&#x27;re developing with the Alexa-GPT integration).\n\n3.
**Navigate to the Test Tab**:\n   - In the skill dashboard, click on the **Test** tab
at the top.\n   - Ensure that the skill is in **Development** mode. You can switch to
**Development** by toggling the switch to **Enabled**.\n\n4. **Enter Utterances**:\n
- In the **Test** tab, under **Alexa Simulator**, you can simulate user requests.\n
- In the text box where it says &quot;Enter Utterance,&quot; type in the request you
want to test (e.g., &quot;Ask GPT, what is the capital of France?&quot;).\n   - You
can also test your invocation name (e.g., &quot;Open [Your Skill Name]&quot;) to start
the skill.\n\n5. **Monitor the Responses**:\n   - After typing your utterance, Alexa
will simulate the response. You\u2019ll see both the text and voice response Alexa
would give.\n   - If your skill makes an external API request (such as to GPT-4 via
OpenAI), ensure that it is working correctly and returning the expected output.\n\n6.
**Test with Voice**:\n   - You can also simulate voice input by clicking the
**Microphone** button and speaking the utterance you want to test.\n\n7. **View JSON
Input and Output**:\n   - If you want to see the detailed request/response interaction,
scroll down to view the **JSON Input** (what Alexa sends to your skill) and **JSON
Output** (what your skill returns to Alexa). This can help in debugging or verifying
the data flow.\n\n8. **Review Logs** (Optional):\n   - If you&#x27;re using Glitch or
another hosting platform, check the logs in your Glitch project for any errors or
issues when handling the requests.\n\nOnce the skill works as expected in the Alexa
Developer Console, you can also test it on your Alexa-enabled devices if the skill is
enabled for your account.\n\nLet me know if you encounter any issues while testing!"]},
"status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2ee7c-b536-4a7d-824f-4087327072df", "request_id": "8d2ac154cc814bed-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2ee7c-b536-4a7d-824f-4087327072df", "children": ["aaa2a8c1-e445-4185-96db-
1a44997f6c24"]}, "aaa2a8c1-e445-4185-96db-1a44997f6c24": {"id": "aaa2a8c1-e445-4185-
96db-1a44997f6c24", "message": {"id": "aaa2a8c1-e445-4185-96db-1a44997f6c24", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728942381.030035,
"update_time": null, "content": {"content_type": "text", "parts": ["i&#x27;m using
glitch "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2ac2f569990e46-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "47a9ea5f-2c8c-
4f72-b9b0-b953b89b2e0f", "children": ["cb7f1646-f4a2-49be-b415-ae46a83435eb"]},
"cb7f1646-f4a2-49be-b415-ae46a83435eb": {"id": "cb7f1646-f4a2-49be-b415-ae46a83435eb",
"message": {"id": "cb7f1646-f4a2-49be-b415-ae46a83435eb", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728942407.244267,
"update_time": null, "content": {"content_type": "text", "parts": ["If you&#x27;re
using **Glitch** to host your Alexa skill and want to simulate an Alexa request, you
can still use the **Alexa Developer Console** as described earlier, but now
you\u2019ll need to connect the skill to the Glitch server. Here\u2019s how you can
test your skill using Glitch:\n\n### Steps to Simulate an Alexa Request While Hosting
on Glitch:\n\n1. **Deploy Your Code on Glitch**:\n   - Ensure your project is running
on Glitch, and the code is deployed.\n   - If the project is running, you should see a
live app URL like `https://[project-name].glitch.me`.\n\n2. **Create a Custom Alexa
Skill** (if you haven\u2019t already):\n   - In the [Alexa Developer
Console](https://developer.amazon.com/alexa/console/ask), create a new skill.\n   -
When prompted for a skill type, select **Custom**.\n   - Set up an invocation name,
which is how users will trigger your skill (e.g., &quot;ask GPT&quot;).\n\n3. **Set
the Endpoint to Your Glitch URL**:\n   - In the **Endpoint** section of your skill in
the Alexa Developer Console, choose **HTTPS**.\n   - Paste the Glitch live URL (e.g.,
`https://[your-project-name].glitch.me`) as the endpoint URL.\n\n4. **Configure SSL
Certificates**:\n   - Since Glitch uses `https`, Alexa will recognize the SSL
certificate, and you can select &quot;My development endpoint is a subdomain of a
domain that has a wildcard certificate from a certificate authority.&quot;\n\n5.
**Enable Skill for Testing**:\n   - In the **Test** section of the Alexa Developer
Console, toggle the switch to **Enable** for testing.\n   - Make sure you\u2019re in
**Development** mode.\n\n6. **Simulate Alexa Requests**:\n   - As described earlier,
you can type test utterances (e.g., \u201cAsk GPT what the capital of France is\u201d)
in the **Test** section of the Alexa Developer Console.\n   - This will send requests
to your Glitch-hosted app, and you can see the responses in the console.\n\n7. **Check
Logs on Glitch**:\n   - While testing, you can open the **Logs** tab in your Glitch
project to see what\u2019s happening when your skill receives requests from Alexa.\n
- This will help you debug if there are any issues with handling requests or
responses.\n\nLet me know how it goes or if you need further help!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2a8c1-e445-4185-96db-1a44997f6c24", "request_id": "8d2ac2f569990e46-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2a8c1-e445-4185-96db-1a44997f6c24", "children": ["aaa23ee3-a132-40f4-8f3f-
ac3a751a1f24"]}, "aaa23ee3-a132-40f4-8f3f-ac3a751a1f24": {"id": "aaa23ee3-a132-40f4-
8f3f-ac3a751a1f24", "message": {"id": "aaa23ee3-a132-40f4-8f3f-ac3a751a1f24", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728942873.130404,
"update_time": null, "content": {"content_type": "multimodal_text", "parts":
[{"content_type": "image_asset_pointer", "asset_pointer": "file-service://file-
sCJ2J5BpkyifhVlxP2kmhoWY", "size_bytes": 33286, "width": 1320, "height": 563, "fovea":
null, "metadata": {"dalle": null, "gizmo": null, "emu_omit_glimpse_image": null,
"emu_patches_override": null, "sanitized": true}}, "now what? "]}, "status":
"finished_successfully", "end_turn": null, "weight": 1.0, "metadata": {"attachments":
[{"id": "file-sCJ2J5BpkyifhVlxP2kmhoWY", "size": 33286, "name": "image.png",
"mime_type": "image/png", "width": 1320, "height": 563}], "serialization_metadata":
```

{"custom_symbol_offsets": []}, "request_id": "8d2acef6a80b0f66-MXP", "message_source": null, "timestamp_": "absolute", "message_type": null}, "recipient": "all", "channel": null}, "parent": "cb7f1646-f4a2-49be-b415-ae46a83435eb", "children": ["39e64d0c-69dd-4617-a495-319e6b999192"]}, "39e64d0c-69dd-4617-a495-319e6b999192": {"id": "39e64d0c-69dd-4617-a495-319e6b999192", "message": {"id": "39e64d0c-69dd-4617-a495-319e6b999192", "author": {"role": "assistant", "name": null, "metadata": {}}, "create_time": 1728942906.815694, "update_time": null, "content": {"content_type": "text", "parts": ["It seems like your code started successfully but you need to trigger or simulate an Alexa request to see how it behaves.\n\nSince you&#x27;re using Glitch, the easiest way to simulate an Alexa request without an Echo device is through **Postman** or other similar API testing tools. Here&#x27;s how you can do it:\n\n### Steps to Simulate Alexa Request via Postman:\n1. **Install Postman**: If you don&#x27;t have it installed, download it from [here](https://www.postman.com/downloads/).\n   \n2. **Create a New Request**:\n   - Open Postman and click on `New` to create a new HTTP request.\n   - Choose **POST** as the HTTP method.\n\n3. **Set the URL**: \n   Set the Glitch project URL followed by `/`. For example: \n   ```\n   https://magic-bramble-lobe.glitch.me/\n   ```\n\n4. **Set Headers**:\n   Add these headers:\n   - `Content-Type: application/json`\n\n5. **Set the Body**:\n   - In Postman, select the `Body` tab.\n   - Choose `raw` and set it to `JSON`.\n   - Copy this JSON example to simulate a simple Alexa intent request:\n   ```json\n   {\n &quot;version&quot;: &quot;1.0&quot;,\n   &quot;session&quot;: {\n &quot;new&quot;: true,\n   &quot;sessionId&quot;: &quot;amzn1.echo-api.session.test&quot;,\n   &quot;application&quot;: {\n &quot;applicationId&quot;: &quot;amzn1.ask.skill.test&quot;\n   },\n &quot;user&quot;: {\n   &quot;userId&quot;: &quot;amzn1.ask.account.testUser&quot;\n   }\n   },\n &quot;context&quot;: {\n   &quot;System&quot;: {\n &quot;application&quot;: {\n   &quot;applicationId&quot;: &quot;amzn1.ask.skill.test&quot;\n   },\n   &quot;user&quot;: {\n &quot;userId&quot;: &quot;amzn1.ask.account.testUser&quot;\n   }\n   }\n   },\n &quot;request&quot;: {\n   &quot;type&quot;: &quot;IntentRequest&quot;,\n &quot;intent&quot;: {\n   &quot;name&quot;: &quot;AskToGPTIntent&quot;,\n &quot;slots&quot;: {\n   &quot;question&quot;: {\n &quot;name&quot;: &quot;question&quot;,\n   &quot;value&quot;: &quot;What is the capital of France?&quot;\n   }\n   }\n   }\n   }\n   }\n ```\n\n6. **Send the Request**:\n   - Press `Send` and check the response from your Glitch project.\n\nIf everything is configured correctly, you should get a response from your skill, simulating how Alexa would respond.\n\nLet me know if this works or if you run into any issues!"]}, "status": "finished_successfully","end_turn": true , "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "aaa23ee3-a132-40f4-8f3f-ac3a751a1f24", "request_id": "8d2acef6a80b0f66-MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens": [200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent": "aaa23ee3-a132-40f4-8f3f-ac3a751a1f24", "children": ["aaa20d92-c4aa-4d54-8917-8a65c13e3c6b"]}, "aaa20d92-c4aa-4d54-8917-8a65c13e3c6b": {"id": "aaa20d92-c4aa-4d54-8917-8a65c13e3c6b", "message": {"id": "aaa20d92-c4aa-4d54-8917-8a65c13e3c6b", "author":

~~~
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728943025.356064,
"update_time": null, "content": {"content_type": "text", "parts": ["i&#x27;m using web
postman"]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2ad2b21b58bb19-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "39e64d0c-69dd-
4617-a495-319e6b999192", "children": ["f33733c8-2c98-4a9e-b46e-d9812e9c88dd"]},
"f33733c8-2c98-4a9e-b46e-d9812e9c88dd": {"id": "f33733c8-2c98-4a9e-b46e-d9812e9c88dd",
"message": {"id": "f33733c8-2c98-4a9e-b46e-d9812e9c88dd", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728943060.459612,
"update_time": null, "content": {"content_type": "text", "parts": ["For **web
Postman**, you will follow very similar steps to those in the Postman app. Here&#x27;s
how to do it:\n\n### 1. **Create a New Request** in Web Postman:\n   - Log into your
[Postman Web app](https://web.postman.co/) account.\n   - Click the `New` button, then
select **HTTP Request**.\n\n### 2. **Set the URL**:\n   - In the URL field, type your
Glitch project URL with a trailing `/`. For example:\n     ```\n     https://magic-
bramble-lobe.glitch.me/\n     ```\n\n### 3. **Set the Request Type**:\n   - Change the
request method to **POST** from the dropdown next to the URL.\n\n### 4. **Add
Headers**:\n   - Scroll down to the `Headers` section.\n   - Add the following key-
value pair:\n     ```\n     Key: Content-Type\n     Value: application/json\n
```\n\n### 5. **Set the Body**:\n   - Scroll down to the `Body` tab.\n   - Select
`raw`, then choose `JSON` from the dropdown on the right.\n   - Paste the following
JSON into the body field:\n     ```json\n     {\n       &quot;version&quot;:
&quot;1.0&quot;,\n       &quot;session&quot;: {\n         &quot;new&quot;: true,\n
&quot;sessionId&quot;: &quot;amzn1.echo-api.session.test&quot;,\n
&quot;application&quot;: {\n           &quot;applicationId&quot;:
&quot;amzn1.ask.skill.test&quot;\n         },\n         &quot;user&quot;: {\n
&quot;userId&quot;: &quot;amzn1.ask.account.testUser&quot;\n         }\n       },\n
&quot;context&quot;: {\n         &quot;System&quot;: {\n
&quot;application&quot;: {\n             &quot;applicationId&quot;:
&quot;amzn1.ask.skill.test&quot;\n           },\n           &quot;user&quot;: {\n
&quot;userId&quot;:
&quot;amzn1.ask.account.testUser&quot;\n           }\n         }\n       },\n
&quot;request&quot;: {\n         &quot;type&quot;: &quot;IntentRequest&quot;,\n
&quot;intent&quot;: {\n           &quot;name&quot;: &quot;AskToGPTIntent&quot;,\n
&quot;slots&quot;: {\n             &quot;question&quot;: {\n
&quot;name&quot;: &quot;question&quot;,\n               &quot;value&quot;: &quot;What
is the capital of
France?&quot;\n             }\n           }\n         }\n       }\n     }\n
```\n\n### 6. **Send the Request**:\n   - Once you\u2019ve set everything up, click on
the `Send` button.\n   - You should see a response from your Glitch project simulating
the Alexa skill&#x27;s response.\n\nThis will help you test your Alexa skill locally
through Postman by sending requests to your Glitch-hosted project. If everything is
working correctly, you should get a response from your Alexa skill!\n\nLet me know if
you encounter any issues or need further clarification!"]}, "status":
"finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa20d92-c4aa-4d54-8917-8a65c13e3c6b", "request_id": "8d2ad2b21b58bb19-
~~~

```
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa20d92-c4aa-4d54-8917-8a65c13e3c6b", "children": ["aaa271ee-5e98-43e2-93ca-
727ca893af4f"]}, "aaa271ee-5e98-43e2-93ca-727ca893af4f": {"id": "aaa271ee-5e98-43e2-
93ca-727ca893af4f", "message": {"id": "aaa271ee-5e98-43e2-93ca-727ca893af4f", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728943370.134179,
"update_time": null, "content": {"content_type": "text", "parts": ["got a cloud agen
error cause request was taking longer than 30 secs. i don&#x27;t know. but what if i
took the code from glitch and uploaded it to developer    console into a new skill
could it work? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2adb1cc9915a37-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "f33733c8-2c98-
4a9e-b46e-d9812e9c88dd", "children": ["46945d77-8253-454d-935e-853bea946368"]},
"46945d77-8253-454d-935e-853bea946368": {"id": "46945d77-8253-454d-935e-853bea946368",
"message": {"id": "46945d77-8253-454d-935e-853bea946368", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728943408.110845,
"update_time": null, "content": {"content_type": "text", "parts": ["Yes, you can take
the code from Glitch and upload it directly to the Amazon Developer Console to create
a new Alexa skill. Here&#x27;s how you can do it:\n\n### Steps to Upload Code to
Developer Console:\n\n1. **Open the Amazon Developer Console**:\n   - Log in to your
[Amazon Developer Console](https://developer.amazon.com/alexa/console/ask).\n\n2.
**Create a New Alexa Skill**:\n   - In the console, click **Create Skill**.\n   - Name
your skill, choose the default language, and select the **Custom** model.\n\n3.
**Choose a Method for Hosting the Skill**:\n   - When asked about backend services,
choose **Provision Your Own**.\n   - This allows you to upload your code manually via
an AWS Lambda function or a custom endpoint.\n\n4. **Set Up an AWS Lambda Function
(Optional)**:\n   - You can upload your skill to AWS Lambda if you want it to be
hosted in the cloud.\n   - Set up a Lambda function in AWS, and you can paste the
Node.js code there.\n   - Once done, copy the **Lambda ARN** and paste it into the
Alexa Developer Console in the **Endpoint** section of your skill.\n\n5. **Upload Your
Code**:\n   - Go to the **Code** section of your skill in the Developer Console.\n   -
Here you can directly upload or link to the backend code. Since you\u2019ve been
working in Node.js on Glitch, you can copy and paste that code into the appropriate
place.\n\n6. **Add Your API Key**:\n   - Make sure your `.env` file or environment
variables are set properly.\n   - Add your OpenAI API key and any other configurations
to the Lambda function or server endpoint.\n\n7. **Test Your Skill**:\n   - In the
**Test** section of the Developer Console, enable testing for your skill and simulate
requests.\n   - You can input the text or voice commands you want Alexa to
process.\n\n8. **Deploy the Skill**:\n   - Once testing works, you can submit the
skill for certification to make it publicly available.\n\n### Potential Issues:\n-
**Long Response Time**: If your code takes more than 30 seconds to respond, as in the
case of cloud timeout errors on Glitch, you may need to optimize or paginate the API
requests to ensure Alexa doesn&#x27;t time out.\n- **Function Timeouts**: AWS Lambda
functions have a default timeout of 3 seconds, but this can be extended up to 15
minutes (make sure to increase the timeout in the Lambda configuration if
necessary).\n\nBy following these steps, you can move your Glitch-hosted code to the
official Alexa platform, test it, and deploy it as a fully functional skill."]},
"status": "finished_successfully","end_turn": true
```

```
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa271ee-5e98-43e2-93ca-727ca893af4f", "request_id": "8d2adb1cc9915a37-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa271ee-5e98-43e2-93ca-727ca893af4f", "children": ["aaa2d338-b656-46a8-9f87-
2442b2d8e93e"]}, "aaa2d338-b656-46a8-9f87-2442b2d8e93e": {"id": "aaa2d338-b656-46a8-
9f87-2442b2d8e93e", "message": {"id": "aaa2d338-b656-46a8-9f87-2442b2d8e93e", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728943487.431733,
"update_time": null, "content": {"content_type": "text", "parts": ["but how can we
know if the code is legit or now before going to the trouble of moving it? "]},
"status": "finished_successfully", "end_turn": null, "weight": 1.0, "metadata":
{"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2addf919de0e71-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "46945d77-8253-
454d-935e-853bea946368", "children": ["9cb0badc-45fa-482e-a801-ccf0c9227cc3"]},
"9cb0badc-45fa-482e-a801-ccf0c9227cc3": {"id": "9cb0badc-45fa-482e-a801-ccf0c9227cc3",
"message": {"id": "9cb0badc-45fa-482e-a801-ccf0c9227cc3", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728943517.088734,
"update_time": null, "content": {"content_type": "text", "parts": ["Before moving your
code to the Amazon Developer Console, you can test it locally or in a cloud
environment (like Glitch or Postman) to ensure that it works as expected. Here are
steps you can take to verify your code:\n\n### 1. **Test API Calls Locally
(Postman)**:\n   - Use **Postman** or a similar tool to send requests to your OpenAI
API with the same format your Alexa skill would use.\n   - Check for proper responses
from the GPT-4 API. This ensures that the API integration works before involving
Alexa.\n\n### 2. **Local Simulation via Node.js**:\n   - Run your code locally by
simulating a request.\n   - Make sure you have Node.js installed on your machine.\n
- You can run your `index.js` locally with commands like `node index.js` to see if it
throws any errors or returns the expected responses.\n\n### 3. **Using the Alexa
Simulator**:\n   - In the **Alexa Developer Console**, you can create a bare-bones
version of your skill without fully deploying it.\n   - Use the **Alexa Simulator** to
test the voice interactions and see how the backend responds.\n   - You don\u2019t
need to submit the skill for certification to test it.\n\n### 4. **Check Logs on
Glitch or Lambda**:\n   - Ensure your code is free of errors by checking the logs in
Glitch or AWS Lambda after running it.\n   - Glitch shows logs in the console, and
Lambda logs are available via **CloudWatch**.\n   - This will help identify any
runtime issues, such as missing environment variables or errors with axios
requests.\n\n### 5. **Breakpoints/Debugging**:\n   - Use `console.log` statements to
track where the request might be failing.\n   - For example, log the response from the
GPT-4 API before sending it back to Alexa to ensure you\u2019re getting the expected
output.\n\n### 6. **Time Out Issues**:\n   - If requests are taking longer than
expected (like what happened in Postman), try to reduce the `max_tokens` in your
OpenAI API call or optimize how you handle API responses.\n   - If the issue persists,
it could be an issue with the environment (e.g., Glitch limitations), in which case
deploying to Lambda might be better for response times.\n\nIf your code passes these
tests without significant issues, it\u2019s more likely that it will work properly
when moved to the Alexa Developer Console."]}, "status":
"finished_successfully","end_turn": true
```

, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa2d338-b656-46a8-9f87-2442b2d8e93e", "request_id": "8d2addf919de0e71-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa2d338-b656-46a8-9f87-2442b2d8e93e", "children": ["aaa29b7b-d6fd-4c8d-a5d0-
420d77656bc9"]}, "aaa29b7b-d6fd-4c8d-a5d0-420d77656bc9": {"id": "aaa29b7b-d6fd-4c8d-
a5d0-420d77656bc9", "message": {"id": "aaa29b7b-d6fd-4c8d-a5d0-420d77656bc9", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728943903.256522,
"update_time": null, "content": {"content_type": "text", "parts": ["this is what the
developer website says: \n\n \r\n\r\nConnect your own LLM with Alexa devices and
experiences\r\n\r\nIn addition, we understand that you may have already purpose-built
your own LLMs or leveraged LLMs offered through services like Amazon Bedrock. When you
want to bring your LLM-based experiences to life through expressive voice or ambient
visuals, and distribute them to Alexa\u2019s devices, you will have the ability to
integrate your Alexa Skill with one or more LLMs of your choice with a new tool. To
integrate third-party LLMs, all you\u2019ll have to do is integrate with automatic
speech recognition (ASR) or utterance text using a simple SDK that supports the entire
range of multimodal experiences enabled by Alexa. \r\n\r\nWe\u2019re excited about the
experiences being created by developers, and we\u2019ll have more to share soon from
companies like Character.AI, Splash, BMW, Volley, Sleep Jar, Voice Apps, and OpenTable
who are enabling customers to do things like talk to fictional characters, create
their own songs, play single and multi-agent driven games on Alexa-enabled devices,
learn about the features of their vehicle and so much more. \nwhat does that mean for
me? "]}, "status": "finished_successfully", "end_turn": null, "weight": 1.0,
"metadata": {"serialization_metadata": {"custom_symbol_offsets": []}, "request_id":
"8d2ae81f7af54c5e-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "9cb0badc-45fa-
482e-a801-ccf0c9227cc3", "children": ["5c1d583d-a568-4867-9dbc-b2acbcb8553d"]},
"5c1d583d-a568-4867-9dbc-b2acbcb8553d": {"id": "5c1d583d-a568-4867-9dbc-b2acbcb8553d",
"message": {"id": "5c1d583d-a568-4867-9dbc-b2acbcb8553d", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728943929.346139,
"update_time": null, "content": {"content_type": "text", "parts": ["The statement from
the developer website means that Amazon is opening up new opportunities for developers
to integrate third-party Large Language Models (LLMs) into Alexa experiences. This is
particularly useful if you have developed or want to integrate other LLMs (like GPT-4
or Amazon Bedrock services) into Alexa.\n\n### Key Implications for You:\n1. **Alexa
Skill Integration with LLMs**: \n   - You will be able to use your own LLM (like GPT-4)
to enhance Alexa\u2019s skills by providing a seamless integration with Alexa\u2019s
automatic speech recognition (ASR) and text-based utterance handling.\n   - This
allows you to leverage your own LLM\u2019s capabilities for conversations,
personalized responses, or other AI-driven interactions while benefiting from
Alexa&#x27;s expressive voice and multi-modal experience (such as voice, text, and
visuals).\n\n2. **Multimodal Support**:\n   - The SDK they mention will allow for a
range of experiences, including text, voice, and possibly visuals, so you can enrich
user interactions with your custom model in various formats on Alexa-enabled
devices.\n\n3. **Expanding Alexa\u2019s Capabilities**:\n   - This provides the
potential to build more complex and personalized interactions by allowing Alexa to
\u201coutsource\u201d responses or conversational logic to your own custom AI, which

```
can be particularly powerful for things like voice assistants or customer interactions
powered by your own GPT-4-based backend.\n\n### What to Do Next:\n- **Wait for the
SDK**: It seems Amazon will soon provide a specific SDK to handle this integration.
Keep an eye on announcements from Amazon for the release of the SDK.\n- **Plan Your
Integration**: In the meantime, you can prepare your LLM-powered Alexa skill by
focusing on how your LLM (e.g., GPT-4) could handle user conversations in ways that
leverage Alexa\u2019s voice capabilities. \n- **Stay Updated**: Amazon will share more
use cases from companies like OpenTable and BMW, which might give you further ideas
for how to leverage this functionality.\n\nOverall, this opens up exciting
possibilities for bringing advanced AI-powered features into your Alexa skills!"]},
"status": "finished_successfully","end_turn": true
, "weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "aaa29b7b-d6fd-4c8d-a5d0-420d77656bc9", "request_id": "8d2ae81f7af54c5e-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200002]}, "is_complete": true}, "recipient": "all", "channel": null}, "parent":
"aaa29b7b-d6fd-4c8d-a5d0-420d77656bc9", "children": ["aaa261b3-d62a-44d1-a529-
2cd12cd7a5f8"]}, "aaa261b3-d62a-44d1-a529-2cd12cd7a5f8": {"id": "aaa261b3-d62a-44d1-
a529-2cd12cd7a5f8", "message": {"id": "aaa261b3-d62a-44d1-a529-2cd12cd7a5f8", "author":
{"role": "user", "name": null, "metadata": {}}, "create_time": 1728944036.140033,
"update_time": null, "content": {"content_type": "text", "parts": ["can you check out
the data from this page: \n\nhttps://developer.amazon.com/en-US/blogs/alexa/alexa-
skills-kit/2023/09/alexa-llm-fall-devices-services-sep-2023\n\nsnd tell me whats
available today?"]}, "status": "finished_successfully", "end_turn": null, "weight":
1.0, "metadata": {"serialization_metadata": {"custom_symbol_offsets": []},
"request_id": "8d2aeb5efa67524c-MXP", "message_source": null, "timestamp_": "absolute",
"message_type": null}, "recipient": "all", "channel": null}, "parent": "5c1d583d-a568-
4867-9dbc-b2acbcb8553d", "children": ["72728eb8-4f49-4101-bc98-2d05a9d9b3c2"]},
"72728eb8-4f49-4101-bc98-2d05a9d9b3c2": {"id": "72728eb8-4f49-4101-bc98-2d05a9d9b3c2",
"message": {"id": "72728eb8-4f49-4101-bc98-2d05a9d9b3c2", "author": {"role":
"assistant", "name": null, "metadata": {}}, "create_time": 1728944074.278539,
"update_time": null, "content": {"content_type": "code", "language": "unknown",
"response_format_name": null, "text": "search(&quot;Alexa LLM fall devices services
sep 2023 site:developer.amazon.com&quot;)"}, "status": "finished_successfully",
"end_turn": false, "weight": 1.0, "metadata": {"citations": [], "content_references":
[], "gizmo_id": null, "message_type": null, "model_slug": "gpt-4o",
"default_model_slug": "gpt-4o", "parent_id": "aaa261b3-d62a-44d1-a529-2cd12cd7a5f8",
"finish_details": {"type": "stop", "stop_tokens": [200007]}, "is_complete": true,
"request_id": "8d2aeb5efa67524c-MXP", "timestamp_": "absolute"}, "recipient":
"browser", "channel": null}, "parent": "aaa261b3-d62a-44d1-a529-2cd12cd7a5f8",
"children": ["9a904751-ba2e-4bf0-99f9-1186d377282f"]}, "9a904751-ba2e-4bf0-99f9-
1186d377282f": {"id": "9a904751-ba2e-4bf0-99f9-1186d377282f", "message": {"id":
"9a904751-ba2e-4bf0-99f9-1186d377282f", "author": {"role": "tool", "name": "browser",
"metadata": {}}, "create_time": 1728944074.278655, "update_time": null, "content":
{"content_type": "tether_browsing_display", "result": "# \u30100\u2020Amazon announces
LLM-powered tools for developers at the Devices
and ...\u2020developer.amazon.com\u3011\nTo integrate APIs with Alexa\u2019s LLM, you
will provide: 1) your API endpoints and definitions 2) natural language annotations
and descriptions of your APIs and business logic and 3) CX examples for when to call
```

the APIs. This new integration model allows you to build experiences that enable customers to participate in multi-turn, cross-skill ...\n# \u30101\u2020Alexa AI | Build LLM-powered Alexa experiences - developer.amazon.com\u2020developer.amazon.com\u3011\nCreate more natural, conversational and intuitive Alexa experiences. Developers will be able to integrate their APIs or content into Alexa\u2019s LLM and create experiences that allow customers to access real-time data, enjoy immersive generative AI-enhanced games, perform tasks like booking restaurant reservations, getting a succinct summary of a ...\n# \u30102\u2020Alexa Smart Home Insider - developer.amazon.com\u2020developer.amazon.com\u3011\nBuild with Amazon\u2019s newest devices and services ... September 20, 2023. Sep 20, 2023. New developer tools to build LLM-powered experiences with Alexa. ... Our global network of Alexa Solution Providers works together to provide strategy and development support for Alexa devices and skills. Explore more. Find the latest resources, guides, code ...\n# \u30103\u20203 questions with Amazon leader Jessica Brase on how Amazon Enterprise ...\u2020developer.amazon.com\u3011\nJessica Brase is the business leader of Alexa Enterprise Services (AES), an organization within Amazon that provides 24/7 technical expertise to Alexa\u2019s enterprise partners around the world. Brase joined Amazon in January 2022 and after working in a role focused on financial operations, she decided to pursue a role leading the newly formed ...\n# \u30104\u2020Amazon showcases new customer experiences powered by generative AI at ...\u2020developer.amazon.com\u3011\nThese experiences leverage our new LLM-powered tools announced at our Fall Devices and Services launch event. With these tools, developers will be able to create experiences that allow customers to access real-time data, enjoy immersive generative AI-enhanced games, perform tasks like booking restaurant reservations, getting a succinct summary of a trending news story, and more.\n# \u30105\u2020Amazon Developers\u2020www.developer.amazon.com\u3011\nSeptember 20, 2023. Sep 20, 2023. New developer tools to build LLM-powered experiences with Alexa. Smart Home Skills. Alexa Skills. Smart Home API. July 18, 2023. Jul 18, 2023. ... Registration is easy, free, and the first step to growing your business with Amazon devices and services. Create account. Back to top. Legal. Agreements and Terms;\n# \u30106\u2020LLM\u3092\u6d3b\u7528\u3057\u305fAlexa\u30a8\u30af\u30b9\u30da\u30ea\u30a8\u30f3\u30b9\u3092\u4f5c\u6210\u3059\u308b\u305f\u3081\u306e\u65b0\u3057\u3044\u958b\u767a\u8005\u30c4\u30fc\u30eb\u2020developer.amazon.com\u3011\n\u30b3\u30f3\u30c6\u30f3\u30c4\u3084API\u3092Alexa\u306eLLM\u3068\u7d71\u5408\u3059\u308b\u3053\u3068\u3067\u3001Alexa\u642d\u8f09\u30c7\u30d0\u30a4\u30b9\u5411\u3051\u306e\u4f1a\u8a71\u578b\u30a8\u30af\u30b9\u30da\u30ea\u30a8\u30f3\u30b9\u3092\u4f5c\u6210\u3067\u304d\u307e\u3059\u3002. \u5225\u306e\u65b9\u6cd5\u3068\u3057\u3066\u3001\u4efb\u610f\u306eLLM\u3068\u7d71\u5408\u3059\u308b\u3053\u3068\u3082\u3067\u304d\u307e\u3059\u3002. \u307e\u305f\u3001\u76f4\u611f\u7684\u306a\u30b9\u30de\u30fc\u30c8\u30db\u30fc\u30e0\u5236\u5fa1\u3092\u69cb\u7bc9\u3059\u308b\u305f\u3081\u306e\u30c4\u30fc\u30eb\u3082\u3001\u6839\u672c\u304b\u3089\u7c21\u7d20\u5316\u3055\u308c ...\n# \u30107\u2020Alexa skills developer Hugo.fm describes how bundling skills unlocks growth\u2020developer.amazon.com\u3011\n\u201cAlexa has opened up a world of opportunity for developers, says Catchpole. \u201cI\u2019m really energized by the new developer tools announced in September to build LLM-powered experiences with Alexa, and I appreciate that Amazon is constantly improving Alexa and making it even easier to build the next generation of computing experiences.\u201d\n# \u30108\u2020Best of 2023: Videos showcasing developers shaping the future of voice ...\u2020developer.amazon.com\u3011\nAlexa gives you the power to shape the next generation of customer experiences and connect with users around the world\u2014and this year, we caught up with some of the creators behind groundbreaking solutions that are ushering in a new era of voice technology. From hobbyists-turned-entrepreneurs to globally recognized brands, get a first-hand view of the latest in ambient intelligence with some ...\n# \u30109\u2020Expand Your Utterances - developer.amazon.com\u2020developer.amazon.com\u3011\nYou can improve your existing custom skill\u2019s customer experience right now using a text-based large language model (LLM) of your choice. Use an LLM to improve accuracy by generating a robust set of utterances to add to your intents. In this article: Introduction &amp; prerequisites ; Generate a large set of utterances\n# \u301010\u2020Alexa AI - developer.amazon.com\u2020developer.amazon.com\u3011\nHow you can leverage generative AI. Generative AI will drive the next step function change in ambient intelligence, enabling more personalized and more natural Alexa experiences. With our new tools, you can easily build new conversational experiences or more intuitive smart home controls, without having to write complex code or train specific ...\n# \u301011\u2020Seamless EV charger navigation Experience now available with Alexa\u2020developer.amazon.com\u3011\nCustomers can search for EV charging stations by simply saying, \u201cAlexa, find charging stations near me.\u201d Customers can also ask for more specific parameters such as, \u201cAlexa, find me a charger near Golden Gate bridge,\u201d \u201cAlexa, find me a DC fast charger,\u201d or \u201cAlexa, get me directions to a nearby EVgo charger.\u201d\n# \u301012\u2020Immersive experiences await: APL 2023.3 introduces new animation and ...\u2020developer.amazon.com\u3011\nWith APL 2023.3, we centralized the way parameters can be passed to your Alexa Vector Graphic (AVG) files. This improves readability of AVG properties which in turn helps with faster discovery and debugging of AVG properties used in your APL document.\n# \u301013\u2020Introducing \u2018Custom Triggers for Routines\u2019 for Alexa developers\u2020developer.amazon.com\u3011\nCustom triggers enable developers to unlock an additional layer of value through their services. Once onboarded, the custom triggers appear under the \u201cWhen this happens\u201d section in the Routines hub in the Alexa Mobile app. \u2018Custom Triggers for Routines\u2019 can be used by both non-device- (e.g. Alexa skills) and device-based Alexa ...\n# \u301014\u2020Amazon announces the general availability of the Alexa Routines Kit for ...\u2020developer.amazon.com\u3011\nHelen Hasbun Oct 31, 2023. Share: Alexa Skills. Today, Amazon announced the general availability of the Alexa Routines Kit. The Alexa Routines Kit enables you to configure Routines, and offer them as pre-built Routines to customers during their skill interactions. Customers can use Routines to automatically execute a sequence of actions based ...\n# \u301015\u2020Best of 2023: How leading developers unlocked growth with Alexa\u2020developer.amazon.com\u3011\nDrew Cosgrove. As a computer programmer with decades of experience, Drew Cosgrove immediately saw something unique with Alexa upon its 2016 release in the UK. Traditionally, it took several developers to build a single game\u2014but through Alexa skills, he found it was possible to craft games by himself. Cosgrove soon built engaging quiz games ...\n# \u301016\u2020Video: Alexa Smart Properties is helping residents at Fellowship Square ...\u2020developer.amazon.com\u3011\nIn this video, discover how Alexa Smart

Properties is enabling the residents of Fellowship Square-Mesa to live independent lives thsat are filled with joy and dignity.\n# \u301017\u2020Stack Overflow: A new way for developers to get support for Alexa\u2020developer.amazon.com\u3011\nAlexa Skills. Today, Amazon announced new ways for developers building with Alexa to receive technical support on Stack Overflow. Starting immediately, you will be able to get answers to questions on the AWS Collective page on Stack Overflow. You can use Alexa-related tags to get support from Alexa\u2019s developer advocacy group and the broader ...\n# \u301018\u2020Understand State and Change Reporting | Alexa Skills Kit\u2020developer.amazon.com\u3011\nWhen you create an Alexa skill that interacts with a smart home device, you include support for state and change reporting. Alexa notifies users about the state of their devices by voice response, in the Alexa app, and on Alexa-enabled devices with a screen. For example, in the Alexa app users can see a list of their smart plugs and whether ...\n# \u301019\u2020How Alexa Widgets help the developer of Big Sky double the skill\u2019s ...\u2020developer.amazon.com\u3011\nArkonovich uses Alexa Widgets to improve the customer experience, providing a way for Big Sky customers to receive detailed weather information quickly and visually. The Widget provides standard weather information, such as temperature highs and lows, alongside a summary in natural language. Customers tap the Widget to access a full weather report.\nVisible: 0% - 100%", "summary": null, "assets": [], "tether_id": null}, "status": "finished_successfully", "end_turn": null, "weight": 0.0, "metadata": {"_cite_metadata": {"citation_format": {"name": "tether_og"}, "metadata_list": [{"type": "webpage", "title": "Amazon announces LLM-powered tools for developers at the Devices and ...", "url": "https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/09/alexa-llm-fall-devices-services-sep-2023", "text": "\nTo integrate APIs with Alexa\u2019s LLM, you will provide: 1) your API endpoints and definitions 2) natural language annotations and descriptions of your APIs and business logic and 3) CX examples for when to call the APIs. This new integration model allows you to build experiences that enable customers to participate in multi-turn, cross-skill ...\n", "pub_date": "2024-10-11T01:58:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Alexa AI | Build LLM-powered Alexa experiences - developer.amazon.com", "url": "https://developer.amazon.com/en-US/alexa/alexa-ai", "text": "\nCreate more natural, conversational and intuitive Alexa experiences. Developers will be able to integrate their APIs or content into Alexa\u2019s LLM and create experiences that allow customers to access real-time data, enjoy immersive generative AI-enhanced games, perform tasks like booking restaurant reservations, getting a succinct summary of a ...\n", "pub_date": "2024-10-13T00:21:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Alexa Smart Home Insider - developer.amazon.com", "url": "https://developer.amazon.com/en-US/alexa/smarthomeinsider", "text": "\nBuild with Amazon\u2019s newest devices and services ... September 20, 2023. Sep 20, 2023. New developer tools to build LLM-powered experiences with Alexa. ... Our global network of Alexa Solution Providers works together to provide strategy and development support for Alexa devices and skills. Explore more. Find the latest resources, guides, code ...\n", "pub_date": "2024-10-14T08:33:00.0000000Z", "extra": null}, {"type": "webpage", "title": "3 questions with Amazon leader Jessica Brase on how Amazon Enterprise ...", "url": "https://developer.amazon.com/en-US/blogs/alexa/device-makers/2023/10/alexa-enterprise-services-october-2023", "text": "\nJessica Brase is the business leader of Alexa Enterprise Services (AES), an organization within Amazon that provides 24/7 technical expertise to Alexa\u2019s enterprise partners around the

world. Brase joined Amazon in January 2022 and after working in a role focused on
financial operations, she decided to pursue a role leading the newly formed ...\n",
"pub_date": "2024-10-08T02:13:00.0000000Z", "extra": null}, {"type": "webpage",
"title": "Amazon showcases new customer experiences powered by generative AI at ...",
"url": "https://developer.amazon.com/en-US/blogs/alexa/device-makers/2024/01/alexa-
characteraiandsplash-ces-2024", "text": "\nThese experiences leverage our new LLM-
powered tools announced at our Fall Devices and Services launch event. With these
tools, developers will be able to create experiences that allow customers to access
real-time data, enjoy immersive generative AI-enhanced games, perform tasks like
booking restaurant reservations, getting a succinct summary of a trending news story,
and more.\n", "pub_date": "2024-10-12T16:49:00.0000000Z", "extra": null}, {"type":
"webpage", "title": "Amazon Developers", "url": "https://www.developer.amazon.com/en",
"text": "\nSeptember 20, 2023. Sep 20, 2023. New developer tools to build LLM-powered
experiences with Alexa. Smart Home Skills. Alexa Skills. Smart Home API. July 18, 2023.
Jul 18, 2023. ... Registration is easy, free, and the first step to growing your
business with Amazon devices and services. Create account. Back to top. Legal.
Agreements and Terms;\n", "pub_date": "2024-10-11T17:07:00.0000000Z", "extra": null},
{"type": "webpage", "title":
"LLM\u3092\u6d3b\u7528\u3057\u305fAlexa\u30a8\u30af\u30b9\u30da\u30ea\u30a8\u30f3\u30b
9\u3092\u4f5c\u6210\u3059\u308b\u305f\u3081\u306e\u65b0\u3057\u3044\u958b\u767a\u8005\
u30c4\u30fc\u30eb", "url": "https://developer.amazon.com/ja-JP/blogs/alexa/alexa-
skills-kit/2023/09/alexa-llm-fall-devices-services-sep-2023", "text":
"\n\u30b3\u30f3\u30c6\u30f3\u30c4\u3084API\u3092Alexa\u306eLLM\u3068\u7d71\u5408\u3059
\u308b\u3053\u3068\u3067\u3001Alexa\u642d\u8f09\u30c7\u30d0\u30a4\u30b9\u5411\u3051\u3
06e\u4f1a\u8a71\u578b\u30a8\u30af\u30b9\u30da\u30ea\u30a8\u30f3\u30b9\u3092\u4f5c\u621
0\u3067\u304d\u307e\u3059\u3002.
\u5225\u306e\u65b9\u6cd5\u3068\u3057\u3066\u3001\u4efb\u610f\u306eLLM\u3068\u7d71\u540
8\u3059\u308b\u3053\u3068\u3082\u3067\u304d\u307e\u3059\u3002.
\u307e\u305f\u3001\u76f4\u611f\u7684\u306a\u30b9\u30de\u30fc\u30c8\u30db\u30fc\u30e0\u
5236\u5fa1\u3092\u69cb\u7bc9\u3059\u308b\u305f\u3081\u306e\u30c4\u30fc\u30eb\u3082\u30
01\u6839\u672c\u304b\u3089\u7c21\u7d20\u5316\u3055\u308c ...\n", "pub_date": "2024-10-
03T13:31:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Alexa skills
developer Hugo.fm describes how bundling skills unlocks growth", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/10/hugo-fm-
alexa-october-2023", "text": "\n\u201cAlexa has opened up a world of opportunity for
developers, says Catchpole. \u201cI\u2019m really energized by the new developer tools
announced in September to build LLM-powered experiences with Alexa, and I appreciate
that Amazon is constantly improving Alexa and making it even easier to build the next
generation of computing experiences.\u201d\n", "pub_date": "2024-08-
07T11:37:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Best of 2023:
Videos showcasing developers shaping the future of voice ...", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/12/best-of-2023-
alexa-videos-2023", "text": "\nAlexa gives you the power to shape the next generation
of customer experiences and connect with users around the world\u2014and this year, we
caught up with some of the creators behind groundbreaking solutions that are ushering
in a new era of voice technology. From hobbyists-turned-entrepreneurs to globally
recognized brands, get a first-hand view of the latest in ambient intelligence with
some ...\n", "pub_date": "2024-10-03T00:30:00.0000000Z", "extra": null}, {"type":
"webpage", "title": "Expand Your Utterances - developer.amazon.com", "url":

"https://developer.amazon.com/en-US/alexa/alexa-haus/LLM-utterances", "text": "\nYou
can improve your existing custom skill\u2019s customer experience right now using a
text-based large language model (LLM) of your choice. Use an LLM to improve accuracy
by generating a robust set of utterances to add to your intents. In this article:
Introduction &amp; prerequisites ; Generate a large set of utterances\n", "pub_date":
"2024-10-11T12:52:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Alexa AI
- developer.amazon.com", "url": "https://developer.amazon.com/en-US/alexa/alexa-
ailabcdefghijk", "text": "\nHow you can leverage generative AI. Generative AI will
drive the next step function change in ambient intelligence, enabling more
personalized and more natural Alexa experiences. With our new tools, you can easily
build new conversational experiences or more intuitive smart home controls, without
having to write complex code or train specific ...\n", "pub_date": "2024-10-
07T12:24:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Seamless EV
charger navigation Experience now available with Alexa", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/09/ev-charger-
navigation-alexa-september-2023", "text": "\nCustomers can search for EV charging
stations by simply saying, \u201cAlexa, find charging stations near me.\u201d
Customers can also ask for more specific parameters such as, \u201cAlexa, find me a
charger near Golden Gate bridge,\u201d \u201cAlexa, find me a DC fast charger,\u201d
or \u201cAlexa, get me directions to a nearby EVgo charger.\u201d\n", "pub_date":
"2024-09-28T20:39:00.0000000Z", "extra": null}, {"type": "webpage", "title":
"Immersive experiences await: APL 2023.3 introduces new animation and ...", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/12/apl2023-
alexa-december-2023", "text": "\nWith APL 2023.3, we centralized the way parameters
can be passed to your Alexa Vector Graphic (AVG) files. This improves readability of
AVG properties which in turn helps with faster discovery and debugging of AVG
properties used in your APL document.\n", "pub_date": "2024-10-06T19:39:00.0000000Z",
"extra": null}, {"type": "webpage", "title": "Introducing \u2018Custom Triggers for
Routines\u2019 for Alexa developers", "url": "https://developer.amazon.com/en-
US/blogs/alexa/alexa-skills-kit/2023/09/custom-triggers-routines-alexa-september-2023",
"text": "\nCustom triggers enable developers to unlock an additional layer of value
through their services. Once onboarded, the custom triggers appear under the
\u201cWhen this happens\u201d section in the Routines hub in the Alexa Mobile app.
\u2018Custom Triggers for Routines\u2019 can be used by both non-device- (e.g. Alexa
skills) and device-based Alexa ...\n", "pub_date": "2024-10-10T06:11:00.0000000Z",
"extra": null}, {"type": "webpage", "title": "Amazon announces the general
availability of the Alexa Routines Kit for ...", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/10/routines-kit-
ga-alexa-october-2023", "text": "\nHelen Hasbun Oct 31, 2023. Share: Alexa Skills.
Today, Amazon announced the general availability of the Alexa Routines Kit. The Alexa
Routines Kit enables you to configure Routines, and offer them as pre-built Routines
to customers during their skill interactions. Customers can use Routines to
automatically execute a sequence of actions based ...\n", "pub_date": "2024-09-
26T09:31:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Best of 2023: How
leading developers unlocked growth with Alexa", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/12/best-of-2023-
alexa-grow-december-2023", "text": "\nDrew Cosgrove. As a computer programmer with
decades of experience, Drew Cosgrove immediately saw something unique with Alexa upon
its 2016 release in the UK. Traditionally, it took several developers to build a

single game\u2014but through Alexa skills, he found it was possible to craft games by
himself. Cosgrove soon built engaging quiz games ...\n", "pub_date": "2024-10-
02T17:17:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Video: Alexa
Smart Properties is helping residents at Fellowship Square ...", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/11/fellowship-
square-alexa-smart-properties-october-2023", "text": "\nIn this video, discover how
Alexa Smart Properties is enabling the residents of Fellowship Square-Mesa to live
independent lives thsat are filled with joy and dignity.\n", "pub_date": "2024-09-
29T22:49:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Stack Overflow: A
new way for developers to get support for Alexa", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/09/stack-
overflow-alexa-october-2023", "text": "\nAlexa Skills. Today, Amazon announced new
ways for developers building with Alexa to receive technical support on Stack Overflow.
Starting immediately, you will be able to get answers to questions on the AWS
Collective page on Stack Overflow. You can use Alexa-related tags to get support from
Alexa\u2019s developer advocacy group and the broader ...\n", "pub_date": "2024-09-
28T12:23:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Understand State
and Change Reporting | Alexa Skills Kit", "url": "https://developer.amazon.com/en-
US/docs/alexa/smarthome/state-reporting-for-a-smart-home-skill.html", "text": "\nWhen
you create an Alexa skill that interacts with a smart home device, you include support
for state and change reporting. Alexa notifies users about the state of their devices
by voice response, in the Alexa app, and on Alexa-enabled devices with a screen. For
example, in the Alexa app users can see a list of their smart plugs and whether ...\n",
"pub_date": "2024-10-09T17:44:00.0000000Z", "extra": null}, {"type": "webpage",
"title": "How Alexa Widgets help the developer of Big Sky double the skill\u2019s ...",
"url": "https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/10/big-
sky-alexa-october-2023", "text": "\nArkonovich uses Alexa Widgets to improve the
customer experience, providing a way for Big Sky customers to receive detailed weather
information quickly and visually. The Widget provides standard weather information,
such as temperature highs and lows, alongside a summary in natural language. Customers
tap the Widget to access a full weather report.\nVisible: 0% - 100%", "pub_date":
"2024-10-11T12:49:00.0000000Z", "extra": null}, {"type": "webpage", "title": "Search
results for query: &#x27;Alexa LLM fall devices services sep 2023
site:developer.amazon.com&#x27;", "url": "", "text": "# \u30100\u2020Amazon announces
LLM-powered tools for developers at the Devices
and ...\u2020developer.amazon.com\u3011\nTo integrate APIs with Alexa\u2019s LLM, you
will provide: 1) your API endpoints and definitions 2) natural language annotations
and descriptions of your APIs and business logic and 3) CX examples for when to call
the APIs. This new integration model allows you to build experiences that enable
customers to participate in multi-turn, cross-skill ...\n# \u30101\u2020Alexa AI |
Build LLM-powered Alexa experiences -
developer.amazon.com\u2020developer.amazon.com\u3011\nCreate more natural,
conversational and intuitive Alexa experiences. Developers will be able to integrate
their APIs or content into Alexa\u2019s LLM and create experiences that allow
customers to access real-time data, enjoy immersive generative AI-enhanced games,
perform tasks like booking restaurant reservations, getting a succinct summary of
a ...\n# \u30102\u2020Alexa Smart Home Insider -
developer.amazon.com\u2020developer.amazon.com\u3011\nBuild with Amazon\u2019s newest
devices and services ... September 20, 2023. Sep 20, 2023. New developer tools to

build LLM-powered experiences with Alexa. ... Our global network of Alexa Solution Providers works together to provide strategy and development support for Alexa devices and skills. Explore more. Find the latest resources, guides, code ...\n# \u30103\u20203 questions with Amazon leader Jessica Brase on how Amazon Enterprise ...\u2020developer.amazon.com\u3011\nJessica Brase is the business leader of Alexa Enterprise Services (AES), an organization within Amazon that provides 24/7 technical expertise to Alexa\u2019s enterprise partners around the world. Brase joined Amazon in January 2022 and after working in a role focused on financial operations, she decided to pursue a role leading the newly formed ...\n# \u30104\u2020Amazon showcases new customer experiences powered by generative AI at ...\u2020developer.amazon.com\u3011\nThese experiences leverage our new LLM-powered tools announced at our Fall Devices and Services launch event. With these tools, developers will be able to create experiences that allow customers to access real-time data, enjoy immersive generative AI-enhanced games, perform tasks like booking restaurant reservations, getting a succinct summary of a trending news story, and more.\n# \u30105\u2020Amazon Developers\u2020www.developer.amazon.com\u3011\nSeptember 20, 2023. Sep 20, 2023. New developer tools to build LLM-powered experiences with Alexa. Smart Home Skills. Alexa Skills. Smart Home API. July 18, 2023. Jul 18, 2023. ... Registration is easy, free, and the first step to growing your business with Amazon devices and services. Create account. Back to top. Legal. Agreements and Terms;\n# \u30106\u2020LLM\u3092\u6d3b\u7528\u3057\u305fAlexa\u30a8\u30af\u30b9\u30da\u30ea\u30a8\u30f3\u30b9\u3092\u4f5c\u6210\u3059\u308b\u305f\u3081\u306e\u65b0\u3057\u3044\u958b\u767a\u8005\u30c4\u30fc\u30eb\u2020developer.amazon.com\u3011\n\u30b3\u30f3\u30c6\u30f3\u30c4\u3084API\u3092Alexa\u306eLLM\u3068\u7d71\u5408\u3059\u308b\u3053\u3068\u3067\u3001Alexa\u642d\u8f09\u30c7\u30d0\u30a4\u30b9\u5411\u3051\u306e\u4f1a\u8a71\u578b\u30a8\u30af\u30b9\u30da\u30ea\u30a8\u30f3\u30b9\u3092\u4f5c\u6210\u3067\u304d\u307e\u3059\u3002. \u5225\u306e\u65b9\u6cd5\u3068\u3057\u3066\u3001\u4efb\u610f\u306eLLM\u3068\u7d71\u5408\u3059\u308b\u3053\u3068\u3082\u3067\u304d\u307e\u3059\u3002. \u307e\u305f\u3001\u76f4\u611f\u7684\u306a\u30b9\u30de\u30fc\u30c8\u30db\u30fc\u30e0\u5236\u5fa1\u3092\u69cb\u7bc9\u3059\u308b\u305f\u3081\u306e\u30c4\u30fc\u30eb\u3082\u3001\u6839\u672c\u304b\u3089\u7c21\u7d20\u5316\u3055\u308c ...\n# \u30107\u2020Alexa skills developer Hugo.fm describes how bundling skills unlocks growth\u2020developer.amazon.com\u3011\n\u201cAlexa has opened up a world of opportunity for developers, says Catchpole. \u201cI\u2019m really energized by the new developer tools announced in September to build LLM-powered experiences with Alexa, and I appreciate that Amazon is constantly improving Alexa and making it even easier to build the next generation of computing experiences.\u201d\n# \u30108\u2020Best of 2023: Videos showcasing developers shaping the future of voice ...\u2020developer.amazon.com\u3011\nAlexa gives you the power to shape the next generation of customer experiences and connect with users around the world\u2014and this year, we caught up with some of the creators behind groundbreaking solutions that are ushering in a new era of voice technology. From hobbyists-turned-entrepreneurs to globally recognized brands, get a first-hand view of the latest in ambient intelligence with some ...\n# \u30109\u2020Expand Your Utterances - developer.amazon.com\u2020developer.amazon.com\u3011\nYou can improve your existing custom skill\u2019s customer experience right now using a text-based large language model (LLM) of your choice. Use an LLM to improve accuracy by generating a robust set of utterances to add to your intents. In this article: Introduction &amp; prerequisites ; Generate a large set of utterances\n# \u301010\u2020Alexa AI - developer.amazon.com\u2020developer.amazon.com\u3011\nHow you can leverage generative AI. Generative AI will drive the next step function change in ambient intelligence, enabling more personalized and more natural Alexa experiences. With our new tools, you can easily build new conversational experiences or more intuitive smart home controls, without having to write complex code or train specific ...\n# \u301011\u2020Seamless EV charger navigation Experience now available with Alexa\u2020developer.amazon.com\u3011\nCustomers can search for EV charging stations by simply saying, \u201cAlexa, find charging stations near me.\u201d Customers can also ask for more specific parameters such as, \u201cAlexa, find me a charger near Golden Gate bridge,\u201d \u201cAlexa, find me a DC fast charger,\u201d or \u201cAlexa, get me directions to a nearby EVgo charger.\u201d\n# \u301012\u2020Immersive experiences await: APL 2023.3 introduces new animation and ...\u2020developer.amazon.com\u3011\nWith APL 2023.3, we centralized the way parameters can be passed to your Alexa Vector Graphic (AVG) files. This improves readability of AVG properties which in turn helps with faster discovery and debugging of AVG properties used in your APL document.\n# \u301013\u2020Introducing \u2018Custom Triggers for Routines\u2019 for Alexa developers\u2020developer.amazon.com\u3011\nCustom triggers enable developers to unlock an additional layer of value through their services. Once onboarded, the custom triggers appear under the \u201cWhen this happens\u201d section in the Routines hub in the Alexa Mobile app. \u2018Custom Triggers for Routines\u2019 can be used by both non-device- (e.g. Alexa skills) and device-based Alexa ...\n# \u301014\u2020Amazon announces the general availability of the Alexa Routines Kit for ...\u2020developer.amazon.com\u3011\nHelen Hasbun Oct 31, 2023. Share: Alexa Skills. Today, Amazon announced the general availability of the Alexa Routines Kit. The Alexa Routines Kit enables you to configure Routines, and offer them as pre-built Routines to customers during their skill interactions. Customers can use Routines to automatically execute a sequence of actions based ...\n# \u301015\u2020Best of 2023: How leading developers unlocked growth with Alexa\u2020developer.amazon.com\u3011\nDrew Cosgrove. As a computer programmer with decades of experience, Drew Cosgrove immediately saw something unique with Alexa upon its 2016 release in the UK. Traditionally, it took several developers to build a single game\u2014but through Alexa skills, he found it was possible to craft games by himself. Cosgrove soon built engaging quiz games ...\n# \u301016\u2020Video: Alexa Smart Properties is helping residents at Fellowship Square ...\u2020developer.amazon.com\u3011\nIn this video, discover how Alexa Smart Properties is enabling the residents of Fellowship Square-Mesa to live independent lives thsat are filled with joy and dignity.\n# \u301017\u2020Stack Overflow: A new way for developers to get support for Alexa\u2020developer.amazon.com\u3011\nAlexa Skills. Today, Amazon announced new ways for developers building with Alexa to receive technical support on Stack Overflow. Starting immediately, you will be able to get answers to questions on the AWS Collective page on Stack Overflow. You can use Alexa-related tags to get support from Alexa\u2019s developer advocacy group and the broader ...\n# \u301018\u2020Understand State and Change Reporting | Alexa Skills Kit\u2020developer.amazon.com\u3011\nWhen you create an Alexa skill that interacts with a smart home device, you include support for state and change reporting. Alexa notifies users about the state of their devices by voice response, in the Alexa app,

and on Alexa-enabled devices with a screen. For example, in the Alexa app users can
see a list of their smart plugs and whether ...\n# \u301019\u2020How Alexa Widgets
help the developer of Big Sky double the
skill\u2019s ...\u2020developer.amazon.com\u3011\nArkonovich uses Alexa Widgets to
improve the customer experience, providing a way for Big Sky customers to receive
detailed weather information quickly and visually. The Widget provides standard
weather information, such as temperature highs and lows, alongside a summary in
natural language. Customers tap the Widget to access a full weather report.\nVisible:
0% - 100%", "pub_date": null, "extra": null}], "original_query": null}, "command":
"search", "args": ["Alexa LLM fall devices services sep 2023
site:developer.amazon.com"], "status": "finished", "is_complete": true, "message_type":
null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "72728eb8-
4f49-4101-bc98-2d05a9d9b3c2", "request_id": "8d2aeb5efa67524c-MXP", "timestamp_":
"absolute"}, "recipient": "all", "channel": null}, "parent": "72728eb8-4f49-4101-bc98-
2d05a9d9b3c2", "children": ["5127cb45-d788-4e58-be9e-880bab162041"]}, "5127cb45-d788-
4e58-be9e-880bab162041": {"id": "5127cb45-d788-4e58-be9e-880bab162041", "message":
{"id": "5127cb45-d788-4e58-be9e-880bab162041", "author": {"role": "assistant", "name":
null, "metadata": {}}, "create_time": 1728944074.278734, "update_time": null,
"content": {"content_type": "code", "language": "unknown", "response_format_name":
null, "text": "mclick([0,1,4])"}, "status": "finished_successfully", "end_turn": false,
"weight": 1.0, "metadata": {"citations": [], "content_references": [], "gizmo_id":
null, "message_type": null, "model_slug": "gpt-4o", "default_model_slug": "gpt-4o",
"parent_id": "9a904751-ba2e-4bf0-99f9-1186d377282f", "request_id": "8d2aeb5efa67524c-
MXP", "timestamp_": "absolute", "finish_details": {"type": "stop", "stop_tokens":
[200007]}, "is_complete": true}, "recipient": "browser", "channel": null}, "parent":
"9a904751-ba2e-4bf0-99f9-1186d377282f", "children": ["e7b87423-929a-4ed4-ace3-
4feba57d85a1"]}, "e7b87423-929a-4ed4-ace3-4feba57d85a1": {"id": "e7b87423-929a-4ed4-
ace3-4feba57d85a1", "message": {"id": "e7b87423-929a-4ed4-ace3-4feba57d85a1", "author":
{"role": "tool", "name": "browser", "metadata": {}}, "create_time": 1728944074.278829,
"update_time": null, "content": {"content_type": "tether_quote", "url":
"https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/09/alexa-llm-
fall-devices-services-sep-2023", "domain": "developer.amazon.com", "text": "\n[Image 5:
Alexa Orchestration] \n\nConnect your APIs with Alexa\u2019s LLM\n\nTo integrate APIs
with Alexa\u2019s LLM, you will provide: 1) your API endpoints and definitions 2)
natural language annotations and descriptions of your APIs and business logic and 3)
CX examples for when to call the APIs. This new integration model allows you to build
experiences that enable customers\u00a0to participate in multi-turn, cross-skill
conversations. For example, a customer planning a trip might ask about the best time
of year to visit, places to see, and hotel reservations in the same conversation. To
fulfill this task, relevant APIs will be called based on the API definitions and
descriptions provided by you. As Alexa orchestrates and chains together intermediate
natural language reasoning steps, it decomposes complex tasks into smaller, more
manageable steps, and constructs the response to the customer. Alexa is able to
complete these complex tasks in real time in line with customer expectations for voice
interactions.\n\nWe have also enhanced our Skill manifest with an increased number of
configuration fields to build LLM-based
experiences.\u00a0\n\nmodelFacingName\u00a0and\u00a0modelFacingDescription\u00a0fields
are used by the model to understand the functionality the Alexa Skill
provides.\u00a0actionProvider\u00a0enables a new type of integration for integrating

APIs with Alexa.\u00a0\n\n\u2398Copy code Copied to clipboard\n\n   {\n
&quot;manifest&quot;: {\n   &quot;modelFacingName&quot;: &quot;Restaurant
Reservations&quot;,\n   &quot;modelFacingDescription&quot;: &quot;Use this to
book restaurant reservations&quot;,\n   &quot;publishingInformation&quot;:
{\n   // same as before\n   },\n   &quot;apis&quot;: {\n
&quot;actionProvider&quot;: {\n   &quot;endpoint&quot;: { \n
&quot;uri&quot;: &quot;arn:aws:lambda:us-east-1:452493640596:function:restaurantReservationsHandler&quot;\n   }\n
                }\n   }\n   }\n   }\n\nTo make it simpler for developers
to author and maintain API specifications and logic, we will support API declarations
using our SDKs for the most common interface languages, starting with
TypeScript.\u00a0\n\n\u2398Copy code Copied to clipboard\n\n   @Service\n   export
class RestaurantReservationService {\n   @Description(&quot;Use this method to find
restaurants by the zipCode&quot;)\n   public findRestaurants(zipCode: string) :
Promise&lt;Restaurant&gt; {\n   // restaurant search logic\n   }\n\n
@Description(&quot;Use this to reserve a restaurant, confirm with the user before
reserving&quot;)\n   public reserveRestaurant(restaurant: Restaurant, partySize:
number, date: string): Promise&lt;string&gt; {\n   // reservation logic
\n   }\n   }\n\n   @Description(&quot;Restaurant schema&quot;)\n   export
interface Restaurant {\n   restaurantId: string;\n   name: string;\n
address: string;\n   address2: string;\n   city: string;\n   state:
string;\n   country: string;\n   }\n\nConnect your content with Alexa\u2019s
LLM\u00a0\n\nYou will also be able to build content-driven, ambient experiences and
deliver them to over half a billion Alexa-enabled devices. You can use unstructured
data like product manuals or websites, or structured data from existing databases to
enable customers to find the perfect show on Prime Video, get summaries of earnings
reports, access recipes from an expert chef and more. These experiences will be
powered by techniques such as retrieval augmented generation without needing to train
core LLM models on your data.\u00a0 \n\nTo integrate content, create a content spec
that includes 1) a short natural language description of the content (e.g., FAQs on
amenities and guest services) 2) instructions on the desired use-cases to support and
3) the unstructured content or structured data, and ingest them using our new tools.
These tools include managed instances of vector databases, which eliminate the need
for you to provision and maintain your own services. When a customer asks for
information related to your content, Alexa searches the content store, and forms a
response using the most relevant information.\n\n\u2398Copy code Copied to
clipboard\n\n   {\n   &quot;manifest&quot;: {\n
&quot;modelFacingName&quot;: &quot;Food Recipes&quot;,\n
&quot;modelFacingDescription&quot;: &quot;Use this to find food recipes.&quot;,\n
&quot;publishingInformation&quot;: {\n   // same as
before\n   },\n   &quot;apis&quot;: {\n
&quot;contentProvider&quot;: {\n   &quot;sources&quot;: [\n
{\n   &quot;location&quot;:
&quot;skillpackage://content/food-recipes.pdf&quot;\n   }\n   ]\n   }
\n   }\n   }\n   }\n\nConnect your own LLM with Alexa devices and
experiences\n\nIn addition, we understand that you may have already purpose-built your
own LLMs or leveraged LLMs offered through services like Amazon Bedrock. When you want
to bring your LLM-based experiences to life through expressive voice or ambient

visuals, and distribute them to Alexa\u2019s devices, you will have the ability to integrate your Alexa Skill with one or more LLMs of your choice with a new tool. To integrate third-party LLMs, all you\u2019ll have to do is integrate with automatic speech recognition (ASR) or utterance text using a simple SDK that supports the entire range of multimodal experiences enabled by Alexa.\u00a0 \n\nWe\u2019re excited about the experiences being created by developers, and we\u2019ll have more to share soon from companies like Character.AI, Splash, BMW, Volley, Sleep Jar, Voice Apps, and OpenTable who are enabling customers to do things like talk to fictional characters, create their own songs, play\u00a0single and multi-agent driven games on Alexa-enabled devices, learn about the features of their vehicle and so much more.\u00a0 \n\nLeverage differentiated offerings for smart home developers \n\nToday, customers have connected over 400 million smart home devices to Alexa. Our partners have been critical in helping build smart home experiences\u00a0with devices from all brands working seamlessly together to make customers\u2019 lives easier. We will be leveraging generative AI to allow customers to control their smart home devices for a variety of actions through flexible and natural conversations, which Alexa\u2019s LLM will map to the unique and differentiated features of their products. To cover the broadest set of use cases, we will be providing developers with a path for both common integrations such as on/off or start/stop actions, and more complex ones such as actions related to dynamic lighting or custom cleaning operations for a vacuum cleaner. \n\nSmart home Generic Controller Interfaces such as our existing Alexa.RangeController and Alexa.ModeController remain a simple yet effective way to integrate devices with Alexa for common use cases. A new interface called Action Controller extends this further by enabling device makers to model simple actions, targets, and states to control their devices, giving customers the ability to interact naturally with their devices. This means customers using Alexa\u2019s LLM can say \u201cAlexa, the floor is dirty,\u201d and Alexa will be able to infer the intended action is \u2018vacuum\u2019, and state \u2018start.\u2019 Action Controller makes it simple for device makers to model new supported actions and targets. To take advantage of Action Controller, you need to 1) define your supported device actions such as clean, mop, or dock 2) define a list of targets such as \u2018kitchen\u2019 or \u2018living room,\u2019 3) choose the state(s) to support (choose one or more actions from start/stop/pause/resume) and 4) implement the required directive and state updates when Alexa sends the requested actions to your skills.\n\nBelow is an example of a multi-functional device modeled using the Action Controller instance. In this example, the device supports cleaning type actions (such as \u2018vacuum,\u2019) a list of target zones and device action states (start/stop/pause/resume.)\u00a0\n\n\u2398Copy code Copied to clipboard\n\n { &quot;capabilities&quot;: [{\n   &quot;type&quot;: &quot;AlexaInterface&quot;,\n   &quot;interface&quot;: &quot;Alexa.ActionController&quot;,\n   &quot;instance&quot;: &quot;Robot.Clean&quot;,\n   &quot;configuration&quot;: {\n &quot;supportedActions&quot;: [{ //An identifier device maker defined for the actions reported.\n   &quot;id&quot;: &quot;Vacuum-device-maker-defined-identifier&quot;,\n   &quot;friendlyNames&quot;: [{\n &quot;@type&quot;: &quot;asset&quot;,\n   &quot;value&quot;: {\n &quot;assetId&quot;: &quot;Alexa.Actions.Vacuum&quot;\n   }}]}],\n &quot;supportedStates&quot;: [\n   &quot;START&quot;,\n &quot;STOP&quot;,\n   &quot;PAUSE&quot;,\n

&quot;RESUME&quot;\n   ],\n   &quot;target&quot;: {\n &quot;supportedTargets&quot;: [{ //A list of identifiers device maker defined for the targets reported\n   &quot;id&quot;: &quot;space-id-1&quot;,\n &quot;friendlyNames&quot;: [\n   {\n &quot;@type&quot;: &quot;asset&quot;,\n   &quot;value&quot;: {\n &quot;assetId&quot;: &quot;Alexa.Space.Kitchen&quot;\n   }}]}]}}}]}\n\nBelow is the example of the directive a device maker will receive for the start action:\n\n\u2398Copy code Copied to clipboard\n\n   {\n   &quot;directive&quot;: {\n   &quot;header&quot;: {\n   &quot;namespace&quot;: &quot;Alexa.ActionController&quot;,\n   &quot;instance&quot;: &quot;Robot.Clean&quot;,\n   &quot;name&quot;: &quot;PerformAction&quot;,\n &quot;messageId&quot;: &quot;message id&quot;,\n   }\n   &quot;payload&quot;: {\n   &quot;actionId&quot;: &quot;Vacuum-device-maker-defined-identifier&quot;,\n &quot;targetIds&quot;: [&quot;&lt;the target identifier&gt;&quot;],\n &quot;state&quot;: &quot;START&quot;\n   }}}\n\nFor device control beyond a simple action, more complex device states, or parameters, we are introducing Dynamic Controller, a new tool allowing developers to define a custom interface. Dynamic Controller enables customers to use a wider range of utterances to control the unique features of their smart home devices in a more conversational way (e.g. a customer can request a lighting solution provider to \u201cmake the lights like a misty forest\u201d \u2013 in the past they would only have been able to select from a list of predefined options). Dynamic Controller makes it simpler for smart home dev", "title": "Amazon announces LLM-powered tools for developers at the Devices and Services Fall Launch Event", "tether_id": null}, "status": "finished_successfully", "end_turn": null, "weight": 0.0, "metadata": {"_cite_metadata": {"citation_format": {"name": "tether_og"}, "metadata_list": [{"type": "webpage", "title": "Amazon announces LLM-powered tools for developers at the Devices and Services Fall Launch Event", "url": "https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2023/09/alexa-llm-fall-devices-services-sep-2023", "text": "\n[Image 5: Alexa Orchestration] \n\nConnect your APIs with Alexa\u2019s LLM\n\nTo integrate APIs with Alexa\u2019s LLM, you will provide: 1) your API endpoints and definitions 2) natural language annotations and descriptions of your APIs and business logic and 3) CX examples for when to call the APIs. This new integration model allows you to build experiences that enable customers\u00a0to participate in multi-turn, cross-skill conversations. For example, a customer planning a trip might ask about the best time of year to visit, places to see, and hotel reservations in the same conversation. To fulfill this task, relevant APIs will be called based on the API definitions and descriptions provided by you. As Alexa orchestrates and chains together intermediate natural language reasoning steps, it decomposes complex tasks into smaller, more manageable steps, and constructs the response to the customer. Alexa is able to complete these complex tasks in real time in line with customer expectations for voice interactions.\n\nWe have also enhanced our Skill manifest with an increased number of configuration fields to build LLM-based experiences.\u00a0\n\nmodelFacingName\u00a0and\u00a0modelFacingDescription\u00a0fields are used by the model to understand the functionality the Alexa Skill provides.\u00a0actionProvider\u00a0enables a new type of integration for integrating APIs with Alexa.\u00a0\n\n\u2398Copy code Copied to clipboard\n\n   {\n &quot;manifest&quot;: {\n   &quot;modelFacingName&quot;: &quot;Restaurant Reservations&quot;,\n   &quot;modelFacingDescription&quot;: &quot;Use this to

book restaurant reservations&quot;,\n &quot;publishingInformation&quot;: {\n // same as before\n },\n &quot;apis&quot;: {\n &quot;actionProvider&quot;: {\n &quot;endpoint&quot;: { \n &quot;uri&quot;: &quot;arn:aws:lambda:us-east-1:452493640596:function:restaurantReservationsHandler&quot;\n }\n }\n }\n }\n }\n\nTo make it simpler for developers to author and maintain API specifications and logic, we will support API declarations using our SDKs for the most common interface languages, starting with TypeScript.\u00a0\n\n\u2398Copy code Copied to clipboard\n\n @Service\n export class RestaurantReservationService {\n @Description(&quot;Use this method to find restaurants by the zipCode&quot;)\n public findRestaurants(zipCode: string) : Promise&lt;Restaurant&gt; {\n // restaurant search logic\n }\n\n @Description(&quot;Use this to reserve a restaurant, confirm with the user before reserving&quot;)\n public reserveRestaurant(restaurant: Restaurant, partySize: number, date: string): Promise&lt;string&gt; {\n // reservation logic \n }\n }\n\n @Description(&quot;Restaurant schema&quot;)\n export interface Restaurant {\n restaurantId: string;\n name: string;\n address: string;\n address2: string;\n city: string;\n state: string;\n country: string;\n }\n\nConnect your content with Alexa\u2019s LLM\u00a0\n\nYou will also be able to build content-driven, ambient experiences and deliver them to over half a billion Alexa-enabled devices. You can use unstructured data like product manuals or websites, or structured data from existing databases to enable customers to find the perfect show on Prime Video, get summaries of earnings reports, access recipes from an expert chef and more. These experiences will be powered by techniques such as retrieval augmented generation without needing to train core LLM models on your data.\u00a0 \n\nTo integrate content, create a content spec that includes 1) a short natural language description of the content (e.g., FAQs on amenities and guest services) 2) instructions on the desired use-cases to support and 3) the unstructured content or structured data, and ingest them using our new tools. These tools include managed instances of vector databases, which eliminate the need for you to provision and maintain your own services. When a customer asks for information related to your content, Alexa searches the content store, and forms a response using the most relevant information.\n\n\u2398Copy code Copied to clipboard\n\n {\n &quot;manifest&quot;: {\n &quot;modelFacingName&quot;: &quot;Food Recipes&quot;,\n &quot;modelFacingDescription&quot;: &quot;Use this to find food recipes.&quot;,\n &quot;publishingInformation&quot;: {\n // same as before\n },\n &quot;apis&quot;: {\n &quot;contentProvider&quot;: {\n &quot;sources&quot;: [\n {\n &quot;location&quot;: &quot;skillpackage://content/food-recipes.pdf&quot;\n }\n]\n } \n }\n }\n }\n\nConnect your own LLM with Alexa devices and experiences\n\nIn addition, we understand that you may have already purpose-built your own LLMs or leveraged LLMs offered through services like Amazon Bedrock. When you want to bring your LLM-based experiences to life through expressive voice or ambient visuals, and distribute them to Alexa\u2019s devices, you will have the ability to integrate your Alexa Skill with one or more LLMs of your choice with a new tool. To integrate third-party LLMs, all you\u2019ll have to do is integrate with automatic speech recognition (ASR) or utterance text using a simple SDK that supports the entire range of multimodal experiences enabled by Alexa.\u00a0 \n\nWe\u2019re excited about the experiences being created by developers, and we\u2019ll have more to share soon from companies like Character.AI, Splash, BMW, Volley, Sleep Jar, Voice Apps, and OpenTable who are enabling customers to do things like talk to fictional characters, create their own songs, play\u00a0single and multi-agent driven games on Alexa-enabled devices, learn about the features of their vehicle and so much more.\u00a0 \n\nLeverage differentiated offerings for smart home developers \n\nToday, customers have connected over 400 million smart home devices to Alexa. Our partners have been critical in helping build smart home experiences\u00a0with devices from all brands working seamlessly together to make customers\u2019 lives easier. We will be leveraging generative AI to allow customers to control their smart home devices for a variety of actions through flexible and natural conversations, which Alexa\u2019s LLM will map to the unique and differentiated features of their products. To cover the broadest set of use cases, we will be providing developers with a path for both common integrations such as on/off or start/stop actions, and more complex ones such as actions related to dynamic lighting or custom cleaning operations for a vacuum cleaner. \n\nSmart home Generic Controller Interfaces such as our existing Alexa.RangeController and Alexa.ModeController remain a simple yet effective way to integrate devices with Alexa for common use cases. A new interface called Action Controller extends this further by enabling device makers to model simple actions, targets, and states to control their devices, giving customers the ability to interact naturally with their devices. This means customers using Alexa\u2019s LLM can say \u201cAlexa, the floor is dirty,\u201d and Alexa will be able to infer the intended action is \u2018vacuum\u2019, and state \u2018start.\u2019 Action Controller makes it simple for device makers to model new supported actions and targets. To take advantage of Action Controller, you need to 1) define your supported device actions such as clean, mop, or dock 2) define a list of targets such as \u2018kitchen\u2019 or \u2018living room,\u2019 3) choose the state(s) to support (choose one or more actions from start/stop/pause/resume) and 4) implement the required directive and state updates when Alexa sends the requested actions to your skills.\n\nBelow is an example of a multi-functional device modeled using the Action Controller instance. In this example, the device supports cleaning type actions (such as \u2018vacuum,\u2019) a list of target zones and device action states (start/stop/pause/resume.)\u00a0\n\n\u2398Copy code Copied to clipboard\n\n { &quot;capabilities&quot;: [{\n   &quot;type&quot;: &quot;AlexaInterface&quot;,\n   &quot;interface&quot;: &quot;Alexa.ActionController&quot;,\n   &quot;instance&quot;: &quot;Robot.Clean&quot;,\n   &quot;configuration&quot;: {\n &quot;supportedActions&quot;: [{ //An identifier device maker defined for the actions reported.\n   &quot;id&quot;: &quot;Vacuum-device-maker-defined-identifier&quot;,\n   &quot;friendlyNames&quot;: [{\n &quot;@type&quot;: &quot;asset&quot;,\n   &quot;value&quot;: {\n &quot;assetId&quot;: &quot;Alexa.Actions.Vacuum&quot;\n   }}]}],\n &quot;supportedStates&quot;: [\n   &quot;START&quot;,\n &quot;STOP&quot;,\n   &quot;PAUSE&quot;,\n &quot;RESUME&quot;\n   ],\n   &quot;target&quot;: {\n &quot;supportedTargets&quot;: [{ //A list of identifiers device maker defined for the targets reported\n   &quot;id&quot;: &quot;space-id-1&quot;,\n

```
&quot;friendlyNames&quot;: [\n                    {\n
&quot;@type&quot;: &quot;asset&quot;,\n                    &quot;value&quot;: {\n
&quot;assetId&quot;: &quot;Alexa.Space.Kitchen&quot;\n    }}]}]}}}]}\n\nBelow is the
example of the directive a device maker will receive for the start
action:\n\n\u2398Copy code Copied to clipboard\n\n    {\n      &quot;directive&quot;:
{\n      &quot;header&quot;: {\n      &quot;namespace&quot;:
&quot;Alexa.ActionController&quot;,\n      &quot;instance&quot;:
&quot;Robot.Clean&quot;,\n      &quot;name&quot;: &quot;PerformAction&quot;,\n
&quot;messageId&quot;: &quot;message id&quot;,\n      }\n      &quot;payload&quot;:
{\n      &quot;actionId&quot;: &quot;Vacuum-device-maker-defined-identifier&quot;,\n
&quot;targetIds&quot;: [&quot;&lt;the target identifier&gt;&quot;],\n
&quot;state&quot;: &quot;START&quot;\n      }}}\n\nFor device control beyond a simple
action, more complex device states, or parameters, we are introducing Dynamic
Controller, a new tool allowing developers to define a custom interface. Dynamic
Controller enables customers to use a wider range of utterances to control the unique
features of their smart home devices in a more conversational way (e.g. a customer can
request a lighting solution provider to \u201cmake the lights like a misty
forest\u201d \u2013 in the past they would only have been able to select from a list
of predefined options). Dynamic Controller makes it simpler for smart home dev",
"pub_date": null, "extra": null}], "original_query": null}, "command": "mclick",
"args": ["[0, 1, 4]"], "status": "finished", "is_complete": true, "message_type": null,
"model_slug": "gpt-4o", "default_model_slug": "gpt-4o", "parent_id": "5127cb45-d788-
4e58-be9e-880bab162041", "request_id": "8d2aeb5efa67524c-MXP", "timestamp_":
"absolute"}, "recipient": "all", "channel": null}, "parent": "5127cb45-d788-4e58-be9e-
880bab162041", "children": ["10883603-dbdf-4b9a-98c8-a3221e1510b0"]}, "10883603-dbdf-
4b9a-98c8-a3221e1510b0": {"id": "10883603-dbdf-4b9a-98c8-a3221e1510b0", "message":
{"id": "10883603-dbdf-4b9a-98c8-a3221e1510b0", "author": {"role": "tool", "name":
"browser", "metadata": {}}, "create_time": 1728944074.278905, "update_time": null,
"content": {"content_type": "tether_quote", "url": "https://developer.amazon.com/en-
US/alexa/alexa-ai", "domain": "developer.amazon.com", "text": "\nURL:
https://developer.amazon.com/en-US/alexa/alexa-ai\n\n\u30100\u2020 \u3011 \n\n  *
\u30101\u2020Sign In\u3011 \n  *   *   \n\n\u30102\u2020 Documentation \u3011
\u30103\u2020 Stack Overflow \u2020stackoverflow.com\u3011 \u30104\u2020 Contact Us
\u3011 \n  * \u30100\u2020 Alexa Developer Home \u3011 \u30105\u2020 Alexa Developer
Console \u3011 \u30106\u2020 Alexa Voice Service \u3011 \u30107\u2020 Alexa
Certification Console \u3011 \u30108\u2020 Alexa Connect Kit \u3011 \n\n  *
\u30109\u2020 Skill Builders \u3011 \n\n    * \u30109\u2020 Alexa Skills Kit \u3011 \n
* \u301010\u2020 Feature Updates \u3011 \n    * \u301011\u2020 Documentation \u3011 \n
* \u301012\u2020 Grow Your Business \u3011 \n    * \u30105\u2020 Developer Console
\u3011 \n    * \u301013\u2020 Tech Talks \u3011 \n\n  * \u301014\u2020 Device Makers
\u3011 \n\n    * Alexa Built-in Devices Develop Alexa built-in devices with Alexa
Voice Service\n\n      * \u301015\u2020 Overview \u3011 \n\n    * Connected Devices
Connect your smart devices to Alexa\n\n      * \u301016\u2020 Overview \u3011 \n
* \u301017\u2020 Development Resources \u3011 \n      * \u301018\u2020 Business
Resources \u3011 \n      * \u301019\u2020 What&#x27;s New \u3011 \n      *
\u301020\u2020 Consoles \u3011 \n\n    * \u301013\u2020 Tech Talks \u3011 \n\n  *
\u301021\u2020 Solution Providers \u3011 \n\n    * \u301022\u2020 Skill Agencies
\u3011 \n    * \u301023\u2020 Alexa Connect Kit \u3011 \n    * \u301024\u2020 Alexa
Smart Properties \u3011 \n    * \u301013\u2020 Tech Talks \u3011 \n\n  *
```